LEARN THE DYNAMICS, ENFORCE THE PHYSICS: GENERATIVE LATENT OPERATORS FOR HIGH-DIMENSIONAL BEAM EVOLUTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Particle-in-cell (PIC) simulation is indispensable for modeling beam dynamics in particle accelerators, but its computational cost limits many-query workflows, such as configuration exploration, virtual diagnostics, and controller design. We introduce *GLOBE*, a physics-constrained **G**enerative **L**atent **O**perator for **B**eam **E**volution, modeling the nonlinear, control-dependent, multiscale evolution of 3D beam–field dynamics covering 10^7 spatial cells. GLOBE compresses beam–field states into a compact latent space and factors their evolution into two components. A control-conditioned neural operator learns the dominant dynamics, while an amplitude-normalized generative model captures the residual distribution, recovering fine-scale structures and conditional variations. GLOBE enforces field constraints by regime-specific, parameter-free operators: a Poisson solve reconstructs the electric field in the electrostatic regime, and a spectral projection imposes Gauss's law and a divergence-free magnetic field in the electromagnetic regime. On two high-resolution 3D particle-accelerator benchmarks, bunched-beam transport and plasma-wakefield acceleration, GLOBE lowers relative-L_2 error by 9% and 13%, low-frequency spectral error by 69% and 51%, and mid-frequency spectral error by 54% and 20% relative to the strongest baseline (DPOT), satisfies field constraints, and reduces hours of PIC simulation to a minute per rollout.

1 INTRODUCTION

Particle accelerators are foundational instruments across high-energy physics, materials science, and medical therapy. Their beam dynamics are governed by the Vlasov–Maxwell equations, where charged particles evolve collectively with their self-generated electromagnetic fields, producing nonlinear phenomena such as emittance growth, instabilities, and particle loss. As direct experimentation on the physical machine is constrained, high-fidelity simulation is indispensable to accelerator design and operation. The particle-in-cell (PIC) method (Birdsall & Langdon, 2004; Hockney & Eastwood, 2021) is the standard tool, advancing millions of macroparticles together with their electromagnetic fields resolved on a spatial mesh. At high resolution and over long horizons, however, each simulation costs hours of GPU time, prohibitive for many-query workflows such as configuration exploration, virtual diagnostics, and controller design (Emma et al., 2018; Edelen et al., 2020; Arpaia et al., 2021). This computational burden has motivated efficient data-driven surrogates for particle accelerators. Existing works map machine settings to low-dimensional beam quantities (Edelen et al., 2016; 2020), reconstruct phase-space projections from indirect measurements (Roussel et al., 2023; Scheinker, 2024a;b; 2025; Scheinker et al., 2026), or forecast beam evolution across successive linac modules (Rautela et al., 2024). Yet efficient long-horizon prediction for the 3D beam–field evolution on the simulation mesh remains largely unexplored.

Advances in scientific machine learning have established two prominent paradigms for PDE modeling. Neural operators learn function-to-function mappings for PDE solution operators and achieve strong performance across various physical systems (Lu et al., 2021; Li et al., 2021; Cao, 2021; Li et al., 2023a; Hao et al., 2023; Li et al., 2023b; Hao et al., 2024; Wu et al., 2024; Rahman et al., 2024; Wang et al., 2025). But they exhibit spectral bias, yielding over-smoothed predictions with unresolved fine-scale structures (Rahaman et al., 2019; Xu et al., 2025). Generative models offer a complementary capability in recovering fine-scale structures and representing predictive variability,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

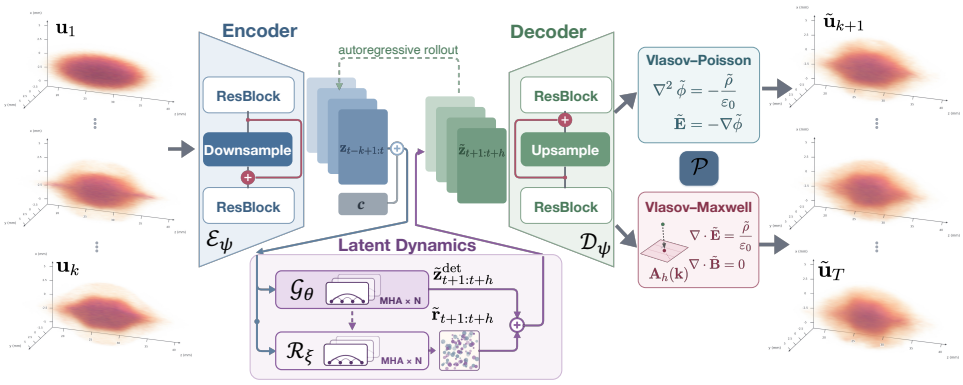

Figure 1: **Framework Overview:** The encoder $\mathcal{E}_\psi$ compresses initial beam–field states $\mathbf{u}_{1:k}$ into latent $\mathbf{z}_{1:k}$. At each rollout step t, the latest k latents $\mathbf{z}_{t-k+1:t}$ and machine operation controls $\mathbf{c}$ are given to the latent dynamics models. The neural operator $\mathcal{G}_\theta$ predicts the dominant evolution $\tilde{\mathbf{z}}^{\text{det}}_{t+1:t+h}$ and the flow-matching model $\mathcal{R}_\xi$ samples the residual $\tilde{\mathbf{r}}_{t+1:t+h}$. Their sum $\tilde{\mathbf{z}}_{t+1:t+h}$ is appended to the history and the prediction advances. Finally, $\mathcal{D}_\psi$ reconstructs the mesh-resolved states and $\mathcal{P}$ enforces regime-specific field constraints.

with successes in partially observed PDEs, turbulent-flow modeling, and probabilistic weather forecasting (Shysheya et al., 2024; Huang et al., 2024; Lienen et al., 2024; Du et al., 2024; Li et al., 2024; Price et al., 2025). But evolving the forecast generatively is challenging, as the model must jointly learn temporal dynamics and conditional variations. These complementary strengths have motivated hybrid surrogates that combine deterministic forecast with generative refinement, either to resolve spectral content (Oommen et al., 2025; Khodakarami et al., 2025) or to model correction distributions around deterministic predictions (Couairon et al., 2026; Mardani et al., 2025). However, these hybrid designs scale poorly to high-resolution 3D systems with heterogeneous field amplitudes.

Motivated by these gaps, we propose *GLOBE*, a physics-constrained **G**enerative **L**atent **O**perator for **B**eam **E**volution, extending hybrid PDE modeling to high-resolution 3D beam–field evolution over 10^7 spatial cells. For tractability, GLOBE compresses full-resolution beam–field states into a compact latent space. In this latent space, a control-conditioned neural operator captures the dominant dynamics, while an amplitude-normalized conditional generative model learns the residual distribution to recover fine-scale structures and sampling-induced conditional variability. Regime-specific, parameter-free operators enforce field constraints by construction, separating dynamics learning from constraint satisfaction. Our main contributions are summarized as follows:

- We design GLOBE, to our knowledge, the first hybrid surrogate for 3D beam–field evolution accommodating 10^7 spatial cells, factoring the latent forecast into a control-conditioned deterministic operator and an amplitude-normalized generative residual.
- We introduce regime-specific, parameter-free operators that enforce field constraints by construction: a Poisson solve for electric-field reconstruction in the electrostatic regime and a spectral Maxwell projection in the electromagnetic regime.
- We release two high-resolution 3D particle-accelerator benchmarks. On both, GLOBE attains lower relative-L_2 error by 9% and 13%, low-frequency spectral error by 69% and 51%, and mid-frequency spectral error by 54% and 20% relative to the strongest baseline (DPOT), satisfies field constraints, and reduces hours of simulation to a minute per rollout.

2 Literature Review

2.1 Deep Learning for PDE Modeling

Neural operators learn mappings between function spaces, enabling one model to generalize across families of initial conditions, boundary conditions, and system parameters. Representative architectures include DeepONet (Lu et al., 2021), Fourier Neural Operator (Li et al., 2021; Kovachki et al., 2021; Guibas et al., 2022; Hao et al., 2024), and attention-based operators (Cao, 2021; Li et al., 2023a; Hao et al., 2023; Li et al., 2023b; Wu et al., 2024; Rahman et al., 2024; McCabe et al., 2024; Wang et al., 2025; Holzschuh et al., 2026). To reduce cost in high dimensions, latent operator methods evolve dynamics in compressed latent spaces (Wu et al., 2022; Wang & Wang, 2024; Alkin et al.,

2024; Serrano et al., 2024). However, neural operators exhibit spectral bias, yielding over-smoothed predictions missing fine-scale features and sharp gradients (Rahaman et al., 2019; Xu et al., 2025).

In parallel, generative models learn distributions over PDE solutions and dynamical states. Diffusion-, score-, and flow-based models have enabled conditional generation, inverse reconstruction, and spatiotemporal prediction of PDE systems (Shysheya et al., 2024; Huang et al., 2024; Li et al., 2025), with applications to turbulent-flow modeling and probabilistic weather forecasting (Lienen et al., 2024; Du et al., 2024; Li et al., 2024; Andrae et al., 2025; Price et al., 2025; Kohl et al., 2026). However, modeling the complete evolution and its conditional structure generatively is challenging, and repeated denoising or numerical integration at each forecast increases rollout cost.

To combine neural operators' modeling capability with generative models' probabilistic representation, recent studies explore hybrid deterministic–generative formulations. Oommen et al. (2025); Khodakarami et al. (2025) couple neural operators with conditional diffusion to recover high-frequency structures missed by deterministic forecasts, mitigating spectral bias in turbulent and multiscale systems. Couairon et al. (2026); Mardani et al. (2025) learn generative corrections around deterministic predictions for high-resolution downscaling and probabilistic weather forecasting. These studies demonstrate the effectiveness of hybrid decomposition across turbulence and weather applications, but its extension to high-resolution 3D systems remains challenging: the cost of state-space model scales with the grid size, and generative training is biased by heterogeneous field amplitudes.

2.2 Learning-based Methods for Particle Accelerators

Machine learning is emerging as a promising approach in accelerator physics, helping address the challenges of high-dimensional configuration spaces and expensive simulations (Edelen et al., 2018; Arpaia et al., 2021). Early methods learn mappings from accelerator configurations to beam-quality metrics for rapid modeling and design (Edelen et al., 2016; 2020), and pursue data-driven tuning and control via adaptive feedback (Scheinker et al., 2018; 2019; Leemann et al., 2019), Bayesian optimization (Duris et al., 2020; Roussel et al., 2024a), and reinforcement learning (Kain et al., 2020; Kaiser et al., 2024). Recent works reconstruct the beam's phase-space distribution from indirect diagnostics. Approaches include differentiable beam-dynamics simulations fit to measurements (Roussel et al., 2023; 2024b), adaptive latent-space tuning that tracks time-varying distributions (Scheinker et al., 2021; Scheinker, 2025), and conditional generative models that map machine signals to phase-space projections (Scheinker, 2024a;b; Scheinker et al., 2026). Beyond reconstruction, Rautela et al. (2024) forecast downstream phase-space projections across successive linac modules with a latent autoregressive generative model. Despite this progress, efficient long-horizon prediction of 3D mesh-resolved, self-consistent beam–field states remains underexplored.

3 Preliminary: Beam Dynamics

Vlasov–Maxwell Equations. The collective beam dynamics in particle accelerators are governed by the relativistic Vlasov–Maxwell equations (Glassey, 1996; Davidson & Qin, 2001). For particles of charge q and mass m, let $f(\mathbf{r}, \mathbf{p}, t)$ be the phase-space distribution of beam population at time t, position $\mathbf{r} = (x, y, z) \in \mathbb{R}^3$, and momentum $\mathbf{p} = \gamma m \mathbf{v} \in \mathbb{R}^3$, where velocity $\mathbf{v} = \dot{\mathbf{r}}$, relativistic factor $\gamma = \sqrt{1 + |\mathbf{p}|^2/(m^2c^2)}$, and speed of light c. It evolves according to the Vlasov equation,

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{r}} f + \mathbf{F} \cdot \nabla_{\mathbf{p}} f = 0, \qquad \mathbf{F} = \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = q\left(\mathbf{E} + \mathbf{v} \times \mathbf{B}\right), \tag{1}$$

where $\mathbf{F}$ is the Lorentz force, $\mathbf{E}$ and $\mathbf{B}$ are the electric and magnetic fields. These fields consist of externally applied fields and self-fields generated by the beam: $\mathbf{E} = \mathbf{E}_{\text{ext}} + \mathbf{E}_{\text{beam}}$ and $\mathbf{B} = \mathbf{B}_{\text{ext}} + \mathbf{B}_{\text{beam}}$. External fields are generated by accelerator components such as magnets and RF cavities, while self-fields are generated by beam charge density ρ and current density $\mathbf{J}$, given by

$$\rho(\mathbf{r}, t) = q \int_{\mathbb{R}^3} f(\mathbf{r}, \mathbf{p}, t)\, \mathrm{d}\mathbf{p}, \quad \mathbf{J}(\mathbf{r}, t) = q \int_{\mathbb{R}^3} \mathbf{v}(\mathbf{p}) f(\mathbf{r}, \mathbf{p}, t)\, \mathrm{d}\mathbf{p}.$$

The beam self-fields satisfy Maxwell's equations:

$$\nabla_{\mathbf{r}} \cdot \mathbf{E}_{\text{beam}} = \frac{\rho}{\varepsilon_0}, \nabla_{\mathbf{r}} \cdot \mathbf{B}_{\text{beam}} = 0, \frac{\partial \mathbf{B}_{\text{beam}}}{\partial t} = -\nabla_{\mathbf{r}} \times \mathbf{E}_{\text{beam}}, \frac{\partial \mathbf{E}_{\text{beam}}}{\partial t} = c^2 \nabla_{\mathbf{r}} \times \mathbf{B}_{\text{beam}} - \frac{\mathbf{J}}{\varepsilon_0}, \tag{2}$$

where ε_0 is the vacuum permittivity. Equations 1–2 form a nonlinear self-consistent system: f determines $(\rho, \mathbf{J})$ and the beam self-fields, which in turn govern f through the Lorentz force $\mathbf{F}$.

From Vlasov–Maxwell to Vlasov–Poisson. In the low-energy regime, the magnetic and inductive effects associated with the beam self-fields are negligible. Therefore we take the time-varying beam magnetic field $\mathbf{B}_{\text{beam}} \approx 0$. By Faraday's law, this gives $\nabla_{\mathbf{r}} \times \mathbf{E}_{\text{beam}} \approx 0$, so the beam electric field is approximately curl-free and can be expressed as $\mathbf{E}_{\text{beam}} = -\nabla_r \phi$, where $\phi(\mathbf{r}, t)$ is the electrostatic scalar potential. The Vlasov–Maxwell equation then reduces to the Vlasov–Poisson equation,

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{r}} f + q\left(\mathbf{E}_{\text{ext}} + \mathbf{v} \times \mathbf{B}_{\text{ext}} - \nabla_{\mathbf{r}} \phi\right) \cdot \nabla_{\mathbf{p}} f = 0, \quad -\nabla_{\mathbf{r}}^2 \phi = \frac{\rho}{\varepsilon_0} = \frac{q}{\varepsilon_0} \int_{\mathbb{R}^3} f \, \mathrm{d}\mathbf{p}. \tag{3}$$

Unlike the full electromagnetic case, the electrostatic beam field has no independent temporal evolution. Given the instantaneous charge density ρ and corresponding boundary conditions, ϕ is determined by the Poisson equation, and $\mathbf{E}_{\text{beam}}$ follows directly from $-\nabla_{\mathbf{r}} \phi$ (Davidson & Qin, 2001).

4 Proposed Framework

This section introduces GLOBE, our physics-constrained Generative Latent Operator for Beam Evolution. As shown in Figure 1, GLOBE has three modules: (i) a compressive autoencoder $(\mathcal{E}_\psi, \mathcal{D}_\psi)$ learning a compact representation of beam–field states, (ii) a latent dynamics model combining a deterministic neural operator $\mathcal{G}_\theta$ with a flow-matching generative residual $\mathcal{R}_\xi$, and (iii) regime-specific, parameter-free operators $\mathcal{P}$ enforcing instantaneous field constraints. We detail each module below.

Problem Setup. Let $\mathbf{u}_t \in \mathbb{R}^{C \times Z \times Y \times X}$ be a beam–field state at time t on a $Z \times Y \times X$ grid with C variables. Given the observed history $\mathbf{u}_{1:k}$ and machine operation controls $\mathbf{c}$, GLOBE models the conditional distribution of future states $p(\mathbf{u}_{k+1:T} \mid \mathbf{u}_{1:k}, \mathbf{c})$, rolled out autoregressively in blocks of h steps. We denote $\hat{\mathbf{u}}(\mathbf{k}) = \int_{\mathbb{R}^3} \mathbf{u}(\mathbf{x}) e^{-i\mathbf{k}\cdot\mathbf{x}} \mathrm{d}\mathbf{x}$ as the Fourier transform of $\mathbf{u}$ at frequency $\mathbf{k} \in \mathbb{R}^3$.

4.1 Compressive Autoencoder

GLOBE maps the high-dimensional 3D accelerator beam–field states into a compact latent representation, in which subsequent dynamics are modeled. As illustrated in Figure 2, we employ a convolutional autoencoder $(\mathcal{E}_\psi, \mathcal{D}_\psi)$ to learn this representation. The encoder compresses each beam–field state $\mathbf{u}_t$ into latent $\mathbf{z}_t = \mathcal{E}_\psi(\mathbf{u}_t) \in \mathbb{R}^{C' \times Z' \times Y' \times X'}$ at reduced resolution, from which the decoder reconstructs $\tilde{\mathbf{u}}_t = \mathcal{D}_\psi(\mathbf{z}_t)$. The compression substantially reduces the number of spatial tokens from 10^7 to $\sim 10^4 - 10^5$ while retaining a structured 3D grid for latent dynamics modeling.

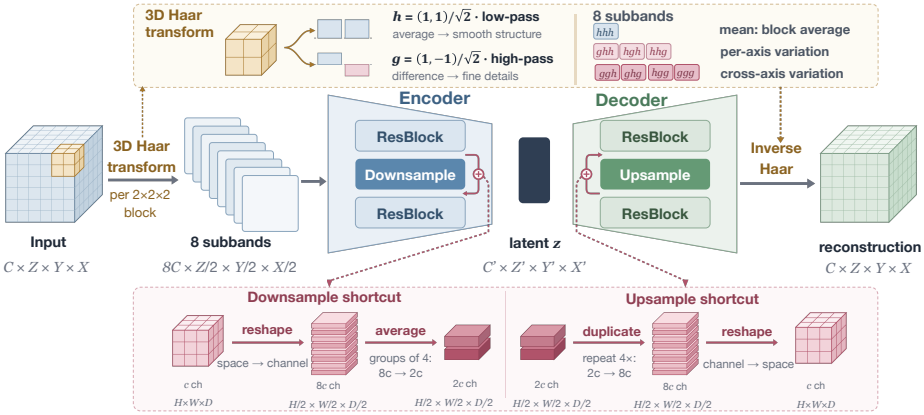

Figure 2: **Compressive Autoencoder**: For each 3D input, every $2 \times 2 \times 2$ block is split by the low-/high-pass pair $(\mathbf{h}, \mathbf{g})$ into 8 wavelet subbands, then the encoder $\mathcal{E}_\psi$ compresses it to a grid latent $\mathbf{z}$. The decoder $\mathcal{D}_\psi$ mirrors the hierarchy, closing with the inverse Haar transform. Around every down-/up-sampling module, a parameter-free space–channel shortcut ($\oplus$) provides references across resolution by channel averaging and duplication with reshaping, so the corresponding convolutional branch fits only a residual correction.

Autoencoder Backbone. The encoder composes a sequence of residual blocks and down-sampling modules, while the decoder mirrors this structure with up-sampling modules and residual blocks, without encoder–decoder skip connections. Each residual block operates at a fixed resolution for progressive feature refinement, whereas the down- and up-sampling modules perform the transitions between resolution scales. To preserve fine-scale structures and stabilize training under strong compression, we augment the backbone with two parameter-free designs.

Haar Transform and Multiscale Resolution Shortcuts. First, the 3D Haar transform decomposes each non-overlapping $2 \times 2 \times 2$ spatial block into eight wavelet subbands: one local average, three per-axis variations, and four cross-axis variations. It is applied before the encoder and inverted after the decoder. By separating local variations from smooth background before learned compression, the transform helps preserve fine beam structures (Dai et al., 2026). Second, each learned down- and up-sampling module is paired with a space–channel shortcut that maps features between resolutions (Chen et al., 2025). During down-sampling, the shortcut reshapes spatial blocks into channels and adjusts the channel width through group averaging. During up-sampling, channel features are duplicated and rearranged back into the spatial dimensions. This multiscale path provides a reference across resolutions, allowing the learned resolution-changing modules to focus only on the correction needed for accurate reconstruction, which eases optimization at high compression ratios.

Noise-robust Reconstruction Objective. The autoencoder is trained to accurately reconstruct the beam and field quantities while preserving the spatial structures with three complementary losses:

$$\mathcal{L}_{\text{rec}}(\mathbf{u}_t, \tilde{\mathbf{u}}_t) = \underbrace{\ell(\mathbf{u}_t, \tilde{\mathbf{u}}_t)}_{\mathcal{L}_{\text{rel}}} + \lambda_{\text{fd}} \underbrace{\sum_{j=1}^{3} \ell\big(\partial_{x_j} \mathbf{u}_t, \partial_{x_j} \tilde{\mathbf{u}}_t\big)}_{\mathcal{L}_{\text{fd}}} + \lambda_{\text{spec}} \underbrace{\ell\big(\mathbf{w} \odot \hat{\mathbf{u}}_t,\ \mathbf{w} \odot \hat{\tilde{\mathbf{u}}}_t\big)}_{\mathcal{L}_{\text{spec}}}, \tag{4}$$

where $\ell(\mathbf{a}, \mathbf{b}) = \frac{\|\mathbf{a}-\mathbf{b}\|_2}{\max(\|\mathbf{a}\|_2, \epsilon)}$ is the relative-L_2 error with $\epsilon \in \mathbb{R}^+$ for numerical stability, $\odot$ is element-wise multiplication, and $\lambda_{\text{spec}}, \lambda_{\text{fd}} \in \mathbb{R}^+$ are loss weights. $\mathcal{L}_{\text{rel}}$ measures the overall relative error between the original and reconstructed states. $\mathcal{L}_{\text{fd}}$ preserves local gradient consistency, retaining sharp spatial variations such as beam edges that are under-weighted by relative reconstruction loss alone. $\mathcal{L}_{\text{spec}}$ promotes consistency of spectral content with $\mathbf{w} = (w(\mathbf{k}))_{\mathbf{k}}$, a vector of spectral weights over all Fourier modes that grow with $|\mathbf{k}|$ and additionally emphasize the mid-frequency band. It encourages coherent mid-frequency structure while reducing the dominance of low-frequency energy and suppressing high-frequency noise. To make the decoder robust to imperfect predicted latent states during inference, we apply $\mathcal{L}_{\text{rec}}$ to both the original latent $\mathbf{z}_t$ and the noise-perturbed latent. Together, we formulate the noise-robust reconstruction objective as:

$$\mathcal{L}_{\text{AE}}(\mathbf{u}_t, \tilde{\mathbf{u}}_t) = \mathcal{L}_{\text{rec}}\left(\mathbf{u}_t, \tilde{\mathbf{u}}_t\right) + \lambda_{\text{noise}} \mathcal{L}_{\text{rec}}\Big(\mathbf{u}_t, \mathcal{D}_\psi\big(\mathbf{z}_t + \sigma \text{std}(\mathbf{z}_t)\boldsymbol{\epsilon}\big)\Big),\ \sigma \sim \mathcal{U}(0, \sigma_{\max}),\ \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}).$$

This objective maintains reconstruction fidelity while improving decoder's robustness to latent error.

4.2 Latent Dynamics: Deterministic Anchor and Generative Residual

Deterministic neural operators can model dynamical evolution accurately, but tend to attenuate weaker fine-scale structures and average over realization-dependent variations. Fully generative dynamics, in contrast, burden the generative model with learning evolution that is largely predictable. GLOBE therefore uses a deterministic forecast as the anchor and pairs it with a conditional generative residual that refines unresolved fine-scale structures and captures sampling-induced variations.

Deterministic Latent Neural Operator. Given the latest k-step latent histories and prescribed machine operation controls $\mathbf{c}$, the deterministic operator $\mathcal{G}_\theta$ predicts the next h-step latent states, trained against the encoded ground truth with a relative-L_2 objective,

$$\tilde{\mathbf{z}}^{\text{det}}_{t+1:t+h} = \mathcal{G}_\theta(\mathbf{z}_{t-k+1:t}, \mathbf{c}), \quad \mathcal{L}_{\mathcal{G}_\theta} = \frac{\|\mathbf{z}_{t+1:t+h} - \tilde{\mathbf{z}}_{t+1:t+h}\|_2}{\max(\|\mathbf{z}_{t+1:t+h}\|_2, \epsilon)}, \quad \epsilon \in \mathbb{R}^+.$$

The prediction $\tilde{\mathbf{z}}^{\text{det}}$ captures the dominant temporal evolution, and serves as a dynamical anchor providing the reference around which the residual distribution is modeled. Since autoregressive rollout feeds the operator its imperfect predictions rather than the exact latent histories seen in training, we use two techniques to limit error accumulation. First, Gaussian perturbations are added to the input latent histories during training, exposing the operator to small deviations from the encoded trajectory (Sanchez-Gonzalez et al., 2020). In addition, we employ pushforward training, feeding the operator its own prediction for a subsequent supervised forecast (Brandstetter et al., 2022).

Generative Latent Residual. We define the discrepancy around the deterministic anchor as $\mathbf{r}_{t+1:t+h} = \mathbf{z}_{t+1:t+h} - \tilde{\mathbf{z}}^{\text{det}}_{t+1:t+h}$ and model it with a generative model $\mathcal{R}_\xi$. Because its magnitude varies substantially across physical variables, forecast stages, and spatial locations, modeling the raw residual would place disproportionate weight on large-amplitude components. We therefore normalize it before training by a conditional amplitude factorized at three granularities,

$$\mathbf{s}_\omega(\mathbf{p}) = \boldsymbol{\sigma}_r \odot s_{\text{win}}(\mathbf{p}) \odot s_{\text{tok}}(\mathbf{p}), \quad \bar{\mathbf{r}}_{t+1:t+h} = \mathbf{r}_{t+1:t+h} \oslash \mathbf{s}_\omega(\mathbf{p}), \quad \mathbf{p} = (\mathbf{z}_{t-k+1:t}, \tilde{\mathbf{z}}^{\text{det}}_{t+1:t+h}),$$

where $\odot$ and $\oslash$ denote element-wise multiplication and division. $\boldsymbol{\sigma}_r \in \mathbb{R}^{1\times C'\times 1\times 1\times 1}$ is the per-channel root-mean-square (RMS) of the residual over the training set, $s_{\text{win}} > 0$ adjusts the amplitude of each forecast block, and $s_{\text{tok}} \in \mathbb{R}^{1\times 1\times Z'\times Y'\times X'}$ captures spatial amplitude variation shared across channels and steps. The latter two normalizers are predicted sequentially by small networks from the conditioning context $\mathbf{p}$. Each network minimizes the zero-mean negative Gaussian log-likelihood $\frac{m}{s(\mathbf{p})^2} + 2\log\big(s(\mathbf{p})\big)$, where m is the mean-square residual after the preceding normalization. For s_{win}, m is averaged over the full forecast block after channel normalization, and for s_{tok}, m is averaged over steps and channels at each spatial token after channel and window normalization. The conditional optimum satisfies $s(\mathbf{p})^2 = \mathbb{E}[m \mid \mathbf{p}]$, so each network predicts the conditional RMS at its granularity, yielding an approximately unit-RMS normalized residual $\bar{\mathbf{r}}$.

We learn the normalized residual distribution with conditional flow matching (Liu et al., 2023; Lipman et al., 2023). Given Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and a logit-normal flow time (Esser et al., 2024) $\tau = \text{sigmoid}(\eta),\ \eta \sim \mathcal{N}(0,1)$, we construct the linear interpolant $\mathbf{x}_\tau = (1-\tau)\boldsymbol{\epsilon} + \tau\bar{\mathbf{r}}$ and train the conditional velocity field $\mathbf{v}_\xi$ to regress its constant velocity $(\bar{\mathbf{r}} - \boldsymbol{\epsilon})$,

$$\mathcal{L}_{\mathcal{R}_\xi} = \mathbb{E}_{\tau,\boldsymbol{\epsilon},\bar{\mathbf{r}}}\left[\|\mathbf{v}_\xi(\mathbf{x}_\tau; \mathbf{p}, \mathbf{c}, \tau) - (\bar{\mathbf{r}} - \boldsymbol{\epsilon})\|_2^2\right] + \lambda_{\text{ES}}\mathcal{L}_{\text{ES}}, \quad \lambda_{\text{ES}} \in \mathbb{R}^+,$$

where $\mathcal{L}_{\text{ES}}$ is an energy-score regularizer (Pacchiardi et al., 2024) that keeps generated samples close to target while preserving ensemble dispersion. At inference, $\mathcal{R}_\xi$ maps Gaussian noise into a normalized residual by integrating $\mathbf{v}_\xi$ with Euler's method. Rescaling by $\mathbf{s}_\omega(\mathbf{p})$, the predicted residual $\tilde{\mathbf{r}}_{t+1:t+h}$ is added to the deterministic forecast, yielding the final latent prediction $\tilde{\mathbf{z}}_{t+1:t+h}$.

Backbone Architecture. Transformer (Vaswani et al., 2017) with self-attention over spatial latent tokens is used for the deterministic neural operator $\mathcal{G}_\theta$ and generative residual model $\mathcal{R}_\xi$. As self-attention is permutation-equivariant, sinusoidal Fourier features of the normalized 3D coordinates are used for positional embeddings (Tancik et al., 2020). Machine operation controls $\mathbf{c}$, together with flow time τ for $\mathcal{R}_\xi$, are injected into each block via adaLN-Zero (Peebles & Xie, 2023).

4.3 Physics-constrained Operators

To preserve the instantaneous field constraints, GLOBE applies a parameter-free, regime-specific operator $\mathcal{P}$ after decoding, enforcing instantaneous field constraint satisfaction by construction.

Vlasov–Maxwell Constraint Enforcement. In the electromagnetic regime, the charge density and electromagnetic fields carry independent dynamical information and are therefore predicted jointly. By Equation 2, the decoded prediction $\tilde{\mathbf{u}} = (\tilde{\rho}, \tilde{\mathbf{E}}, \tilde{\mathbf{B}})$ should satisfy

$$\text{Gauss's Law: } \nabla\cdot\mathbf{E} - \frac{\rho}{\varepsilon_0} = 0, \qquad \text{Divergence-free magnetic field: } \nabla\cdot\mathbf{B} = 0. \tag{5}$$

These constraints are linear differential relations. Under the Fourier transform, spatial differentiation becomes multiplication by the corresponding frequency component, $\partial_{x_j} \to ik_j$. Constraints in Equation 5 therefore become, independently at each Fourier mode $\mathbf{k}$,

$$i\mathbf{k}^\mathsf{T}\hat{\mathbf{E}}(\mathbf{k}) - \frac{\hat{\rho}(\mathbf{k})}{\varepsilon_0} = 0, \qquad i\mathbf{k}^\mathsf{T}\hat{\mathbf{B}}(\mathbf{k}) = 0.$$

As a result, the admissible state values form a linear subspace that can be written as $\mathbf{A}(\mathbf{k})\hat{\mathbf{u}}(\mathbf{k}) = \mathbf{0}$. We define $\mathcal{P}_\mathbf{k}$ as a weighted equality-constraint optimization solution operator that projects the decoded prediction $\hat{\tilde{\mathbf{u}}}(\mathbf{k})$ to its closest constraint-satisfying coefficient $\hat{\mathbf{v}}^\star(\mathbf{k})$:

$$\hat{\mathbf{v}}^\star(\mathbf{k}) := \mathcal{P}_\mathbf{k}\big(\hat{\tilde{\mathbf{u}}}(\mathbf{k})\big) = \underset{\mathbf{A}(\mathbf{k})\hat{\mathbf{v}}(\mathbf{k})=0}{\arg\min} \|\hat{\mathbf{v}}(\mathbf{k}) - \hat{\tilde{\mathbf{u}}}(\mathbf{k})\|_{\mathbf{W}^{-1}} = \Big[\mathbf{I} - \mathbf{W}\mathbf{A}(\mathbf{k})^\mathsf{H}\big(\mathbf{A}(\mathbf{k})\mathbf{W}\mathbf{A}(\mathbf{k})^\mathsf{H}\big)^\dagger\mathbf{A}(\mathbf{k})\Big]\hat{\tilde{\mathbf{u}}}(\mathbf{k}).$$

where $\mathbf{W} \succ 0$ accounts for value-scale differences across variables. $(\cdot)^\mathsf{H}$ and $(\cdot)^\dagger$ denote the Hermitian transpose and pseudo-inverse. Applying the projection to each Fourier mode and transforming the corrected coefficients back to the physical space yields a state $\mathbf{v}^\star$ satisfying the instantaneous Maxwell constraints by construction. In implementation, derivatives are evaluated as finite differences, acting in Fourier space as multiplication by a modified frequency, e.g. $i\sin(k_j h)/h$ for a central difference of spacing h. Thus we build $\mathbf{A}_h(\mathbf{k})$ from these multipliers so the projected state satisfies the constraints in simulation's own discretization. Details are in Appendix B.4.

Vlasov–Poisson Field Reconstruction. As established in Section 3, in the electrostatic regime, the beam field is uniquely determined by the charge density and the boundary condition:

$$\text{Poisson Equation: } -\nabla_\mathbf{r}^2\phi = \frac{\rho}{\varepsilon_0}, \qquad \mathbf{E} = -\nabla_\mathbf{r}\phi.$$

Therefore, GLOBE predicts only the charge density $\tilde{\rho}$, solves the Poisson equation for the predicted scalar potential $\tilde{\phi}$ with an iterative Krylov solver (Jacobi-preconditioned BiCGStab (van der Vorst, 1992)), and obtains the predicted beam electric field $\tilde{\mathbf{E}}$ by applying the negative discrete gradient to $\tilde{\phi}$. This enforces the Poisson equation, Gauss's law, and the curl-free condition by construction.

5 EXPERIMENTS

We evaluate GLOBE on two high-resolution 3D particle-accelerator benchmarks generated with WarpX (Vay et al., 2018), one per regime in Section 3. (1) *Bunched-beam transport (electrostatic regime)*: a $750\,\mathrm{keV}$ proton bunch propagates through a periodic quadrupole–RF focusing channel. The quadrupole and RF strengths are tuned with bunch charge to maintain beam matching, as mismatch drives envelope oscillations, emittance growth, and halo formation that degrade downstream beam quality. The benchmark varies focusing strength, bunch charge, and initial mismatch across 464 simulations (52 for testing), each with $T = 401$ charge-density frames on a $256 \times 336 \times 336$ grid over 80 focusing periods, requiring $\sim$ 6600s GPU per trajectory. (2) *Plasma-wakefield acceleration (PWFA; electromagnetic regime)*: an ultra-relativistic electron driver excites a nonlinear plasma wake with multi-GV/m gradients, accelerating a trailing witness bunch. Driver charge, energy, and size, and witness charge and position, determine the energy gain and witness beam quality. The benchmark varies these parameters across 504 simulations (56 for testing), each with $T = 60$ frames over ten variables, four species-resolved charge densities and ($\mathbf{E}$, $\mathbf{B}$), on a $336 \times 216 \times 216$ grid over $19.6\,\mathrm{mm}$ of propagation, requiring ~ 4130s GPU per trajectory. Details are in Appendix A.

Baselines: We compare against four representative baselines at matched model size: DPOT (Hao et al., 2024), UPT (Alkin et al., 2024), MPP (McCabe et al., 2024), and P3D (Holzschuh et al., 2026). For P3D, both deterministic (P3D-D) and generative formulation (P3D-G) are compared with. We report three GLOBE variants sharing the same autoencoder: GLOBE-D, the deterministic operator $\mathcal{G}_\theta$ alone, GLOBE-C, the full model with both deterministic operator $\mathcal{G}_\theta$ and generative residual $\mathcal{R}_\xi$, and GLOBE-G, an ablation that forecasts the latent directly with a single flow-matching model.

Evaluation Metrics: We evaluate predictive quality with relative-L_2 error (Rel-L_2) and spectral error $L_\mathcal{F}$, averaged over rollout steps and test trajectories. Following Khodakarami et al. (2025), we split spectrum into low ($|\mathbf{k}| \leq 0.34 k_{\mathrm{Nyq}}$), mid ($0.34 k_{\mathrm{Nyq}} < |\mathbf{k}| \leq 0.49 k_{\mathrm{Nyq}}$), and high ($|\mathbf{k}| > 0.49 k_{\mathrm{Nyq}}$) frequency bands, with k_{Nyq} the per-axis Nyquist frequency. $L_\mathcal{F}$ is the relative error between the radially averaged power spectra of prediction and reference in each band. For generative models we report the mean error over 8 samples. Inference time per trajectory is also reported.

5.1 QUANTITATIVE COMPARISON WITH BASELINES

Table 1 compares the proposed GLOBE models with four representative baselines on both benchmarks. The deterministic baselines include DPOT, UPT, MPP, and P3D-D. Within this group, GLOBE-D achieves the lowest Rel-L_2 error, reaching 0.144 on bunched-beam transport and 0.174 on PWFA, while using fewer parameters than the baselines. We attribute this improvement to full attention in the compressed latent space, which enables global spatial interactions that would be intractable in the raw 3D space. However, like other deterministic models, GLOBE-D has much larger errors at mid-frequency than at low-frequency. Since higher frequencies correspond to smaller spatial scales, this indicates reduced accuracy in capturing finer-scale spatial structure.

Adding the generative residual model yields GLOBE-C, which preserves the overall accuracy of GLOBE-D while better capturing finer-scale beam structures. Notably, it reduces mid-frequency

Table 1: Quantitative comparison of GLOBE and baselines on bunched-beam transport and PWFA.

	Bunched-Beam Transport				**PWFA**			
Method	Params	Rel-L_2 ↓	$L_\mathcal{F}$ (low / mid / high) ↓	Time ↓	Params	Rel-L_2 ↓	$L_\mathcal{F}$ (low / mid / high) ↓	Time ↓
DPOT	91.4M	0.177	0.112 / 0.986 / **0.954**	**7.55**s	109.4M	0.224	0.156 / 0.377 / 0.634	**3.04**s
UPT	92.6M	0.261	0.147 / 0.999 / 1.000	1384s	107.7M	0.378	0.251 / 0.640 / 2.585	114.5s
MPP	90.0M	0.347	0.384 / 0.991 / 0.978	943s	110.9M	0.325	0.443 / 1.279 / 3.212	73.5s
P3D-D	91.4M	0.329	0.197 / 0.961 / 2.160	91.3s	110.9M	0.520	0.314 / 0.789 / 19.3	17.9s
P3D-G	91.4M	0.490	0.437 / **0.323** / 1.115	1490s	111.0M	1.219	0.794 / 151 / 6002	309s
GLOBE-D (Ours)	63.6M	**0.144**	0.039 / 0.994 / 1.003	16.3s	82.4M	**0.174**	0.083 / 0.350 / **0.592**	5.4s
GLOBE-G (Ours)	63.9M	0.251	0.153 / 0.545 / 0.967	73.1s	83.1M	1.954	11.3 / 225 / 909	23.6s
GLOBE-C (Ours)	91.4M	0.161	**0.035** / 0.457 / 0.973	64.7s	111.8M	0.194	**0.077** / **0.300** / 0.618	21.8s

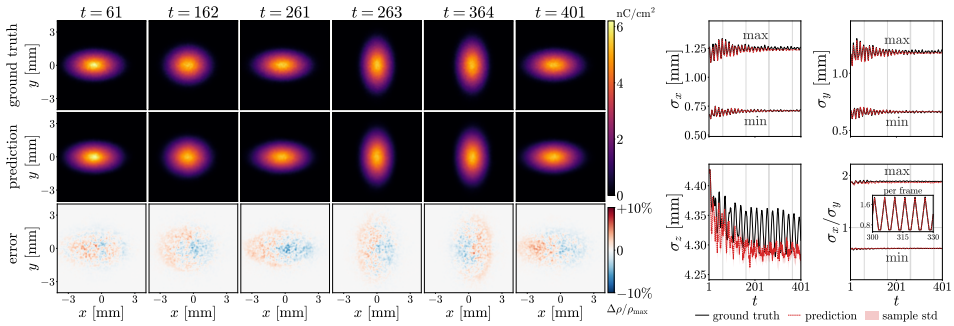

Figure 3: **Bunched-beam transport.** Left: Transverse charge projection $\int \rho \, \mathrm{d}z$, top: ground truth, bottom: one prediction sample from GLOBE-C. Right: Evolution of σ_x, σ_y, σ_z, and σ_x/σ_y. Due to rapid oscillation, the main curves of transverse quantities show period-wise extrema and the insets show frame-wise evolution. Prediction curve is the mean of eight GLOBE-C samples, and shading indicates the sample standard deviation.

error from 0.994 to 0.457 on bunched-beam transport and from 0.350 to 0.300 on PWFA. In contrast, directly modeling the full dynamics via generative models performs poorly. Both P3D-G and GLOBE-G exhibit much larger Rel-L_2 and spectral errors, especially on PWFA. These results support our staged design, where the deterministic model captures the dominant beam–field evolution and the generative model predicts the residual to recover unresolved fine-scale structures.

GLOBE is efficient at inference, as modeling occurs in the compressed latent space. Among deterministic baselines, GLOBE-D uses only 16.3s / 5.4s per rollout. UPT (1384s / 114.5s) incurs query-wise cross-attention during decoding, MPP (943s / 73.5s) attends over spatio-temporal tokens of the full history, and P3D-D (91.3s / 17.9 s) repeatedly processes full 3D states with convolutional and windowed-attention blocks. DPOT remains fastest (7.55s / 3.04s), operating on patchified states with lightweight input/output mappings, whereas GLOBE decodes latent forecasts via 3D convolutional blocks. GLOBE-C avoids full-state generative rollout by integrating flows in latent space and decoding only the final forecasts, achieving 64.7s / 21.8s versus P3D-G 1490 / 309s, whose iterative generation operates on full 3D states. Relative to WarpX, GLOBE-C gains $102\times$ / $189\times$ speedups.

5.2 Long-Horizon Prediction and Visualization

We visualize long-horizon predictions on both benchmarks. Figure 3 shows a held-out test trajectory for bunched-beam transport, conditioned on the initial eight frames and operation controls, followed by a 393-frame prediction with forecast window $h = 8$. In the left panel, we observe that GLOBE-C closely reproduces the time-varying transverse beam shape, the gradual change in beam size, and the halo effect, i.e., the diffuse particle population extending beyond the dense beam core, throughout the trajectory. The right panel further quantifies the beam-envelope evolution through the transverse beam widths σ_x and σ_y, longitudinal bunch size σ_z, and transverse aspect ratio σ_x/σ_y. All four predicted quantities closely track the ground truth without noticeable error accumulation.

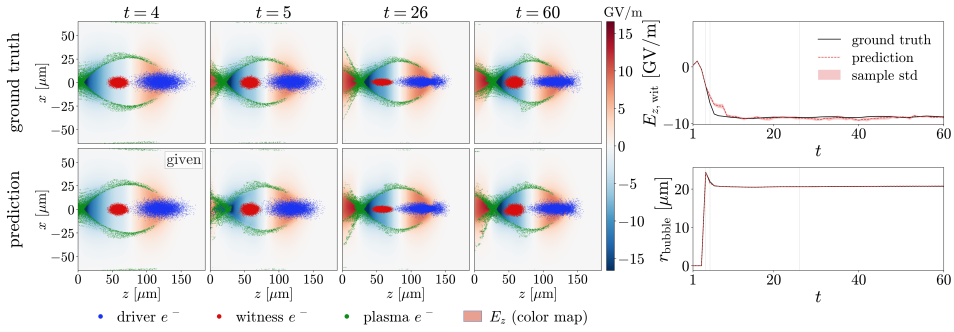

Figure 4: **PWFA.** Left: longitudinal electric field E_z in the x–z mid-plane, overlaid with pseudo-particles of driver, witness, and plasma electrons sampled from charge density fields of WarpX (top) and GLOBE-C (bottom). First column is the last input frame; the rest show the forecast. Right: evolution of $E_{z,\mathrm{wit}}$ and r_{bubble}.

Figure 4 shows a held-out PWFA test trajectory, using the first four frames as input and predicting the subsequent 56 frames with forecast window $h = 4$. In the left panels, the driver bunch (blue)

expels plasma electrons (green), creating an ion cavity enclosed by the curved green electron sheath and a longitudinal wakefield E_z. The witness bunch (red) is located where E_z provides an accelerating force, increasing its energy as it propagates. GLOBE-C closely reproduces the cavity shape, surrounding electron sheath, and witness-bunch evolution in the ground truth. The right panel quantifies this evolution by the accelerating field at the witness centroid $E_{z,\mathrm{wit}}$ and the maximum cavity radius r_{bubble}. Both predictions closely track the ground truth throughout the rollout.

5.3 Ablation Studies

Compressive Autoencoder Design Ablation. Table 2 isolates the benefits of compressive autoencoder's two designs in Section 4.1: the Haar transform and multiscale resolution shortcut. The Haar transform maps each $2 \times 2 \times 2$ block into eight subbands, reducing the spatial grid by $8\times$ while separating local variations from the smooth background. This reduces training runtime by $6.2\times$ and lowers the three losses in Equation 4 by 39%, 31%, and 19%. Adding the shortcut further reduces losses by 16%, 7%, and 14% with nearly unchanged runtime. These results show that the Haar transform provides the main efficiency gain, while the shortcut provides complementary accuracy gains via fixed multiscale references for the learned convolutional branch.

Table 2: Ablation of two designs in our autoencoder: Haar transform (Haar) and multiscale resolution shortcut (Shortcut). The last row combines both designs and corresponds to the proposed autoencoder.

Haar	Shortcut	$\mathcal{L}_{\mathrm{rel}}\downarrow$	$\mathcal{L}_{\mathrm{spec}}\downarrow$	$\mathcal{L}_{\mathrm{fd}}\downarrow$	Runtime (s) $\downarrow$
✗	✗	0.0891	0.1070	0.697	6.04
✓	✗	0.0547	0.0740	0.566	**0.97**
✓	✓	**0.0457**	**0.0685**	**0.489**	1.00

Physics-constrained Operator Ablation. Table 3 evaluates the parameter-free physics-constrained decoding in Section 4.3 on the PWFA dataset. We compare GLOBE-C without the physics-constrained operator (denoted as w/o $\mathcal{P}$) against GLOBE-C with $\mathcal{P}$ applied to enforce the field constraints (denoted as w/ $\mathcal{P}$). Applying $\mathcal{P}$ maintains comparable prediction accuracy (Rel-L_2 0.191 vs. 0.194) while reducing violations of Gauss's law and divergence-free magnetic-field constraint from 10^{-2}–10^{-3} to 10^{-7}, improving physical consistency by four to five orders of magnitude with negligible accuracy loss.

Table 3: Ablation of the physics-constrained operator on PWFA. Constraint residuals are normalized RMS values averaged over frames and trajectories.

	Rel-$L_2\downarrow$	$\nabla\cdot\mathbf{E}-\rho/\varepsilon_0\downarrow$	$\nabla\cdot\mathbf{B}\downarrow$
w/o $\mathcal{P}$	**0.191**	2.72×10^{-2}	6.85×10^{-3}
w/ $\mathcal{P}$	0.194	$\mathbf{2.77\times10^{-7}}$	$\mathbf{2.78\times10^{-7}}$

Latent Backbone Ablation. Table 4 compares GLOBE's global self-attention latent Transformer in Section 4.2 against HFS (Khodakarami et al., 2025), a U-Net based neural operator with diffusion refinement. As the original HFS does not scale to our high-resolution 3D grids, we place it in the same learned latent space as GLOBE with a matched model size. HFS is competitive with GLOBE-D on the single-channel bunched-beam transport (Rel-L_2 0.163 vs. 0.144), and diffusion refinement (HFS + DM) reduces its mid-frequency error from 0.995 to 0.682. On the multi-channel PWFA task, however, HFS degrades substantially (Rel-L_2 0.835 vs. 0.174), with particularly high spectral prediction errors, and HFS+DM further diverges. The larger gap in PWFA highlights the benefit of global attention for strongly coupled multi-channel dynamics. Overall, the results support GLOBE's hybrid design: a robust deterministic operator first captures the dominant dynamics, leaving the generative model to refine the remaining variability.

Table 4: Ablation of latent Transformer backbone vs. HFS.

	Bunched-Beam Transport		**PWFA**	
Method	Rel-$L_2\downarrow$	$L_{\mathcal{F}}\downarrow$	Rel-$L_2\downarrow$	$L_{\mathcal{F}}\downarrow$
HFS	0.163	0.090 / 0.995 / 1.004	0.835	2.04 / 6.44 / 258
GLOBE-D (Ours)	**0.144**	**0.039 / 0.994 / 1.003**	**0.174**	**0.083 / 0.350 / 0.592**
HFS+DM	0.173	0.088 / 0.682 / 1.034	3.258	46.8 / 1301 / 2446
GLOBE-C (Ours)	**0.161**	**0.035 / 0.457 / 0.973**	**0.194**	**0.077 / 0.300 / 0.618**

6 Conclusion

This work introduced GLOBE, a hybrid learning framework for high-resolution 3D beam dynamics in particle accelerators. GLOBE combines a compressed latent representation, a control-conditioned Transformer for dominant dynamics, a generative residual model for fine-scale refinement, and parameter-free operators imposing regime-specific field constraints. Across two particle accelerator benchmarks, GLOBE achieves the best prediction accuracy among the evaluated baselines, satisfies instantaneous field constraints, and reduces hours of PIC simulation to about one minute per rollout. Future work will extend GLOBE toward a foundation model for particle accelerator simulation across machines, control settings, and initial/boundary conditions. An alternative direction is to leverage the learned dynamics for beam-state estimation and control, enabling accurate virtual diagnostics and real-time decision-making across diverse accelerator applications.

AI USE STATEMENT

In this work, we used generative AI tools to assist with coding and implementation of methods designed by the authors, and to polish parts of the manuscript for readability. We did not use generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, propose or refine hypotheses, design or provide feedback on research methodology or experiments, clean or reformat datasets, or interpret results. Providing ingredients for or writing proofs, assisting with translation, and supporting qualitative or thematic data analysis are not applicable to this work. AI-assisted implementations were checked and validated by the authors, and all AI-assisted writing was reviewed for accuracy against the underlying methods and results. The authors take full responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Benedikt Alkin, Andreas Fürst, Simon Schmid, Lukas Gruber, Markus Holzleitner, and Johannes Brandstetter. Universal physics transformers: A framework for efficiently scaling neural operators. *Advances in Neural Information Processing Systems*, 37:25152–25194, 2024.

Martin Andrae, Tomas Landelius, Joel Oskarsson, and Fredrik Lindsten. Continuous ensemble weather forecasting with diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=ePEZvQNFDW.

Pasquale Arpaia, Gabriella Azzopardi, Frederic Blanc, Giuseppe Bregliozzi, Xavier Buffat, Loic Coyle, Elena Fol, Francesco Giordano, Massimo Giovannozzi, Tatiana Pieloni, et al. Machine learning for beam dynamics studies at the cern large hadron collider. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 985:164652, 2021.

Charles K Birdsall and A Bruce Langdon. *Plasma physics via computer simulation*. CRC Press, 2004.

Johannes Brandstetter, Daniel E. Worrall, and Max Welling. Message passing neural PDE solvers. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=vSix3HPYKSU.

Shuhao Cao. Choose a transformer: Fourier or Galerkin. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=ssohLcmn4-r.

Junyu Chen, Han Cai, Junsong Chen, Enze Xie, Shang Yang, Haotian Tang, Muyang Li, and Song Han. Deep compression autoencoder for efficient high-resolution diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=wH8XXUOUZU.

Guillaume Couairon, Renu Singh, Anastase Charantonis, Christian Lessig, and Claire Monteleoni. Archesweathergen: Skillful and compute-efficient probabilistic weather forecasting with machine learning. *Science Advances*, 12(17):eadx2372, 2026. doi: 10.1126/sciadv.adx2372. URL https://www.science.org/doi/abs/10.1126/sciadv.adx2372.

Yilong Dai, Yiming Sun, Yiheng Chen, Ziyi Wang, Shengyu Chen, Xiaowei Jia, and Runlong Yu. Physics-preserving latent compression for zero-shot resolution transfer in 3D turbulence. *arXiv preprint arXiv:2606.21781*, 2026.

Ronald C Davidson and Hong Qin. *Physics of Intense Charged Particle Beams in High Energy Accelerators*. World Scientific, 2001. doi: 10.1142/p250. URL https://www.worldscientific.com/doi/abs/10.1142/p250.

Pan Du, Meet Hemant Parikh, Xiantao Fan, Xin-Yang Liu, and Jian-Xun Wang. Conditional neural field latent diffusion model for generating spatiotemporal turbulence. *Nature Communications*, 15(1):10416, 2024.

Joseph Duris, Dylan Kennedy, Adi Hanuka, Jane Shtalenkova, Auralee Edelen, P Baxevanis, Adam Egger, T Cope, M McIntire, S Ermon, et al. Bayesian optimization of a free-electron laser. *Physical review letters*, 124(12):124801, 2020.

Auralee Edelen, Christopher Mayes, Daniel Bowring, Daniel Ratner, Andreas Adelmann, Rasmus Ischebeck, Jochem Snuverink, Ilya Agapov, Raimund Kammering, Jonathan Edelen, Ivan Bazarov, Gianluca Valentino, and Jorg Wenninger. Opportunities in machine learning for particle accelerators, 2018. URL https://arxiv.org/abs/1811.03172.

Auralee Edelen, Nicole Neveu, Matthias Frey, Yannick Huber, Christopher Mayes, and Andreas Adelmann. Machine learning for orders of magnitude speedup in multiobjective optimization of particle accelerator systems. *Physical Review Accelerators and Beams*, 23(4):044601, 2020.

Auralee Linscott Edelen, SG Biedron, BE Chase, D Edstrom, SV Milton, and P Stabile. Neural networks for modeling and control of particle accelerators. *IEEE Transactions on Nuclear Science*, 63(2):878–897, 2016.

Claudio Emma, A Edelen, MJ Hogan, B O'Shea, G White, and V Yakimenko. Machine learning-based longitudinal phase space prediction of particle accelerators. *Physical Review Accelerators and Beams*, 21(11):112802, 2018.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=FPnUhsQJ5B.

Robert T Glassey. *The Cauchy problem in kinetic theory*. SIAM, 1996.

John Guibas, Morteza Mardani, Zongyi Li, Andrew Tao, Anima Anandkumar, and Bryan Catanzaro. Efficient token mixing for transformers via adaptive fourier neural operators. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=EXHG-A3jlM.

Zhongkai Hao, Zhengyi Wang, Hang Su, Chengyang Ying, Yinpeng Dong, Songming Liu, Ze Cheng, Jian Song, and Jun Zhu. Gnot: A general neural operator transformer for operator learning. In *International conference on machine learning*, pp. 12556–12569. PMLR, 2023.

Zhongkai Hao, Chang Su, Songming Liu, Julius Berner, Chengyang Ying, Hang Su, Anima Anandkumar, Jian Song, and Jun Zhu. DPOT: Auto-regressive denoising operator transformer for large-scale PDE pre-training. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=X7UnDevHOM.

Roger W Hockney and James W Eastwood. *Computer simulation using particles*. CRC Press, 2021.

Benjamin Holzschuh, Georg Kohl, Florian Redinger, and Nils Thuerey. P3D: Highly scalable 3D neural surrogates for physics simulations with global context. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=8UdCE5nhFl.

Jiahe Huang, Guandao Yang, Zichen Wang, and Jeong Joon Park. Diffusionpde: Generative pde-solving under partial observation. *Advances in Neural Information Processing Systems*, 37: 130291–130323, 2024.

Verena Kain, Simon Hirlander, Brennan Goddard, Francesco Maria Velotti, Giovanni Zevi Della Porta, Niky Bruchon, and Gianluca Valentino. Sample-efficient reinforcement learning for cern accelerator control. *Physical Review Accelerators and Beams*, 23(12):124801, 2020.

Jan Kaiser, Chenran Xu, Annika Eichler, Andrea Santamaria Garcia, Oliver Stein, Erik Bründermann, Willi Kuropka, Hannes Dinter, Frank Mayet, Thomas Vinatier, et al. Reinforcement learning-trained optimisers and bayesian optimisation for online particle accelerator tuning. *Scientific reports*, 14(1):15733, 2024.

Siavash Khodakarami, Vivek Oommen, Aniruddha Bora, and George Em Karniadakis. Mitigating spectral bias in neural operators via high-frequency scaling for physical systems. *Neural Networks*, pp. 108027, 2025.

Georg Kohl, Li-Wei Chen, and Nils Thuerey. Benchmarking autoregressive conditional diffusion models for turbulent flow simulation. *Neural Networks*, pp. 108641, 2026.

Nikola Kovachki, Samuel Lanthaler, and Siddhartha Mishra. On universal approximation and error bounds for Fourier neural operators. *Journal of Machine Learning Research*, 22(290):1–76, 2021.

SC Leemann, S Liu, A Hexemer, MA Marcus, CN Melton, H Nishimura, and C Sun. Demonstration of machine learning-based model-independent stabilization of source properties in synchrotron light sources. *Physical review letters*, 123(19):194801, 2019.

Edward Li, Zichen Wang, Jiahe Huang, and Jeong Joon Park. Videopde: Unified generative pde solving via video inpainting diffusion models. *arXiv preprint arXiv:2506.13754*, 2025.

Lizao Li, Robert Carver, Ignacio Lopez-Gomez, Fei Sha, and John Anderson. Generative emulation of weather forecast ensembles with diffusion models. *Science Advances*, 10(13):eadk4489, 2024.

Zijie Li, Kazem Meidani, and Amir Barati Farimani. Transformer for partial differential equations' operator learning. *Transactions on Machine Learning Research*, 2023a. ISSN 2835-8856. URL https://openreview.net/forum?id=EPPqt3uERT.

Zijie Li, Dule Shu, and Amir Barati Farimani. Scalable transformer for pde surrogate modeling. *Advances in Neural Information Processing Systems*, 36:28010–28039, 2023b.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations (ICLR)*, 2021.

Marten Lienen, David Lüdke, Jan Hansen-Palmus, and Stephan Günnemann. From zero to turbulence: Generative modeling for 3D flow simulation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ZhlwoC1XaN.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021.

Morteza Mardani, Noah Brenowitz, Yair Cohen, Jaideep Pathak, Chieh-Yu Chen, Cheng-Chin Liu, Arash Vahdat, Mohammad Amin Nabian, Tao Ge, Akshay Subramaniam, et al. Residual corrective diffusion modeling for km-scale atmospheric downscaling. *Communications Earth & Environment*, 6(1):124, 2025.

Michael McCabe, Bruno Régaldo-Saint Blancard, Liam Holden Parker, Ruben Ohana, Miles Cranmer, Alberto Bietti, Michael Eickenberg, Siavash Golkar, Geraud Krawezik, Francois Lanusse, Mariel Pettee, Tiberiu Tesileanu, Kyunghyun Cho, and Shirley Ho. Multiple physics pretraining for spatiotemporal surrogate models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=DKSI3bULiZ.

Vivek Oommen, Aniruddha Bora, Zhen Zhang, and George Em Karniadakis. Integrating neural operators with diffusion models improves spectral representation in turbulence modelling. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 481(2309): 20240819, 2025. doi: 10.1098/rspa.2024.0819.

Lorenzo Pacchiardi, Rilwan A. Adewoyin, Peter Dueben, and Ritabrata Dutta. Probabilistic forecasting with generative networks via scoring rule minimization. *Journal of Machine Learning Research*, 25(45):1–64, 2024. URL http://jmlr.org/papers/v25/23-0038.html.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, October 2023.

Ilan Price, Alvaro Sanchez-Gonzalez, Ferran Alet, Tom R Andersson, Andrew El-Kadi, Dominic Masters, Timo Ewalds, Jacklynn Stott, Shakir Mohamed, Peter Battaglia, et al. Probabilistic weather forecasting with machine learning. *Nature*, 637(8044):84–90, 2025.

Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks, 2019. URL https://openreview.net/forum?id=r1gR2sC9FX.

Md Ashiqur Rahman, Robert Joseph George, Mogab Elleithy, Daniel Leibovici, Zongyi Li, Boris Bonev, Colin White, Julius Berner, Raymond A. Yeh, Jean Kossaifi, Kamyar Azizzadenesheli, and Anima Anandkumar. Pretraining codomain attention neural operators for solving multiphysics PDEs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=wSpIdUXZYX.

Mahindra Rautela, Alan Williams, and Alexander Scheinker. A conditional latent autoregressive recurrent model for generation and forecasting of beam dynamics in particle accelerators. *Scientific Reports*, 14(1):18157, 2024.

Ryan Roussel, Auralee Edelen, Christopher Mayes, Daniel Ratner, Juan Pablo Gonzalez-Aguilera, Seongyeol Kim, Eric Wisniewski, and John Power. Phase space reconstruction from accelerator beam measurements using neural networks and differentiable simulations. *Physical Review Letters*, 130(14):145001, 2023.

Ryan Roussel, Auralee L Edelen, Tobias Boltz, Dylan Kennedy, Zhe Zhang, Fuhao Ji, Xiaobiao Huang, Daniel Ratner, Andrea Santamaria Garcia, Chenran Xu, et al. Bayesian optimization algorithms for accelerator physics. *Physical review accelerators and beams*, 27(8):084801, 2024a.

Ryan Roussel, Juan Pablo Gonzalez-Aguilera, Eric Wisniewski, Alexander Ody, Wanming Liu, John Power, Young-Kee Kim, and Auralee Edelen. Efficient six-dimensional phase space reconstructions from experimental measurements using generative machine learning. *Physical Review Accelerators and Beams*, 27(9):094601, 2024b.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *International conference on machine learning*, pp. 8459–8468. Pmlr, 2020.

Alexander Scheinker. cDVAE: VAE-guided diffusion for particle accelerator beam 6D phase space projection diagnostics. *Scientific Reports*, 14(1):29303, 2024a.

Alexander Scheinker. Conditional guided generative diffusion for particle accelerator beam diagnostics. *Scientific Reports*, 14(1):19210, 2024b.

Alexander Scheinker. Physics-constrained superresolution diffusion for six-dimensional phase space diagnostics. *Physical Review Research*, 7(2):023091, 2025.

Alexander Scheinker, Auralee Edelen, Dorian Bohler, Claudio Emma, and Alberto Lutman. Demonstration of model-independent control of the longitudinal phase space of electron beams in the linac-coherent light source with femtosecond resolution. *Physical review letters*, 121(4):044801, 2018.

Alexander Scheinker, Dorian Bohler, Sergey Tomin, Raimund Kammering, Igor Zagorodnov, Holger Schlarb, Matthias Scholz, Bolko Beutner, and Winfried Decking. Model-independent tuning for maximizing free electron laser pulse energy. *Physical review accelerators and beams*, 22(8): 082802, 2019.

Alexander Scheinker, Frederick Cropp, Sergio Paiagua, and Daniele Filippetto. An adaptive approach to machine learning for compact particle accelerators. *Scientific reports*, 11(1):19187, 2021.

Alexander Scheinker, Alexander Plastun, and Peter Ostroumov. PhaseFlow4D: Physically constrained 4D beam reconstruction via feedback-guided latent diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 4729–4737, 2026.

Louis Serrano, Thomas X Wang, Etienne Le Naour, Jean-Noël Vittaut, and Patrick Gallinari. AROMA: Preserving spatial structure for latent PDE modeling with local neural fields. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=Aj8RKCGwjE.

Aliaksandra Shysheya, Cristiana Diaconu, Federico Bergamin, Paris Perdikaris, José M Hernández-Lobato, Richard E Turner, and Emile Mathieu. On conditional diffusion models for pde simulations. *Advances in Neural Information Processing Systems*, 37:23246–23300, 2024.

Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020.

H. A. van der Vorst. Bi-cgstab: A fast and smoothly converging variant of bi-cg for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 13(2): 631–644, 1992. doi: 10.1137/0913035. URL https://doi.org/10.1137/0913035.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Jean-Luc Vay, Marco Acciarri, Ann Almgren, Lígia Diana Amorim, Igor Andriyash, Justin Ray Angus, Thierry Antoun, Daniel Belkin, David Bizzozero, Aurore Blelly, Stephen Eric Clark, Eya Dammak, Luca Fedeli, Arianna Formenti, Marco Garten, Lixin Ge, Lorenzo Giacomel, Kevin Gott, Roelof E. Groenewald, David Grote, Junmin Gu, Cyrus Harrison, Muhammad Haseeb, Axel Huebl, Revathi Jambunathan, Ilian Kara-Mostefa, Hannah Klion, Prabhat Kumar, Remi Lehe, Burlen Loring, Thomas Marks, Phil Miller, Andrew Myers, Juliette Pech, Elisa Rheaume, Glenn Richardson, Michael E. Rowan, Ryan Thor Sandberg, Peter Scherpelz, Alexander Sinn, Olga Shapoval, Davide Terzani, Maxence Thévenet, Kale Weichman, Eloise Yang, Neïl Zaim, Weiqun Zhang, Yinjian Zhao, Kevin Z. Zhu, and Edoardo Zoni. WarpX: An advanced Particle-In-Cell code. https://blast-warpx.github.io, 2018. URL https://doi.org/10.5281/zenodo.4571577. https://doi.org/10.5281/zenodo.4571577.

Sifan Wang, Jacob H Seidman, Shyam Sankaran, Hanwen Wang, George J. Pappas, and Paris Perdikaris. CViT: Continuous vision transformer for operator learning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=cRnCcuLvyr.

Tian Wang and Chuang Wang. Latent neural operator for solving forward and inverse pde problems. *Advances in Neural Information Processing Systems*, 37:33085–33107, 2024.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for PDEs on general geometries. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=Ywl6pODXjB.

Tailin Wu, Takashi Maruyama, and Jure Leskovec. Learning to accelerate partial differential equations via latent global evolution. *Advances in Neural Information Processing Systems*, 35:2240–2253, 2022.

Zhi-Qin John Xu, Lulu Zhang, and Wei Cai. On understanding and overcoming spectral biases of deep neural network learning methods for solving pdes. *Journal of Computational Physics*, 530: 113905, 2025.

CONTENTS OF SUPPLEMENTARY MATERIALS

DATASET:

- Appendix A.1: Dataset I: Space-charge-dominated Bunched-beam Transport
- Appendix A.2: Dataset II: Plasma-Wakefield Acceleration (PWFA)

FRAMEWORK DETAILS:

- Appendix B.1: Compressive Autoencoder
- Appendix B.2: Deterministic Latent Neural Operator
- Appendix B.3: Generative Latent Residual
- Appendix B.4: Physics-Constrained Operators

BASELINE DESCRIPTIONS:

- Appendix C.1: Denoising Operator Transformer (DPOT)
- Appendix C.2: Universal Physics Transformer (UPT)
- Appendix C.3: Multiple Physics Pretraining (MPP)
- Appendix C.4: P3D

SUPPLEMENTARY EXPERIMENT RESULTS:

- Appendix D.1: Amplitude-normalization Ablation
- Appendix D.2: Space-charge-dominated Bunched-beam Transport
- Appendix D.3: Plasma Wakefield Acceleration

Computation Resources: All experiments in this work were performed on a server with Nvidia RTX Pro 6000 Blackwell Server Edition 96 GB GPU.

A Dataset Generation

We construct two complementary 3D particle accelerator benchmarks generated with WarpX (Vay et al., 2018). The first one models space-charge-dominated bunched-beam transport in the electrostatic Vlasov-Poisson regime, while the second models plasma-wakefield acceleration (PWFA) in the fully electromagnetic Vlasov-Maxwell regime. Together, these datasets span two distinct accelerator settings that require different physical scales, simulation geometries, control variables, state representations, and field treatments.

A.1 Space-Charge-Dominated Bunched-Beam Transport

The first benchmark is inspired by the low-energy, high-current proton transport conditions at the entrance to the first drift-tube-linac (DTL) tank of the Los Alamos Neutron Science Center (LANSCE). The reference beam energy of $750\,\mathrm{keV}$, RF frequency $201.25\,\mathrm{MHz}$, and $7.5\,\mathrm{mm}$ bore radius follow LANSCE Tank-1 entrance settings. The benchmark is not intended as a cell-by-cell digital twin of the LANSCE lattice. Instead, it retains the essential ingredients of intense linac injection, strong space charge, alternating-gradient transverse focusing, longitudinal RF confinement, and controlled mismatch and centroid perturbations, that drive envelope oscillation, emittance growth, halo formation, and particle loss.

At this energy level, the beam self-field is well described in the electrostatic Vlasov–Poisson regime. We simulate a proton bunch in an idealized periodic focusing channel. In the beam co-moving frame, the longitudinal domain spans one RF wavelength, $L_z \approx 59.5\,\mathrm{mm}$, while the transverse computational box is $16\,\mathrm{mm} \times 16\,\mathrm{mm}$. A grounded cylindrical conductor of radius $7.5\,\mathrm{mm}$ models the accelerator aperture. The longitudinal domain is taken to be periodic, and the applied lattice combines alternating-gradient quadrupole focusing with a periodic RF bucket. The state for model learning is the beam charge density only, from which the electrostatic self-field can be reconstructed directly. The applied external fields are prescribed by machine configuration.

Design Parameters. Each trajectory corresponds to a different beam and accelerator configuration. We sample five physically interpretable dimensionless beam-dynamics quantities and control parameters:

$$\sigma_{0t} \in [60^\circ, 85^\circ], \quad \frac{\sigma_{0l}}{\sigma_{0t}} \in [0.5, 1.2], \quad \eta_t \in [0.4, 0.9], \quad M \in [1.0, 1.4], \quad \frac{x_{\mathrm{off}}}{a} \in [0, 0.15].$$

Here, σ_{0t} denotes the zero-current transverse phase advance per focusing period, $\frac{\sigma_{0l}}{\sigma_{0t}}$ controls the relative longitudinal focusing strength, and η_t is the transverse tune depression and determines the strength of the space-charge interaction. The remaining two describe the initial beam state: M controls the initial transverse mismatch, and $\frac{x_{\mathrm{off}}}{a}$ specifies the normalized horizontal beam offset. The synchronous RF phase is fixed at $\phi_s = -30^\circ$.

For each sampled design, an optics-inversion procedure maps the dimensionless design variables to the physical input parameters required by WarpX, including the quadrupole strength E_{Q0}, RF-field amplitude E_{0T}, total bunch charge q_{tot}, and matched beam dimensions. These quantities play two different roles in the PIC simulation. The accelerator settings $(E_{Q0}, E_{0T}, q_{\mathrm{tot}})$ are deployment-known machine controls and are provided explicitly to the surrogate as control parameters $\mathbf{c}$. In contrast, the mismatch M and centroid offset x_{off} define the initial beam condition and are not supplied as additional control variables. Feasibility and stability checks are applied before simulation so that the retained configurations represent physically meaningful transport conditions rather than failed numerical or optical setups.

Dataset Construction. The main design space is sampled using maximin latin-hypercube points. We supplement these samples with nine representative anchor configurations and two additional high-phase-advance cases with $\sigma_{0t} = 87^\circ, 89^\circ$, which probe dynamics near the envelope-instability regime. PIC simulations contain stochastic fluctuations arising from finite-particle sampling. To quantify this variability, we select 16 feasible physical configurations distributed throughout the

design space and simulate each using four independent seeds. The two high-phase-advance cases are also evaluated using four independent seeds. All replicate simulations corresponding to the same physical configuration are assigned to the same data split to prevent leakage. In total, the dataset contains 420 distinct physically feasible configurations with 474 simulation runs. After simulations, we removed ten simulations where the beam loss is significant ($> 20\%$ of injected charge scraped on the bore wall), resulting in a dataset of 464 simulations used in our experiments. The training, validation, and test sets contain 364, 48, and 52 simulations, respectively.

Numerical Configuration and Data Storage. We simulate the trajectory using approximately 8×10^5 proton macroparticles on a Cartesian grid of size $256 \times 336 \times 336$ ($Z \times Y \times X$). One alternating-gradient focusing period is resolved using 50 PIC time steps. Each simulation is evolved for 4000 steps, corresponding to 80 focusing periods. Mesh states and beam diagnostics are stored every 10 steps, resulting in $T = 401$ frames per trajectory. For the standard trajectories, only charge density ρ is stored in the compressed dataset. For a subset of audit trajectories, we additionally retain scalar potential ϕ and electric fields $\mathbf{E}$, which provides independent field-level ground truth for evaluating the Poisson-based field reconstruction. Physically meaningful beam diagnostics that can be derived directly from the predicted charge density, including the beam centroid, rms beam size, transmitted charge, and aperture loss, are also recorded for each simulation. These quantities can be calculated analytically from ρ, allowing the learned surrogate to be evaluated not only through pointwise field errors but also through accelerator-relevant physical observables. The total computational cost of this dataset is about 900 GPU hours.

A.2 Vlasov-Maxwell Dataset: Plasma-Wakefield Acceleration (PWFA)

The second benchmark models three-dimensional beam-driven plasma-wakefield acceleration (PWFA), an advanced accelerator concept in which the accelerating structure is created dynamically by a plasma rather than conventional RF cavity. An ultra-relativistic electron driver bunch is injected into a preformed plasma and displaces the plasma electrons, exciting a co-moving electromagnetic wake. A trailing witness bunch is placed at an accelerating phase of this wake and gains energy from the longitudinal wakefield. Thus, the accelerating field is generated by the collective plasma response to the injected driver bunch, not by an externally prescribed accelerating cavity. Because the driver, witness, plasma electrons, and electromagnetic fields evolve self-consistently, this benchmark exercises the full Vlasov–Maxwell regime.

The simulation contains four particle species: an electron driver bunch, an electron witness bunch, plasma electrons, and stationary hydrogen ions. The driver and witness are represented by approximately 10^6 and 2.5×10^5 macroparticles, respectively. The background plasma uses two electron macroparticles per cell and one ion macroparticle per cell, with nominal density $n_0 = 10^{23}\mathrm{m}^{-3} = 10^{17}\mathrm{cm}^{-3}$. At the plasma entrance, the density rises smoothly from zero to n_0 over a $2\,\mathrm{mm}$ up-ramp and is then held constant.

Design Parameters. The PWFA campaign is constructed by sampling six dimensionless physical parameters that span distinct wakefield and beam-loading regimes:

$$\Lambda \in [0.05, 1.2], \quad \gamma_d \in [1500, 6000], \quad k_p\sigma_z \in [0.30, 1.19],$$
$$M \in [0.5, 2.5], \quad \frac{x_{\mathrm{off}}}{\sigma_x} \in [0, 0.5], \quad \eta_{\mathrm{load}} \in [0.2, 1.5].$$

Here, Λ controls the normalized driver strength and spans weak excitation through strongly nonlinear wake formation. γ_d sets the driver energy. $k_p\sigma_z$ determines the bunch duration relative to plasma scale. M controls the transverse driver size relative to its matched value. $\frac{x_{\mathrm{off}}}{\sigma_x}$ introduces transverse symmetry breaking, and η_{load} controls witness beam loading.

For each sampled design, these dimensionless quantities are converted into the physical parameters required to initialize the WarpX simulation containing

$$(Q_d,\, \gamma_d,\, \sigma_{z,d},\, \sigma_{r,d},\, x_{d,\mathrm{off}},\, Q_w,\, \Delta z_{dw},\, I_w).$$

Q_d is the driver charge, γ_d is the driver Lorentz factor, $\sigma_{z,d}$ and $\sigma_{r,d}$ are its longitudinal and transverse rms sizes, $x_{d,\mathrm{off}}$ is its transverse centroid offset, Q_w is the witness charge, Δz_{dw} is the driver–witness separation, and I_w indicates whether a witness bunch is present. These physical parameters are provided to the surrogate as control parameters $\mathbf{c}$.

Dataset Construction. The main design space is sampled using maximin latin-hypercube points together with nine representative anchor configurations. To probe additional physically important effects, we supplement the main sweep with several controlled subsets. First, 24 *imperfection twins* are created from selected baseline simulations. These simulations preserve the underlying particle realization while introducing controlled imperfections such as head–tail tilt, transverse emittance asymmetry, correlated energy chirp, and witness offset. Such paired simulations isolate the influence of physical perturbations from ordinary PIC sampling noise. Second, 40 *driver-only twins* are generated by removing the witness bunch from selected configurations while retaining the same driver and plasma setup. These pairs enable controlled comparison of loaded and unloaded plasma wakes. Finally, six hand-selected configurations probe more extreme wake and depletion regimes and are reserved for evaluation. Similar to the Vlasov-Poisson dataset, we additionally select 16 representative configurations and repeat them using four independent particle seeds to estimate the stochastic variability introduced by finite-particle PIC sampling. In total, the PWFA dataset contains 456 distinct physically feasible configurations with 504 simulations. The training set, validation set, and test sets contain 399, 49, and 56 simulations, respectively

Numerical Configuration and Data Storage. We simulate the trajectory using a Cartesian grid of size $336 \times 216 \times 216$ $(Z \times Y \times X)$, covering a transverse region of approximately $129.6\,\mu\mathrm{m} \times 129.6\,\mu\mathrm{m}$ and a longitudinal moving window of approximately $184.8\,\mu\mathrm{m}$. The corresponding spatial resolutions are approximately $\Delta x = \Delta y = 0.6\,\mu\mathrm{m}$, $\Delta z = 0.55\,\mu\mathrm{m}$. Each simulation evolves 36000 PIC steps, corresponding to approximately $19.6\,\mathrm{mm}$ of plasma propagation. Three-dimensional fields are stored every 600 time steps, producing 61 diagnostic frames per trajectory. The initial frame contains a short numerical initialization transient and is retained for auditing but excluded from model training; consequently, frames $1 - 60$ form the learning trajectory. Each trajectory stores ten three-dimensional channels:

$$(\rho_{\mathrm{driver}}, \rho_{\mathrm{witness}}, \rho_{\text{plasma-}e}, \rho_{\text{plasma-}i})\,, \quad (E_x, E_y, E_z)\,, \quad (B_x, B_y, B_z)\,.$$

The species-resolved charge-density channels describe the evolution of the particle populations, while the six electromagnetic-field components characterize the plasma wake. The total computational cost of this dataset is about 600 GPU hours.

B Framework Details

B.1 Compressive autoencoder

We employ a deterministic 3D convolutional autoencoder $(\mathcal{E}_\psi, \mathcal{D}_\psi)$ to map each mesh-resolved high-dimensional PIC state $\mathbf{u}_t$ to a spatial grid latent $\mathbf{z}_t = \mathcal{E}_\psi(\mathbf{u}_t)$ and reconstruct it as $\tilde{\mathbf{u}}_t = \mathcal{D}_\psi(\mathbf{z}_t)$. The encoder and decoder consist of a sequence of residual blocks and resolution-changing layers. Each residual block applies GroupNorm, SiLU activation, and a 3D convolution twice in succession, with the block input added to the output through an identity residual connection. Each downsampling layer uses a strided 3D convolution to jointly reduce the selected spatial dimensions and transform the feature channels. The decoder mirrors this hierarchy in reverse: each upsampling layer first restores the spatial resolution by trilinear interpolation and then applies a 3D convolution to transform the feature channels. A pointwise channel projection maps the final encoder feature map to the grid latent, with a corresponding projection at the beginning of the decoder. No encoder–decoder skip connections bypass the latent bottleneck.

Latent Dimensions. Both datasets use a $16\times$ reduction in total tensor elements. For the space-charge-dominated bunched-beam transport dataset, the encoder receives the single charge density field $\mathbf{u}_t \in \mathbb{R}^{1\times256\times336\times336}$. The 3D Haar transform first maps the single input channel to eight subband channels while halving each spatial dimension,

$$1 \times 256 \times 336 \times 336 \to 8 \times 128 \times 168 \times 168.$$

A learned convolution projects the 8 Haar channels to 32 feature channels. The subsequent encoder stage changes the channel widths as $32 \to 64 \to 128 \to 128$, while the spatial grid evolves as

$$128 \times 168 \times 168 \to 64 \times 84 \times 84 \to 32 \times 42 \times 42 \to 16 \times 42 \times 42.$$

A final projection maps the 128-channel tensor to the 64-channel latent, so that

$$\mathbf{z}_t \in \mathbb{R}^{64\times16\times42\times42}.$$

For the PWFA dataset, the encoder receives 10-channel state $\mathbf{u}_t \in \mathbb{R}^{10\times336\times216\times216}$. The Haar transform first maps each channel to 8 subbands resulting in 80 channels at resolution $168 \times 108 \times 108$. These channels are projected to 64 learned features and passed through the $64 \to 128 \to 256 \to 256$ hierarchy, with spatial resolutions evolving as

$$168 \times 108 \times 108 \to 84 \times 54 \times 54 \to 42 \times 27 \times 27 \to 42 \times 27 \times 27.$$

A final projection maps the 256-channel tensor to the 320-channel latent, producing the latent

$$\mathbf{z}_t \in \mathbb{R}^{320\times42\times27\times27}.$$

3D Haar Transform. Inspired by Dai et al. (2026), we apply a fixed orthonormal 3D Haar transform before the learned encoder. Let

$$h = \frac{1}{\sqrt{2}}[1, 1], \quad g = \frac{1}{\sqrt{2}}[1, -1]$$

denote the one-dimensional low-pass and high-pass Haar filters, respectively. The low-pass filter computes the local average over two neighboring samples, whereas the high-pass filter computes their local difference. Taking their tensor products along the three spatial axes gives the eight separable filters:

$$(hhh), \quad (ghh, hgh,\ hhg), \quad (ggh, ghg, hgg,\ ggg).$$

The hhh component is the three dimensional low-pass subband and represents the local block average. The three subbands containing a single high-pass factor, ghh, hgh, hhg, represent directional variations along the three spatial axes. The remaining four subbands contain high-pass factors along two or three axes and represent mixed or cross-directional variations within the block. Thus, the transform separates local average and multiaxial detail information before learned processing. Defining

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}, \quad M = H \otimes H \otimes H \in \mathbb{R}^{8\times8},$$

where $\otimes$ is the Kronecker product. The Haar transform of each non-overlapping $2 \times 2 \times 2$ block, flattened as $\mathbf{v} \in \mathbb{R}^8$, is computed as $\mathbf{f} = M\mathbf{v}$, where the eight entries of $\mathbf{f}$ are the corresponding Haar subband coefficients. The transform is applied independently to every input channel with stride 2, reshaping the input shape as:

$$C \times Z \times Y \times X \to 8C \times \frac{Z}{2} \times \frac{Y}{2} \times \frac{X}{2}.$$

Because H is orthogonal, the 3D Haar transform is invertible. The decoder predicts the eight subbands for each output channel and applies the fixed inverse Haar transform to recover the original spatial resolution.

Space–channel Residual Shortcuts. Following Chen et al. (2025), each learned down- and upsampling block is paired with a parameter-free shortcut that transfers information across different spatial scales. For a downsampling layer with input $\mathbf{x} \in \mathbb{R}^{C\times Z\times Y\times X}$ and stride $\mathbf{s} = (s_z, s_y, s_x)$. Let $r = s_z s_y s_x$. The shortcut first applies a space-to-channel rearrangement, which folds the samples within each non-overlapping $s_z \times s_y \times s_x$ spatial block into the channel dimension:

$$C \times Z \times Y \times X \longrightarrow rC \times \frac{Z}{s_z} \times \frac{Y}{s_y} \times \frac{X}{s_x}.$$

This operation is pure reshaping and permutation, therefore introducing no interpolation or learned parameters. The resulting rC channels are then resized to the target width C'. Consecutive groups of rC/C' channels are averaged so that the shortcut output has the same shape as the learned strided-convolution branch. The shortcut output is then added to the learned strided-convolution branch.

The upsampling shortcut applies the reverse sequence. For an upsampling factor $\mathbf{s} = (s_z, s_y, s_x)$, the input channels C' are first repeated to obtain rC channels, after which a channel-to-space rearrangement distributes each group of r channels over an $s_z \times s_y \times s_x$ spatial block:

$$rC \times \frac{Z}{s_z} \times \frac{Y}{s_y} \times \frac{X}{s_x} \to C \times Z \times Y \times X.$$

The shortcut output tensor is added to the learned upsampling branch, which performs trilinear interpolation followed by a 3D convolution. Hence, the shortcut provides a parameter-free shape-matched pathway across every resolution change, while the learned branch supplies the residual transformation.

Reconstruction Objectives. The autoencoder is trained to reconstruct each 3D PIC state using a composite objective:

$$\mathcal{L}_{\text{rec}}(\mathbf{u}_t, \tilde{\mathbf{u}}_t) = \mathcal{L}_{\text{rel}}(\mathbf{u}_t, \tilde{\mathbf{u}}_t) + \lambda_{\text{spec}}\mathcal{L}_{\text{spec}}(\mathbf{u}_t, \tilde{\mathbf{u}}_t) + \lambda_{\text{fd}}\mathcal{L}_{\text{fd}}(\mathbf{u}_t, \tilde{\mathbf{u}}_t).$$

The first and dominant term is the relative-L_2 loss,

$$\mathcal{L}_{\text{rel}}(\mathbf{u}_t, \tilde{\mathbf{u}}_t) = \left(\frac{\|\mathbf{u}_t - \tilde{\mathbf{u}}_t\|_2^2}{\max(\|\mathbf{u}_t\|_2^2, \epsilon)} \right)^{\frac{1}{2}},$$

which measures the normalized pixel-wise reconstruction error and provides a scale-normalized measurement of the overall state reconstruction accuracy.

We measure reconstruction accuracy in local spatial variation using a gradient loss

$$\mathcal{L}_{\text{fd}}(\mathbf{u}_t, \tilde{\mathbf{u}}_t) = \sum_{j=1}^{3} \left(\frac{\|\partial_{x_j}\mathbf{u}_t - \partial_{x_j}\tilde{\mathbf{u}}_t\|_2^2}{\max(\|\partial_{x_j}\mathbf{u}_t\|_2^2, \epsilon)} \right)^{\frac{1}{2}}.$$

The spatial derivatives are evaluated using finite differences along the three coordinate directions. Whereas the relative-L_2 term primarily measures agreement in field values, $\mathcal{L}_{\text{fd}}$ directly penalizes errors in their spatial variation, placing additional emphasis on sharp gradients, beam boundaries, and other localized structures that can contribute only weakly to the pointwise reconstruction error.

To account for reconstruction error across spatial frequencies, we additionally use a weighted spectral loss,

$$\mathcal{L}_{\text{spec}}(\tilde{\mathbf{u}}_t, \mathbf{u}_t) = \left(\frac{\|\mathbf{w} \odot \hat{\mathbf{u}}_t(\mathbf{k}) - \mathbf{w} \odot \hat{\tilde{\mathbf{u}}}_t(\mathbf{k})\|_2^2}{\max(\, \|\mathbf{w} \odot \hat{\mathbf{u}}_t(\mathbf{k})\|_2^2, \epsilon_2)} \right)^{\frac{1}{2}},$$

where $\mathbf{w} = (w(\mathbf{k}))_{\mathbf{k}}$ assigns different contributions to different wavenumber ranges designed as

$$w_0(\mathbf{k}) = 1 + \frac{|\mathbf{k}|}{k_{\max}}, \quad w^2(\mathbf{k}) = \begin{cases} 5w_0(\mathbf{k}), & \frac{1}{3}k_{\max} < |\mathbf{k}| < \frac{2}{3}k_{\max} \\ w_0(\mathbf{k}), & \text{otherwise.} \end{cases}$$

This weighting reduces the dominance of low-wavenumber modes, whose large energy contribution is already well captured by the relative-L_2 loss, while placing additional emphasis on intermediate wavenumbers that contain coherent small-scale beam structure. The highest-wavenumber band is not additionally boosted, since it is largely dominated by finite-particle PIC sampling noise and would otherwise encourage the model to overfit realization-specific fluctuations.

Since the decoder receives predicted rather than exact encoded latents during dynamics rollout, we additionally train our decoder to tolerate latent perturbations. Given $\mathbf{z} = \mathcal{E}_\psi(\mathbf{u})$, we construct

$$\mathbf{z} + \sigma \text{std}(\mathbf{z})\boldsymbol{\epsilon}, \quad \sigma \sim \mathcal{U}(0, 0.15), \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),$$

where $\text{std}(\mathbf{z})$ scales the perturbation relative to the magnitude of the latent representation. The autoencoder is then trained to reconstruct the clean target from both the encoded and perturbed latents,

$$\mathcal{L}_{\text{AE}}(\tilde{\mathbf{u}}_t, \mathbf{u}_t) = \mathcal{L}_{\text{rec}}\left(\tilde{\mathbf{u}}_t, \mathbf{u}_t\right) + \lambda_{\text{noise}} \mathcal{L}_{\text{rec}}\big(\mathcal{D}_\psi(\mathbf{z}_t + \sigma \text{std}(\mathbf{z}_t)\boldsymbol{\epsilon}), \mathbf{u}_t\big).$$

This perturbation exposes the decoder to off-center latent states during training, improving robustness to latent prediction errors encountered during autoregressive rollout.

B.2 Deterministic Latent Neural Operator.

Forecast Representation. Given a k-step latent history $\mathbf{z}_{t-k+1:t} \in \mathbb{R}^{k\times C'\times Z'\times Y'\times X'}$ and machine controls $\mathbf{c}$, the deterministic operator $\mathcal{G}_\theta$ predicts the next h latent states, $\tilde{\mathbf{z}}^{\mathrm{det}}_{t+1:t+h} \in \mathbb{R}^{h\times C'\times Z'\times Y'\times X'}$. Let $N = Z'Y'X'$ denote the number of spatial tokens in the compressed latent grid. We first flatten the spatial dimensions of each latent frame,

$$\mathbf{z}_i \in \mathbb{R}^{C'\times Z'\times Y'\times X'} \;\longrightarrow\; \mathbf{Z}_i = \mathrm{Flat}(\mathbf{z}_i) \in \mathbb{R}^{N\times C'}.$$

The k flattened frames are then concatenated along the feature dimension at each spatial location,

$$\mathbf{Z}_{\mathrm{hist}} = [\mathbf{Z}_{t-k+1}, \ldots, \mathbf{Z}_t] \in \mathbb{R}^{N\times kC'}.$$

Thus, the token dimension indexes the N spatial tokens, while the temporal history is encoded in the kC' feature dimension. Finally, a linear lifting layer maps these features to the Transformer hidden dimension d.

Spatial Positional Encoding. Flattening the latent grid removes its explicit three-dimensional arrangement. We therefore associate each token i with its normalized coordinate $\mathbf{x}_i = (x_i, y_i, z_i) \in [-1, 1]^3$ and encode it using sinusoidal Fourier features (Tancik et al., 2020),

$$\boldsymbol{\phi}(\mathbf{x}_i) = [\sin(2\pi\mathbf{B}\mathbf{x}_i), \cos(2\pi\mathbf{B}\mathbf{x}_i)]\,, \qquad \mathbf{e}^{\mathrm{pos}}_i = \mathbf{W}_{\mathrm{pos}}\boldsymbol{\phi}(\mathbf{x}_i),$$

where $\mathbf{B} \in \mathbb{R}^{M\times 3}$ denotes the prescribed frequency bank, and $\mathbf{W} \in \mathbb{R}^{d\times 2M}$ is a learned linear projection that maps the Fourier-feature vector $(\phi)(\mathbf{x}_i)$ into the Transformer hidden dimension d. Stacking $\mathbf{e}^{\mathrm{pos}}_i$ over all N spatial tokens gives $\mathbf{E}_{\mathrm{pos}} \in \mathbb{R}^{N\times d}$, which is added to the lifted latent tokens before the Transformer. This preserves the physical identity of each spatial token while retaining the global interaction enabled by full attention.

Full-Attention Backbone. We process the resulting N spatial tokens using a Transformer with dense self-attention (Vaswani et al., 2017). Full attention allows every latent location to interact directly with all other locations, providing global spatial coupling across the compressed 3D domain. Because attention is applied after spatial compression, this global interaction is tractable despite the high resolution of the original PIC state. After the final block, the spatial token is projected from hidden dimension d directly to dimension hC', which corresponds to the C' channel features over the h-step latent forecast. The flattened spatial token is reshaped back to the original latent grid,

$$\mathbb{R}^{N\times hC'} \;\longrightarrow\; \mathbb{R}^{h\times N\times C'} \;\longrightarrow\; \mathbb{R}^{h\times C'\times Z'\times Y'\times X'},$$

yielding $\tilde{\mathbf{z}}^{\mathrm{det}}_{t+1:t+h}$. The complete h-step forecast is therefore predicted jointly rather than autoregressively within each forecast block.

Control Conditioning with adaLN-Zero. The machine controls $\mathbf{c}$ are first mapped to a conditioning embedding

$$\mathbf{e}_c = \mathrm{MLP}_c(\mathbf{c}) \in \mathbb{R}^d.$$

Following adaLN-Zero (Peebles & Xie, 2023), the control embedding $\mathbf{e}_c$ is injected into each Transformer block through adaptive layer normalization. For each token $\mathbf{h} \in \mathbb{R}^d$, layer normalization standardizes its hidden features,

$$\mathrm{LN}(\mathbf{h}) = \frac{\mathbf{h} - \mu(\mathbf{h})}{\sqrt{\sigma^2(\mathbf{h}) + \epsilon}},$$

where $\mu(\mathbf{h})$ and $\sigma^2(\mathbf{h})$ are the mean and variance over the d hidden dimensions. A modulation network maps $\mathbf{e}_c$ to channel-wise scale and shift vectors $\boldsymbol{\gamma}, \boldsymbol{\beta} \in \mathbb{R}^d$, giving

$$\mathrm{adaLN}(\mathbf{h}; \mathbf{e}_c) = \big(1 + \boldsymbol{\gamma}(\mathbf{e}_c)\big) \odot \mathrm{LN}(\mathbf{h}) + \boldsymbol{\beta}(\mathbf{e}_c).$$

Thus, $\boldsymbol{\gamma}$ rescales and $\boldsymbol{\beta}$ shifts each hidden feature according to the machine controls.

AdaLN-Zero predicts separate $(\boldsymbol{\gamma}, \boldsymbol{\beta}, \boldsymbol{\alpha})$ for the attention and feed-forward branches, where $\boldsymbol{\alpha} \in \mathbb{R}^d$ gates the corresponding residual update:

$$\mathbf{H}' = \mathbf{H} + \boldsymbol{\alpha}_{\mathrm{attn}}(\mathbf{e}_c) \odot \mathrm{Attn}(\mathrm{adaLN}_{\mathrm{attn}}(\mathbf{H}; \mathbf{e}_c))\,,$$
$$\mathbf{H}^+ = \mathbf{H}' + \boldsymbol{\alpha}_{\mathrm{ffn}}(\mathbf{e}_c) \odot \mathrm{FFN}(\mathrm{adaLN}_{\mathrm{ffn}}(\mathbf{H}'; \mathbf{e}_c))\,.$$

The modulation network therefore produces six d-dimensional vectors per block: scale, shift, and gate parameters for both attention and feed-forward branches. The gates $\boldsymbol{\alpha}$ are initialized to zero, so each block initially behaves as an identity mapping and gradually learns control-dependent updates.

B.3 GENERATIVE LATENT RESIDUAL.

Amplitude-Normalization. The latent residual $\mathbf{r}_{t+1:t+h} = \mathbf{z}_{t+1:t+h} - \tilde{\mathbf{z}}^{\mathrm{det}}_{t+1:t+h} \in \mathbb{R}^{h\times C'\times Z'\times Y'\times X'}$ inherits the strong amplitude heterogeneity of beam dynamics across latent channels, forecast blocks, and spatial regions. Training flow matching directly on these raw residuals biases the objective toward large-amplitude components, making weaker structures difficult to learn consistently. We instead normalize the residual to have approximately unit RMS before generative modeling, while predicting a separate conditional amplitude factor that restores its original scale at inference.

A fixed dataset-level normalization is insufficient because the residual magnitude varies with the current beam state and deterministic forecast, while the true future residual is unavailable at deployment. We therefore predict the scale from

$$\mathbf{p} = \left(\mathbf{z}_{t-k+1:t}, \tilde{\mathbf{z}}^{\mathrm{det}}_{t+1:t+h}\right),$$

which is available at inference, and factor it across three granularities:

$$\mathbf{s}_\omega(\mathbf{p}) = \boldsymbol{\sigma}_r \odot s_{\mathrm{win}}(\mathbf{p}) \odot s_{\mathrm{tok}}(\mathbf{p}), \qquad \bar{\mathbf{r}}_{t+1:t+h} = \mathbf{r}_{t+1:t+h} \oslash \mathbf{s}_\omega(\mathbf{p}).$$

Here, $\boldsymbol{\sigma}_r \in \mathbb{R}^{1\times C'\times 1\times 1\times 1}$ is a global per-channel RMS, $s_{\mathrm{win}}(\mathbf{p}) \in \mathbb{R}_+$ is a conditional normalizer for the whole forecast block, and $s_{\mathrm{tok}}(\mathbf{p}) \in \mathbb{R}^{1\times 1\times Z'\times Y'\times X'}$ is a conditional spatial normalizer shared across channels and forecast steps.

The design follows the RMS normalization principle. For residual values $\{q_i\}_{i=1}^N$ with empirical mean square

$$m = \frac{1}{N}\sum_{i=1}^{N} q_i^2,$$

normalizing by $s = \sqrt{m}$ gives unit RMS,

$$\frac{1}{N}\sum_{i=1}^{N}\left(\frac{q_i}{s}\right)^2 = 1.$$

Because this target RMS depends on the unknown future residual, we instead predict $s(\mathbf{p})$ from the available conditioning information. Using the zero-centered Gaussian scale objective

$$\mathcal{L}_{\mathrm{scale}}\big(s(\mathbf{p})\big) = \frac{m}{s(\mathbf{p})^2} + 2\log s(\mathbf{p}),$$

which is used only for scale estimation and does not assume that the normalized residual itself is Gaussian. For a fixed conditioning context $\mathbf{p}$, the expected objective is

$$\mathbb{E}\big[\mathcal{L}_{\mathrm{scale}}\big(s(\mathbf{p}); m\big) \mid \mathbf{p}\big] = \frac{\mathbb{E}[m \mid \mathbf{p}]}{s(\mathbf{p})^2} + 2\log s(\mathbf{p}).$$

Minimizing this objective with respect to $s(\mathbf{p})$ gives

$$s^\star(\mathbf{p})^2 = \mathbb{E}[m \mid \mathbf{p}], \qquad s^\star(\mathbf{p}) = \sqrt{\mathbb{E}[m \mid \mathbf{p}]}.$$

Thus, when optimized over the training distribution, the scale predictor estimates the conditional RMS of the residual at the corresponding granularity.

We estimate the factors sequentially. First, the global channel scale $\boldsymbol{\sigma}_r$ is computed from the training residuals and used to form $\mathbf{r}_{\mathrm{ch}} = \mathbf{r} \oslash \boldsymbol{\sigma}_r$. For each h-step forecast block, we define

$$m_{\mathrm{win}} = \frac{1}{hC'Z'Y'X'}\sum_{j,c,z,y,x} r^2_{\mathrm{ch},j,c,z,y,x},$$

and train a network conditioned on $\mathbf{p}$ to predict $s_{\mathrm{win}}(\mathbf{p})$ optimized with $\mathcal{L}_{\mathrm{scale}}$. After freezing this network, we form $\mathbf{r}_{\mathrm{win}} = \mathbf{r}_{\mathrm{ch}}/s_{\mathrm{win}}(\mathbf{p})$. The remaining spatial variation is modeled token-wise using

$$m_{\mathrm{tok}}(z,y,x) = \frac{1}{hC'}\sum_{j,c} r^2_{\mathrm{win},j,c,z,y,x},$$

from which a second network predicts $s_{\mathrm{tok}}(\mathbf{p})$ with the same scale objective. The resulting factorized normalizer $\mathbf{s}_\omega = \boldsymbol{\sigma}_r \odot s_{\mathrm{win}} \odot s_{\mathrm{tok}}$ places residual components on comparable scales for flow matching, while the same predicted factors restore their amplitude at inference.

Flow-matching Training. After freezing the residual normalizer, the generative model $\mathcal{R}_\xi$ learns the conditional distribution of normalized residual $\bar{\mathbf{r}}_{t+1:t+h}$ using conditional flow matching. For each training example, we sample $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\eta \sim \mathcal{N}(0, 1)$, set $\tau = \mathrm{sigmoid}(\eta)$, and construct the linear interpolant

$$\mathbf{x}_\tau = (1 - \tau)\boldsymbol{\epsilon} + \tau \bar{\mathbf{r}}_{t+1:t+h}.$$

The velocity network $\mathbf{v}_\xi$ takes $\mathbf{x}_\tau$, the latent history $\mathbf{z}_{t-k+1:t}$ and deterministic forecast $\tilde{\mathbf{z}}^{\mathrm{det}}_{t+1:t+h}$, controls $\mathbf{c}$, and flow time τ as input, and predicts transport velocity along the linear path. The model is trained with

$$\mathcal{L}_{\mathcal{R}_\xi} = \mathbb{E}_{\tau,\boldsymbol{\epsilon},\bar{\mathbf{r}}} \|\mathbf{v}_\xi(\mathbf{x}_\tau; \mathbf{z}_{t-k+1:t}, \tilde{\mathbf{z}}_{t+1:t+h}, \mathbf{c}, \tau) - (\bar{\mathbf{r}} - \boldsymbol{\epsilon})\|_2^2 + \lambda_{\mathrm{ES}} \mathcal{L}_{\mathrm{ES}}.$$

The $\mathcal{L}_{\mathrm{ES}}$ is a two-sample energy-score regularizer on generated residuals. Given two independent noise samples $\boldsymbol{\epsilon}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, through one Euler step from $\tau = 0$,

$$\tilde{\mathbf{r}}^{(i)} = \boldsymbol{\epsilon}^{(i)} + \mathbf{v}_\xi(\boldsymbol{\epsilon}^{(i)}; \mathbf{z}_{t-k+1:t}, \tilde{\mathbf{z}}_{t+1:t+h}, \mathbf{c}, 0), \quad i = 1, 2$$

The energy-score term is calculated as

$$\mathcal{L}_{\mathrm{ES}} = \left[\frac{\|\tilde{\mathbf{r}}^{(1)} - \bar{\mathbf{r}}\|_2 + \|\tilde{\mathbf{r}}^{(2)} - \bar{\mathbf{r}}\|_2}{2} - \frac{\|\tilde{\mathbf{r}}^{(1)} - \tilde{\mathbf{r}}^{(2)}\|_2}{2} \right].$$

The energy-score term complements the pointwise flow-matching loss by encouraging generated residuals to remain consistent with the target while discouraging an under-dispersed conditional ensemble.

At inference, we sample $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and integrate

$$\frac{\mathrm{d}\mathbf{x}_\tau}{\mathrm{d}\tau} = \mathbf{v}_\xi(\mathbf{x}_\tau; \mathbf{z}_{t-k+1:t}, \tilde{\mathbf{z}}_{t+1:t+h}, \mathbf{c}, \tau),$$

from $\tau = 0$ to 1 by 16-step Euler integration. The terminal state $\mathbf{x}_1$ is the normalized sample, which is rescaled as $\tilde{\mathbf{r}}_{t+1:t+h}$ and added to the latent forecast.

Backbone Architecture. Both the deterministic operator and generative flow matching employ the transformer backbone, as visualized in Figure 5. The flow time τ is injected to each transformer block by AdaLN-Zero, same as the control parameters $\mathbf{c}$. For GLOBE-C, the deterministic neural operator $\mathcal{G}_\theta$ has 12 Transformer blocks with hidden dimension $d = 512$. The generative residual model has 10 Transformer blocks with hidden dimension $d = 384$. For the GLOBE-G ablation variant, we keep the hidden dimension and number of Transformer blocks the same as in G_θ. All Transformer blocks use eight attention heads and an MLP ratio of 4.

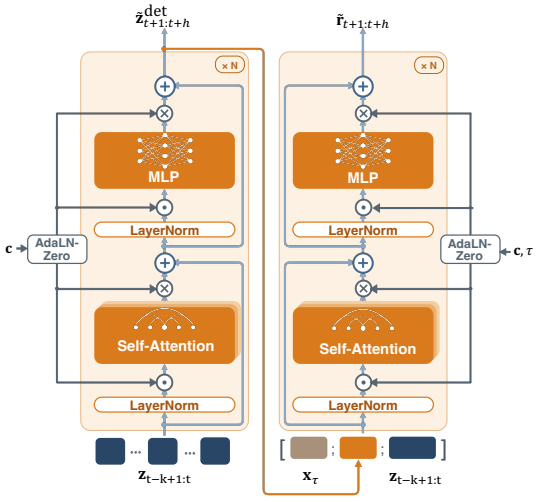

Figure 5: Visualization of the Transformer backbone used by deterministic neural operator $\mathcal{G}_\theta$ and generative residual model $\mathcal{R}_\xi$.

B.4 PHYSICS-CONSTRAINED OPERATOR

Vlasov–Poisson Regime. Given the decoded charge density $\tilde{\rho}$, the predicted electrostatic potential $\tilde{\phi}$ is obtained by solving the discrete Poisson equation

$$-\Delta_h \tilde{\phi} = \frac{\tilde{\rho}}{\varepsilon_0} \quad \text{in } \Omega, \qquad \tilde{\phi}|_{\Gamma_a} = 0, \qquad \tilde{\phi}(x, y, z + L_z) = \tilde{\phi}(x, y, z),$$

where Γ_a denotes the grounded cylindrical beam-pipe boundary and L_z is one RF wavelength in the beam co-moving frame. Longitudinal periodicity is incorporated directly into the discrete Poisson operator by wrapping the finite-difference stencil across the two z faces. The grounded Dirichlet condition fixes the potential reference, so the potential is uniquely determined by $\tilde{\rho}$. In implementation, we use an iterative Krylov solver (Jacobi-preconditioned BiCGStab (van der Vorst, 1992)) to solve $\tilde{\phi}$, with numerical tolerance 10^{-8}.

The electric self-field is subsequently reconstructed on the same collocated grid using the discrete gradient,

$$\tilde{\mathbf{E}} = -\nabla_h \tilde{\phi}.$$

Away from the embedded boundary, ∇_h uses second-order centered differences, for example,

$$(D_j^h \phi)_{\mathbf{n}} = \frac{\phi_{\mathbf{n}+\mathbf{e}_j} - \phi_{\mathbf{n}-\mathbf{e}_j}}{2\Delta_j}, \quad j \in \{x, y, z\},$$

where $\mathbf{e}_j$ denotes one grid step along direction j. Periodic wrapping is used in the longitudinal direction, while boundary-adjacent transverse differences are evaluated consistently with the grounded embedded-boundary discretization.

Vlasov–Maxwell Regime. Unlike the electrostatic regime, in the Vlasov–Maxwell system, the charge density and electromagnetic fields carry independent dynamical information and are therefore predicted jointly. The instantaneous Maxwell constraints are formulated as

$$\nabla_h^E \cdot \mathbf{E} - \frac{\rho}{\varepsilon_0} = 0, \qquad \nabla_h^B \cdot \mathbf{B} = 0. \tag{6}$$

Both discrete divergence operators, ∇_h^E and ∇_h^B, are composed of translation-invariant finite-difference and averaging stencils and therefore diagonalized under the discrete Fourier transform.

Let

$$k_j = \frac{2\pi n_j}{L_j}, \qquad n_j = 0, \cdots, N_j - 1, \qquad j \in \{x, y, z\},$$

represent the discrete Fourier wavenumber along direction j, where $L_j = N_j \Delta_j$. The electric-field divergence uses the wide centered difference

$$(D_j^h f)_{\mathbf{n}} = \frac{f_{\mathbf{n}+\mathbf{e}_j} - f_{\mathbf{n}-\mathbf{e}_j}}{2\Delta_j},$$

whose Fourier symbol is

$$d_j^E(\mathbf{k}) = \frac{i \sin(k_j \Delta_j)}{\Delta_j}.$$

The stored magnetic field follows the centering convention of the CKC/Yee field solver, so its matched discrete divergence has a different symbol. Defining the forward-difference and averaging symbols

$$d_j^+(\mathbf{k}) = \frac{e^{ik_j\Delta_j} - 1}{\Delta_j}, \qquad a_j(\mathbf{k}) = \frac{1 + e^{ik_j\Delta_j}}{2},$$

the magnetic-divergence symbol along component j is

$$d_j^B(\mathbf{k}) = d_j^+(\mathbf{k}) a_{t_1}(\mathbf{k}) a_{t_2}(\mathbf{k}) T_j(\mathbf{k}),$$

where t_1, t_2 denote the two directions transverse to j, and T_j is the fixed CKC transverse-smoothing symbol,

$$T_j = 1 - 2\beta\Big[r_{t_1}\big(1 - \cos(k_{t_1}\Delta_{t_1})\big) + r_{t_2}\big(1 - \cos(k_{t_2}\Delta_{t_2})\big)\Big] - 4\gamma_j\big(1 - \cos(k_{t_1}\Delta_{t_1})\cos(k_{t_2}\Delta_{t_2})\big)$$

where r_{t_j}, β, γ_j are fixed by CKC stencil and the grid spacings. Thus, ∇_j^E and ∇_j^B deliberately use different symbols because they must match the discrete representation of the stored electric and magnetic fields.

For each Fourier mode, define

$$\hat{\mathbf{u}} = \begin{bmatrix} \hat{\rho} & \hat{E}_x & \hat{E}_y & \hat{E}_z & \hat{B}_x & \hat{B}_y & \hat{B}_z \end{bmatrix}^\top$$

Equation 6 then becomes the homogeneous linear system

$$\mathbf{A}_h(\mathbf{k})\hat{\mathbf{u}}(\mathbf{k}) = 0,$$

where

$$\mathbf{A}_h(\mathbf{k}) = \begin{bmatrix} -\frac{1}{\varepsilon_0} & d_x^E & d_y^E & d_z^E & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & d_x^B & d_y^B & d_z^B \end{bmatrix}.$$

$\mathbf{A}_h$ is the Fourier-space matrix representation of Gauss's law and magnetic divergence constraint under the matched finite difference operators.

Independently at each Fourier mode $\mathbf{k}$, given an unconstrained decoder prediction $\hat{\hat{\mathbf{u}}}(\mathbf{k})$, we seek the smallest weighted correction that satisfies Equation 6,

$$\hat{\mathbf{u}}^\star(\mathbf{k}) = \underset{\hat{\mathbf{u}}(\mathbf{k}):\mathbf{A}_h(\mathbf{k})\hat{\mathbf{u}}(\mathbf{k})=0}{\arg\min} \|\hat{\mathbf{u}}(\mathbf{k}) - \hat{\hat{\mathbf{u}}}(\mathbf{k})\|^2_{\mathbf{W}^{-1}},$$

where $\mathbf{W}$ is a diagonal matrix whose diagonal entries contain the per-channel scales used by the reconstruction metrics. The closed-form solution is

$$\hat{\mathbf{u}}^\star(\mathbf{k}) = \hat{\hat{\mathbf{u}}}(\mathbf{k}) - \mathbf{W}\mathbf{A}_h^{\mathsf{H}}(\mathbf{k})\Big(\mathbf{A}_h(\mathbf{k})\mathbf{W}\mathbf{A}_h^{\mathsf{H}}(\mathbf{k})\Big)^\dagger \mathbf{A}_h(\mathbf{k})\hat{\hat{\mathbf{u}}}.$$

where $\cdot^{\mathsf{H}}$ and $\cdot^\dagger$ denote the Hermitian transpose and Moore–Penrose pseudoinverse. After applying this projection independently to all Fourier modes, the corrected coefficients are assembled and mapped back to the computational grid using the inverse discrete Fourier transform.

The spectral projection is implemented using a periodic 3D FFT of the matched discrete difference stencils. Periodicity is used only to diagonalize the operators in Fourier space. The stored simulation window itself is non-periodic. Directly removing the full periodic Gauss residual would couple opposite faces and impose an artificial periodic boundary condition. We therefore compute

$$D_G = \nabla_h^E \cdot \tilde{\mathbf{E}} - \tilde{\rho}/\varepsilon_0$$

in real space and form the masked residual

$$D_G^{\mathrm{proj}} = mD_G,$$

where $m = 0$ within two cells of each face and $m = 1$ in the interior. The Fourier transform of D_G^{proj}, rather than the full $\mathbf{A}_h\hat{\tilde{\mathbf{u}}}$, is supplied to the Gauss row of the minimum-norm correction above. Consequently, the corrected state satisfies Gauss's law exactly in the interior, while the boundary shell remains unconstrained.

Let Ω_{int} denote the unmasked region. By the divergence theorem,

$$\Phi_E = \oint_{\partial\Omega_{\mathrm{int}}} \mathbf{E}\cdot\mathbf{n}\, dA = \int_{\Omega_{\mathrm{int}}} \nabla\cdot\mathbf{E}\, dV,$$

so Gauss's law requires

$$\varepsilon_0\Phi_E = Q_{\mathrm{int}}, \qquad Q_{\mathrm{int}} = \int_{\Omega_{\mathrm{int}}} \rho\, dV.$$

Thus, a nonzero enclosed charge requires a corresponding electric flux through the boundary of the interior region. A fully periodic projection cannot represent this relation because the net flux over a periodic box is identically zero. Leaving the boundary shell unconstrained allows the required flux to pass through the interior boundary while enforcing Gauss's law exactly inside.

This also clarifies the $\mathbf{k} = 0$ mode. For nonzero mode, $\mathbf{E}$ and ρ are corrected simultaneously with respect to Gauss's law in the interior. However, for $\mathbf{k} = 0$, $d_j^E(\mathbf{0}) = 0$, the electric field cannot change the zero-frequency component of the Gauss residual. The zero mode of the masked residual

therefore measures the global mismatch between the electric flux through $\partial\Omega_{\mathrm{int}}$ and the enclosed net charge,

$$D_G(\mathbf{0}) \propto \Phi_E - \frac{Q_{\mathrm{int}}}{\varepsilon_0}.$$

Because ρ is included jointly with $\mathbf{E}$ in $\mathbf{A}_h$, the Gauss row remains nonsingular at $\mathbf{k} = 0$: at this mode the minimum-norm correction acts through the charge-density component, corresponding to a spatially uniform shift of ρ. The corrected state therefore satisfies

$$Q^{\star}_{\mathrm{int}} = \varepsilon_0 \Phi_E,$$

without forcing the simulation window to be charge neutral. At nonzero modes, ρ and $\mathbf{E}$ are corrected jointly according to the same projection defined above.

C Baseline Descriptions

C.1 Denoising Operator Transformer (DPOT)

DPOT (Hao et al., 2024) is a Transformer-based PDE foundation model designed for autoregressive prediction across heterogeneous physical systems. Its architecture first patchifies the input fields and projects each spatial patch into a hidden representation. Information from multiple input time steps is combined through a temporal aggregation module, followed by a stack of Fourier attention blocks based on the Adaptive Fourier Neural Operator (AFNO). A lightweight output projection maps the processed patch representations back to the predicted physical fields. DPOT is trained using a relative-L_2 objective. In our experiments, we use the DPOT architecture at a matched model size and adapt its input and output channels to the corresponding beam–field states. In both benchmarks, DPOT is trained to predict h-step future states given k-step histories.

C.2 Universal Physics Transformer (UPT)

UPT (Alkin et al., 2024) is a latent neural operator consisting of an encoder, a latent dynamics approximator, and a query-based decoder. The encoder compresses the input physical state into a fixed set of latent tokens using local aggregation and Perceiver-style cross-attention. A Transformer approximator then propagates these tokens forward in latent space. To recover the physical field, the decoder embeds the coordinates of the desired output locations as queries and cross-attends to the propagated latent tokens, allowing the latent representation to be evaluated at arbitrary spatial positions. UPT is trained with a next-step prediction loss together with inverse decoding and inverse encoding losses, all implemented with mean-square error, to promote consistency of the latent representation during rollout. In our setting, the full 3D mesh locations are used as decoder queries to recover the beam–field state during inference. During training, due to the design of the model architecture, UPT is trained to predict the next single future state given the past k-step histories. For training tractability, we uniformly select 150528 (bunched-beam transport) and 163296 (PWFA) query points among the spatial grids for loss function evaluation on both datasets.

C.3 Multiple Physics Pretraining (MPP)

MPP (McCabe et al., 2024) is an autoregressive Transformer framework designed to learn dynamics shared across heterogeneous physical systems. It first normalizes and projects different physical fields into a common embedding space, followed by spatial patchification. Its backbone is an Axial Vision Transformer (AViT), which factorizes self-attention across the temporal and individual spatial axes to reduce the cost of processing high-dimensional spatiotemporal fields. The processed tokens are projected back to the physical variables to predict the next state. MPP is trained with a normalized mean-square error. For our comparison, we instantiate the MPP backbone at a model size comparable to GLOBE and adapt the field embedding and output layers to each benchmark. Since MPP's output head decodes a single time slot into one frame, training predicts only the next state given the k-step histories.

C.4 P3D

P3D (Holzschuh et al., 2026) is a scalable backbone developed specifically for high-resolution 3D physics simulations. It combines convolutional blocks for local feature extraction with windowed self-attention in a hierarchical U-shaped encoder–decoder architecture, enabling direct processing of full-resolution 3D states while limiting the cost of attention. We evaluate the same backbone under two training formulations.

P3D-D. For the deterministic baseline, P3D directly maps the input state and conditioning variables to the future physical state and is trained with the supervised mean-square error proposed in the original work. Predictions are then rolled out autoregressively in the full 3D state space.

P3D-G. For the generative baseline, we follow the probabilistic formulation of P3D and train the same 3D backbone with conditional flow matching. Starting from Gaussian noise, P3D-G learns a velocity field transporting the noise distribution toward the conditional distribution of the target physical state. At inference, the corresponding ODE is integrated iteratively to generate each fore-

cast. Unlike GLOBE, which performs flow integration on a compressed latent grid, P3D-G applies the generative backbone directly to the full 3D state.

Table 5: Ablation of the amplitude normalization for the generative residual. GLOBE-C uses the factorized normalizer $s_\omega = \boldsymbol{\sigma}_r \odot s_{\mathrm{win}} \odot s_{\mathrm{tok}}$; the variant keeps only the global per-channel scale $\boldsymbol{\sigma}_r$. GLOBE-D (deterministic anchor, no residual) is shown for reference.

	Bunched-Beam Transport		**PWFA**	
Method	Rel-$L_2 \downarrow$	$L_{\mathcal{F}}$ (low / mid / high) $\downarrow$	Rel-$L_2 \downarrow$	$L_{\mathcal{F}}$ (low / mid / high) $\downarrow$
GLOBE-D (no residual)	**0.144**	0.039 / 0.994 / 1.003	**0.174**	0.083 / 0.350 / **0.592**
GLOBE-C ($\boldsymbol{\sigma}_r$ only)	0.147	0.039 / 0.894 / 0.986	0.266	0.231 / 4.32 / 6.66
GLOBE-C (Ours)	0.161	**0.035 / 0.457 / 0.973**	0.194	**0.077 / 0.300** / 0.618

D Supplementary Experiment Results

D.1 Amplitude-normalization Ablation

Table 5 evaluates the amplitude normalization introduced in Section 4.2 by replacing the factorized normalizer $s_\omega = \boldsymbol{\sigma}_r \odot s_{\mathrm{win}} \odot s_{\mathrm{tok}}$ with the global per-channel scale $\boldsymbol{\sigma}_r$ alone. On bunched-beam transport, the $\boldsymbol{\sigma}_r$-only variant increases the mid-band spectral error from 0.457 to 0.894, approaching the deterministic GLOBE-D baseline (0.994). Without window- and token-level scaling, the flow-matching objective is dominated by larger-amplitude residuals, while weaker fine-scale variations, particularly in the low-density regions near the bunch boundary, receive insufficient weight. Consequently, the residual model provides only limited refinement over the deterministic prediction.

The effect is more pronounced on PWFA, where residual magnitudes vary by orders of magnitude across field channels, forecast windows, and spatial locations. Using only $\boldsymbol{\sigma}_r$ increases Rel-L_2 from 0.194 to 0.266 and the low/mid/high spectral errors from $0.077/0.300/0.618$ to $0.231/4.32/6.66$. The factorized normalizer compensates for these heterogeneous amplitudes, bringing residual targets to comparable, approximately unit-RMS scales across channels, windows, and tokens. This substantially improves the conditioning of the flow-matching objective and enables stable and effective generative refinement on both benchmarks.

D.2 Space-charge-dominated Bunched-Beam Transport

We further visualize representative bunched-beam trajectories from the test set spanning different charges, focusing strengths, initial mismatches, and space-charge regimes. For each trajectory, we report three complementary views. The transverse projection tracks the beam cross section together with the second-order moments σ_x, σ_y, σ_z, and the aspect ratio σ_x/σ_y. The longitudinal projection visualizes the coupled transverse–longitudinal evolution. Finally, an x–z slice at $y = 0$ examines fine-scale spatial structure. For the latter, we additionally show the mid-band component obtained by a 2-D Fourier band-pass retaining $0.168 < |\mathbf{k}| \leq 0.246$ cycles/cell, normalized by the same peak density $\rho_{\max}$ and displayed on a symmetric scale. This isolates the spatial scales at which deterministic regression tends to smooth the PIC realization and highlights the structure recovered by generative residual refinement.

Run 392: mismatched case. Run 392, visualized in Figures 6, 7, and 8 is the representative trajectory with relatively weak space charge but a 20% initial envelope mismatch. The mismatch produces sustained transverse envelope oscillations, with σ_x increasing by approximately 42% over the rollout while σ_z contracts by about 5%. GLOBE accurately follows the alternating transverse deformation and long-term envelope evolution. The deterministic prediction captures the bulk beam shape but smooths much of the realization-dependent mid-band structure. Generative residual refinement restores this texture while preserving the predicted envelope dynamics.

Run 244: low-charge, strongly focused case. Run 244, visualized in Figures 9, 10, and 11 has the lowest bunch charge and strongest focusing among the selected cases, together with the largest initial transverse offset. The beam remains well confined while undergoing a pronounced exchange of transverse aspect ratio, with σ_x/σ_y varying from approximately 2.3 to 0.5. GLOBE closely tracks this repeated x–y deformation and the longitudinal beam size. The spatial comparisons show that the deterministic forecast recovers the smooth envelope but largely suppresses the mid-band

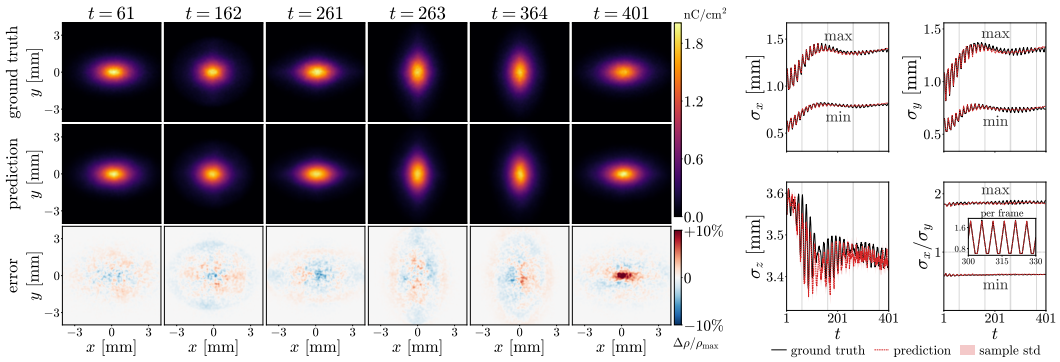

Figure 6: Bunched-beam Transport Run 392: transverse beam evolution. Ground-truth and predicted density projections are shown together with the evolution of σ_x, σ_y, σ_z, and σ_x/σ_y.

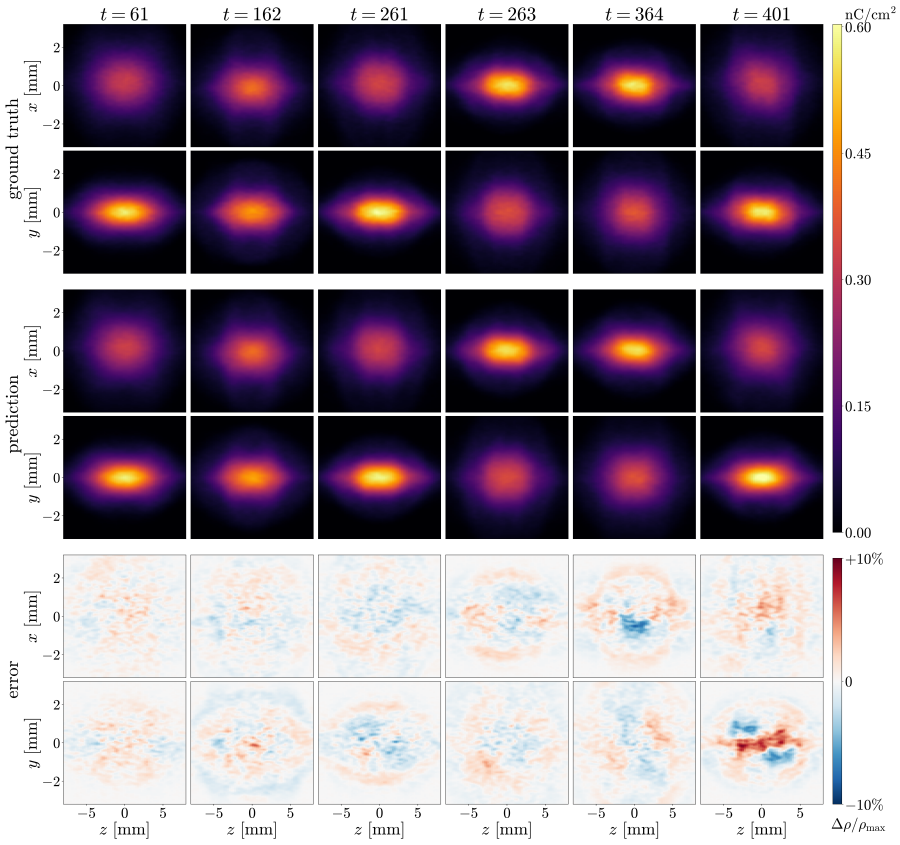

Figure 7: Bunched-beam Transport Run 392: longitudinal beam evolution. Ground-truth and predicted longitudinal projections are compared over the rollout, with relative density errors shown below.

PIC texture, whereas the generative residual restores fluctuations with spatial extent and amplitude similar to the reference.

Run 389: near-matched intermediate case. Run 389, visualized in Figures 12, 13, and 14 is the representative trajectory discussed in the main text, operating with intermediate space charge and a nearly matched initial beam. Its envelope changes only moderately, with roughly 5% transverse growth over the full trajectory. GLOBE reproduces both the recurring transverse shape changes and the nearly stationary longitudinal extent. Although the deterministic prediction already captures the macroscopic evolution accurately, its density field is visibly smoother than the WarpX realization. The generative residual recovers much of the missing mid-band texture without altering the underlying envelope dynamics.

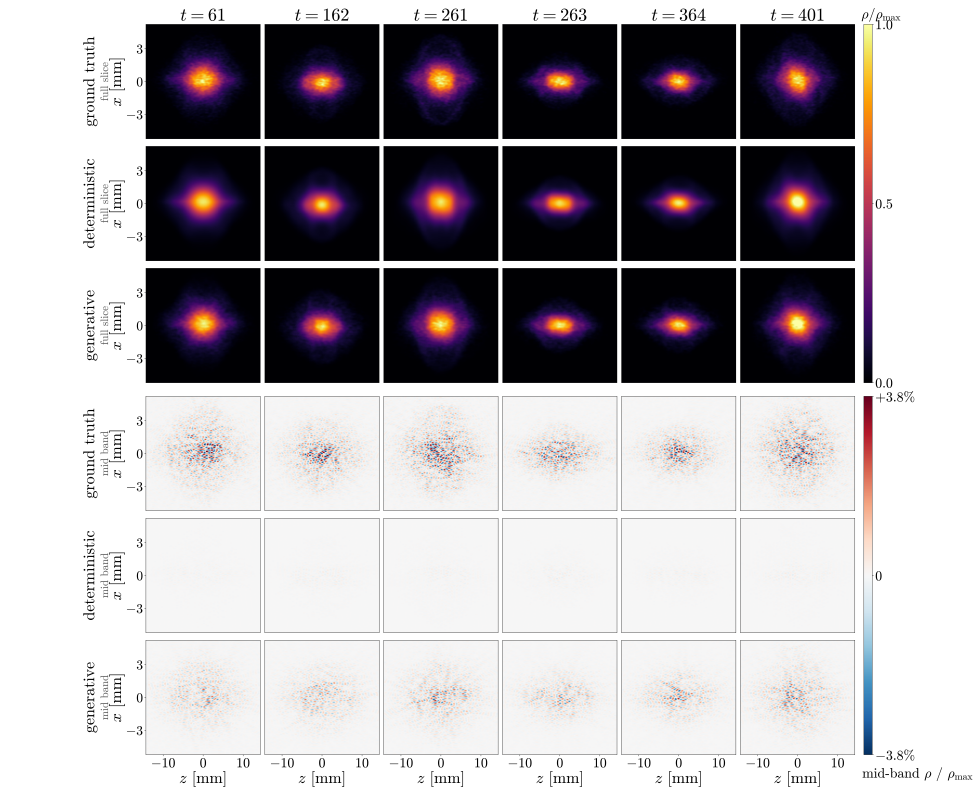

Figure 8: Bunched-beam Transport Run 392: fine-scale structure in the $y = 0$ longitudinal slice. The deterministic prediction captures the bulk density evolution, while generative refinement recovers the realization-dependent mid-band structure observed in WarpX.

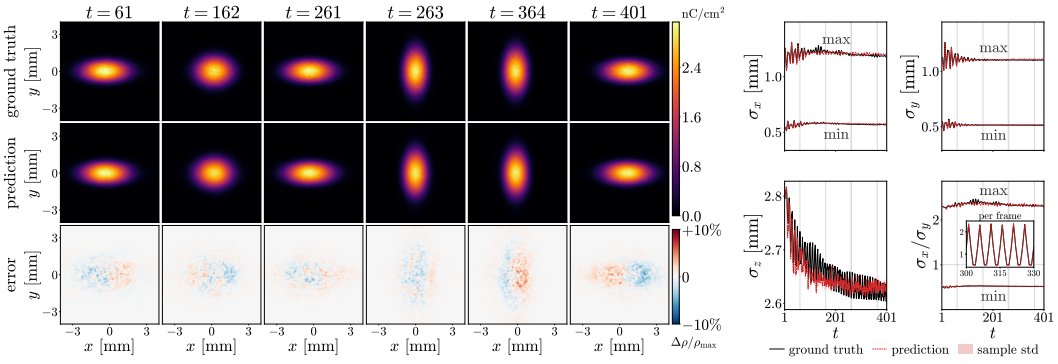

Figure 9: Bunched-beam Transport Run 244: transverse beam evolution. Ground-truth and predicted density projections are shown together with the evolution of σ_x, σ_y, σ_z, and σ_x/σ_y.

Run 87: high-charge, longitudinally extended case. Run 87, visualized in Figures 15, 16, and 17, has the largest bunch charge and weakest RF confinement among the selected trajectories, producing the longest bunch ($\sigma_z \approx 4.5\,\mathrm{mm}$) and stronger space-charge effects. Despite this more challenging operating point, GLOBE tracks the transverse oscillations and longitudinal extent accurately. The projected densities remain accurate throughout the rollout. As in the lower-charge cases, generative refinement restores realization-dependent mid-band structure that is largely absent from the deterministic prediction.

Run 343: strong space-charge case. Run 343, visualized in Figures 18, 19, and 20 is the most challenging selected trajectory. Strong space charge drives approximately 35% transverse envelope growth together with 8.5% longitudinal compression, producing substantial redistribution among the beam dimensions. GLOBE captures the overall expansion, compression, and alternating transverse

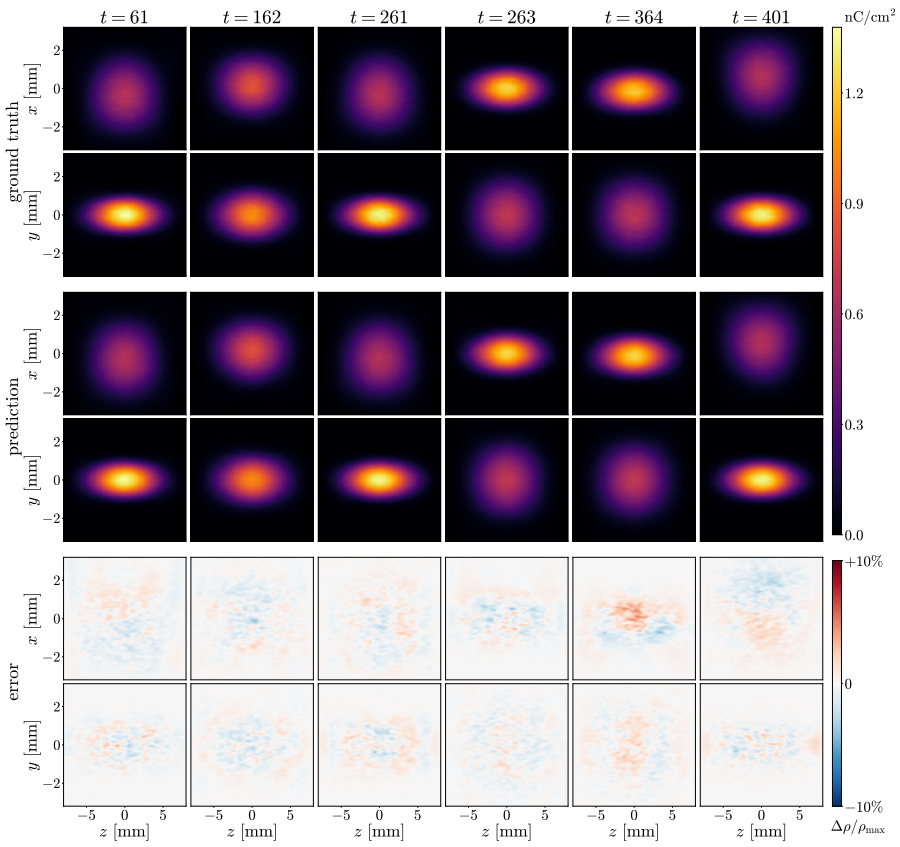

Figure 10: Bunched-beam Transport Run 244: longitudinal beam evolution. Ground-truth and predicted longitudinal projections are compared over the rollout, with relative density errors shown below.

geometry, although deviations accumulate at later times. The density errors are consequently larger and more spatially organized than in the preceding cases. Nevertheless, the generative residual continues to recover a substantial fraction of the mid-band spatial structure that is suppressed by the deterministic forecast, showing that the refinement remains effective even in the strongest space-charge regime considered.

D.3 PWFA Visualization

We also present additional PWFA test trajectories to complement the main-text example. For each case, we provide three complementary visual diagnostics. First, particle-and-wake snapshots together with trajectory-level observables evaluate the evolution of the accelerating field and bubble geometry. Second, at a representative intermediate frame, we compare the ground truth, prediction, and error for E_z, E_x, cB_y, and the derived transverse wakefield $W_x = E_x - cB_y$. While E_z determines longitudinal acceleration, W_x is the effective transverse field acting on an ultra-relativistic witness and therefore directly probes whether the model captures the focusing structure of the wake. Finally, channel-wise relative-error maps at representative times show how prediction errors evolve across the dominant field and density variables. Together, these figures illustrate how GLOBE captures wake formation, beam loading, transverse focusing, and bubble evolution across regimes ranging from quasi-linear operation to strongly nonlinear blowout.

Run 312 (main-text case). Run 312, visualized in Figures 21, 22, and 23, is the main-text reference case, with a moderately nonlinear blowout wake, near-optimal beam loading ($\eta_{\text{load}} = 1.03$), and a slightly offset, relatively wide driver. The particle-and-wake rollout shows close agreement in the evolving bubble geometry and witness location. $E_{z,\text{wit}}$ follows the WarpX trajectory after the initial transient, while r_{bubble} remains particularly accurate. At $t = 25$, GLOBE reproduces the large-scale structure of E_z, E_x, and cB_y, as well as their derived combination $W_x = E_x - cB_y$, indicating that both the accelerating field and transverse focusing structure are captured. The channel-

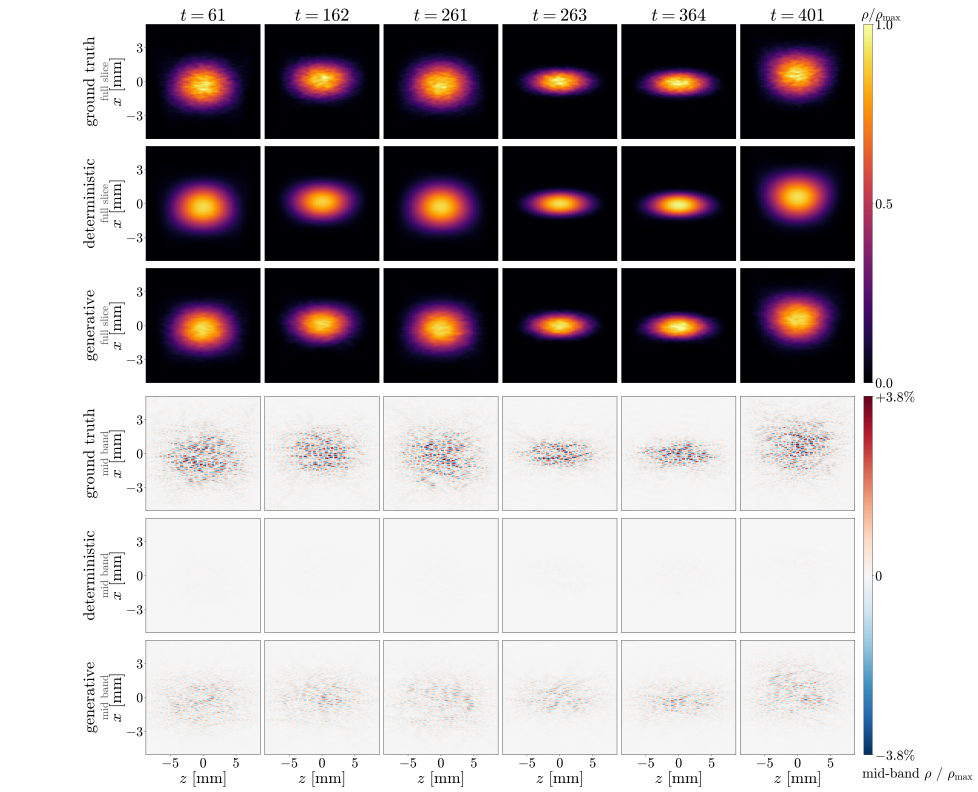

Figure 11: Bunched-beam Transport Run 244: fine-scale structure in the $y = 0$ longitudinal slice. The deterministic prediction captures the bulk density evolution, while generative refinement recovers the realization-dependent mid-band structure observed in WarpX.

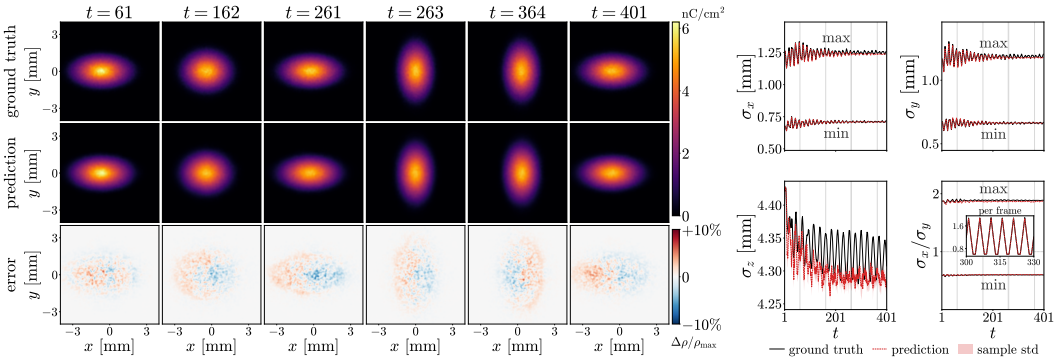

Figure 12: Bunched-beam Transport Run 389: transverse beam evolution. Ground-truth and predicted density projections are shown together with the evolution of σ_x, σ_y, σ_z, and σ_x/σ_y.

wise error maps show that discrepancies remain concentrated around the driver, witness, and thin plasma-sheath structures rather than the bulk wake, even at late times. Over the rollout, the whole-field Rel-L_2 errors are $0.129/0.132/0.144$ for the deterministic, ensemble-mean, and sampled predictions.

Run 355: overloaded, strongly depleted driver case. Run 355, visualized in Figures 24, 25, and 26, combines an overloaded witness ($\eta_{\text{load}} = 1.23$) with the lowest driver energy among the selected cases, producing the strongest driver depletion (about 14%). The macroparticle-and-wake evolution shows that GLOBE captures the loaded cavity, witness acceleration, and progressive driver evolution, with only modest drift in $E_{z,\text{wit}}$ and little error in the bubble radius. At $t = 25$, the predicted E_z, E_x, and cB_y retain the correct phase and spatial organization, while $W_x = E_x - cB_y$ confirms that the transverse focusing structure is preserved despite the stronger beam loading. The

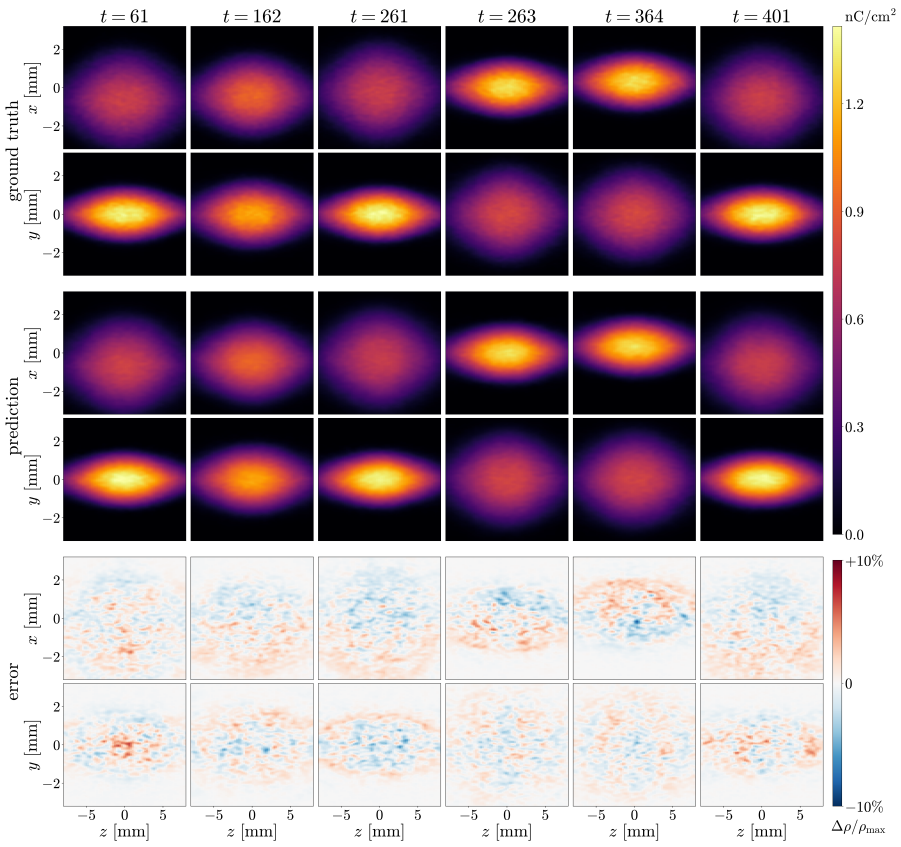

Figure 13: Bunched-beam Transport Run 389: longitudinal beam evolution. Ground-truth and predicted longitudinal projections are compared over the rollout, with relative density errors shown below.

channel-wise error maps show that the largest discrepancies develop near the rear of the cavity and around the witness, where the loading most strongly reshapes the wake.

Run 271: high-charge, underloaded axisymmetric wake case. Run 271, visualized in Figures 27, 28, and 29, has the largest normalized driver charge ($\Lambda = 0.40$), weak transverse offset, and an underloaded witness ($\eta_{\text{load}} = 0.72$), producing the strongest peak field among the selected cases. The macroparticle-and-wake rollout shows a clean, nearly axisymmetric blowout, with GLOBE closely reproducing the driver, witness, plasma sheath, and bubble evolution throughout propagation. At $t = 25$, the model accurately captures the strong accelerating E_z field and the opposing E_x and cB_y contributions; their combination W_x preserves the transverse focusing structure experienced by the witness. The relative-error maps remain localized mainly near the driver and rear sheath, while the dominant cavity structure is maintained through the rollout.

Run 27: high-energy, lightly loaded wake case. Run 27, visualized in Figures 30, 31, and 32, uses the highest driver energy in the selected set together with a narrower-than-matched driver and relatively light witness loading ($\eta_{\text{load}} = 0.86$). The macroparticle-and-wake sequence shows stable propagation of the bubble and witness, with the predicted $E_{z,\text{wit}}$ following the reference trend and the bubble radius remaining nearly indistinguishable from WarpX. At $t = 25$, GLOBE reproduces the longitudinal accelerating structure in E_z as well as the transverse E_x and cB_y fields; their combination W_x remains close to the reference, indicating that the focusing force is reproduced in addition to the accelerating field. The multi-time error maps show that the remaining discrepancies are confined primarily to the narrow driver/witness regions and fine plasma-sheath structure.

Run 118: quasi-linear wake case. Run 118, visualized in Figures 33, 34, and 35, lies in the quasi-linear regime, with a much smaller normalized driver charge ($\Lambda = 0.07$), a ~ 4 GV/m-class wake, and the largest relative transverse offset among the selected cases. The macroparticle-and-wake rollout shows that GLOBE follows the weak plasma response and witness evolution through the full

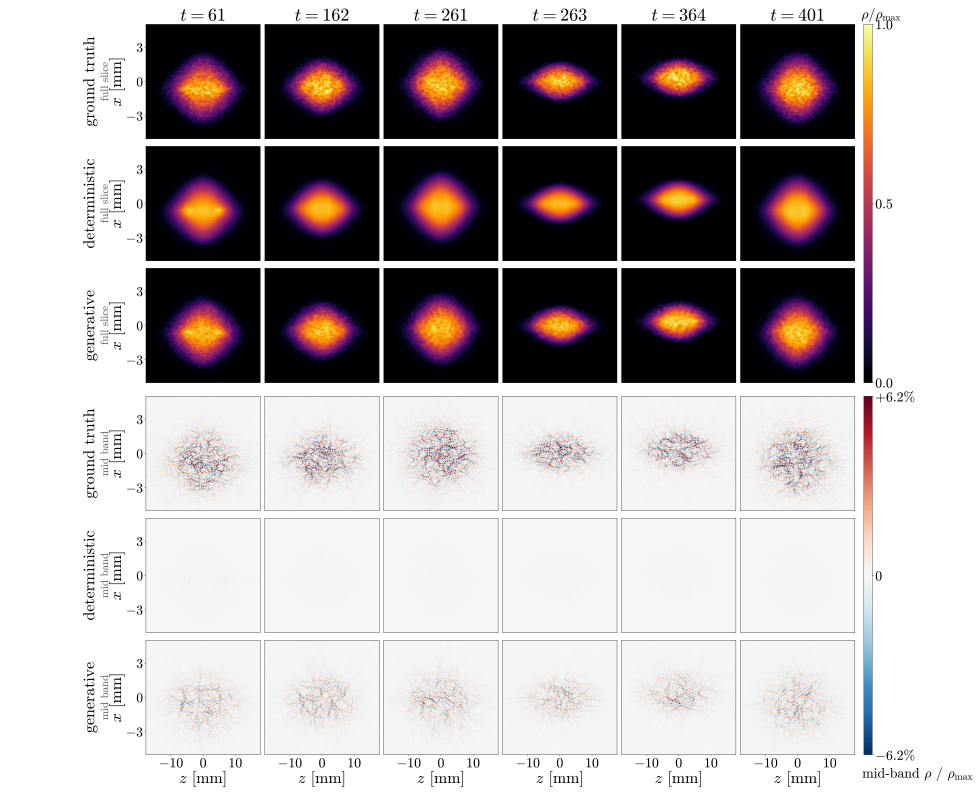

Figure 14: Bunched-beam Transport Run 389: fine-scale structure in the $y = 0$ longitudinal slice. The deterministic prediction captures the bulk density evolution, while generative refinement recovers the realization-dependent mid-band structure observed in WarpX.

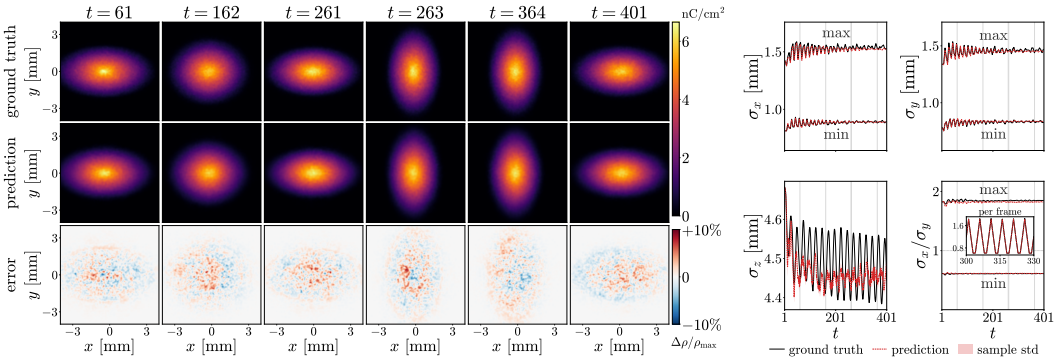

Figure 15: Bunched-beam Transport Run 87: transverse beam evolution. Ground-truth and predicted density projections are shown together with the evolution of σ_x, σ_y, σ_z, and σ_x/σ_y.

horizon despite the absence of a well-developed blowout cavity. At $t = 25$, the model reproduces the smoother E_z, E_x, and cB_y patterns and their derived transverse field W_x, demonstrating that the same model captures both acceleration and focusing outside the bubble regime. The multi-time error maps show substantially weaker field and density structures than in the nonlinear cases, with the predicted perturbations remaining aligned with the reference as the wake propagates. The witness carries only 7 pC and lies at the diagnostic floor, so its transverse-size error is not reported.

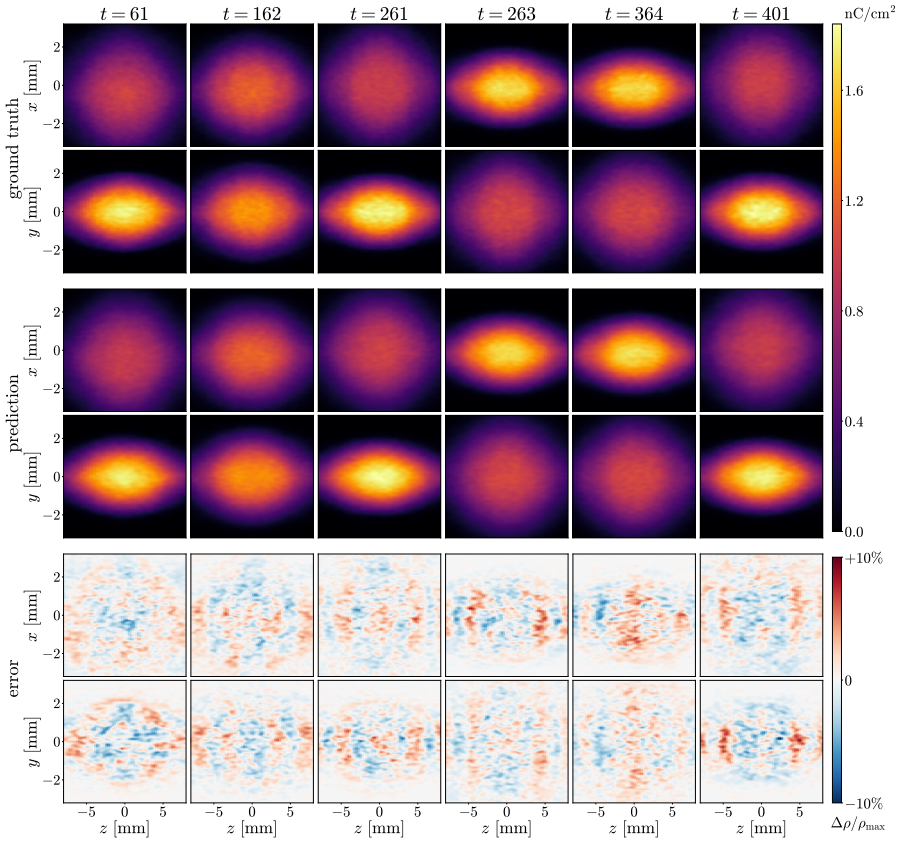

Figure 16: Bunched-beam Transport Run 87: longitudinal beam evolution. Ground-truth and predicted longitudinal projections are compared over the rollout, with relative density errors shown below.

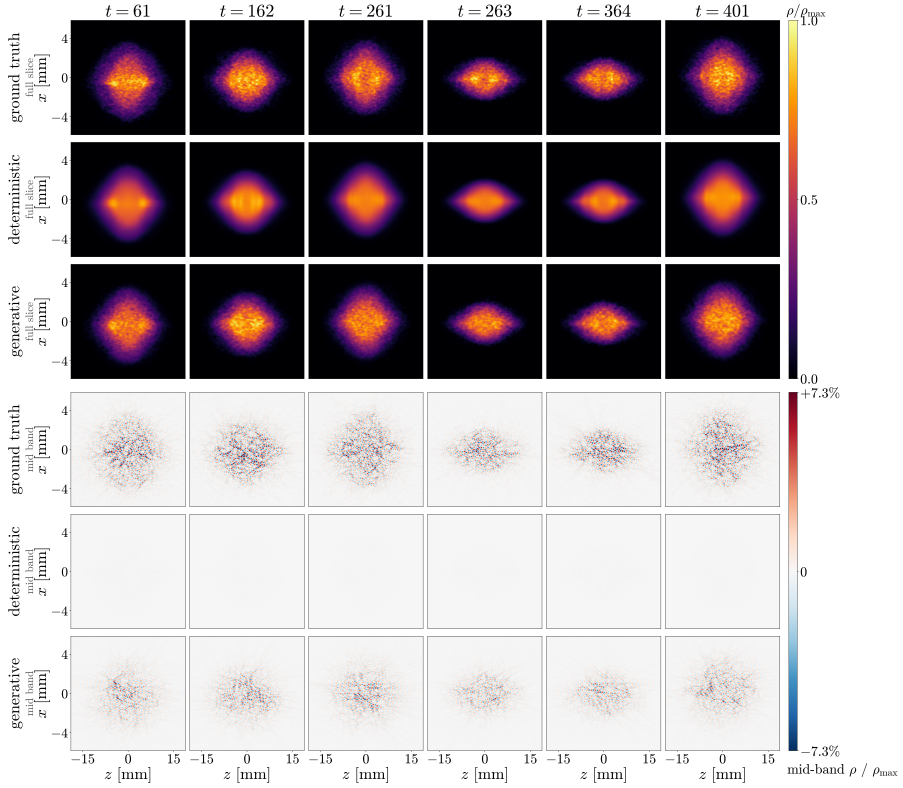

Figure 17: Bunched-beam Transport Run 87: fine-scale structure in the $y = 0$ longitudinal slice. The deterministic prediction captures the bulk density evolution, while generative refinement recovers the realization-dependent mid-band structure observed in WarpX.

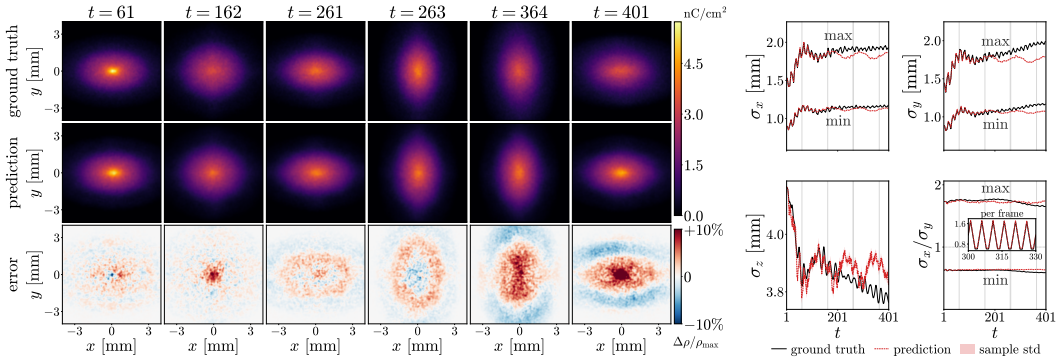

Figure 18: Bunched-beam Transport Run 343: transverse beam evolution. Ground-truth and predicted density projections are shown together with the evolution of σ_x, σ_y, σ_z, and σ_x/σ_y.

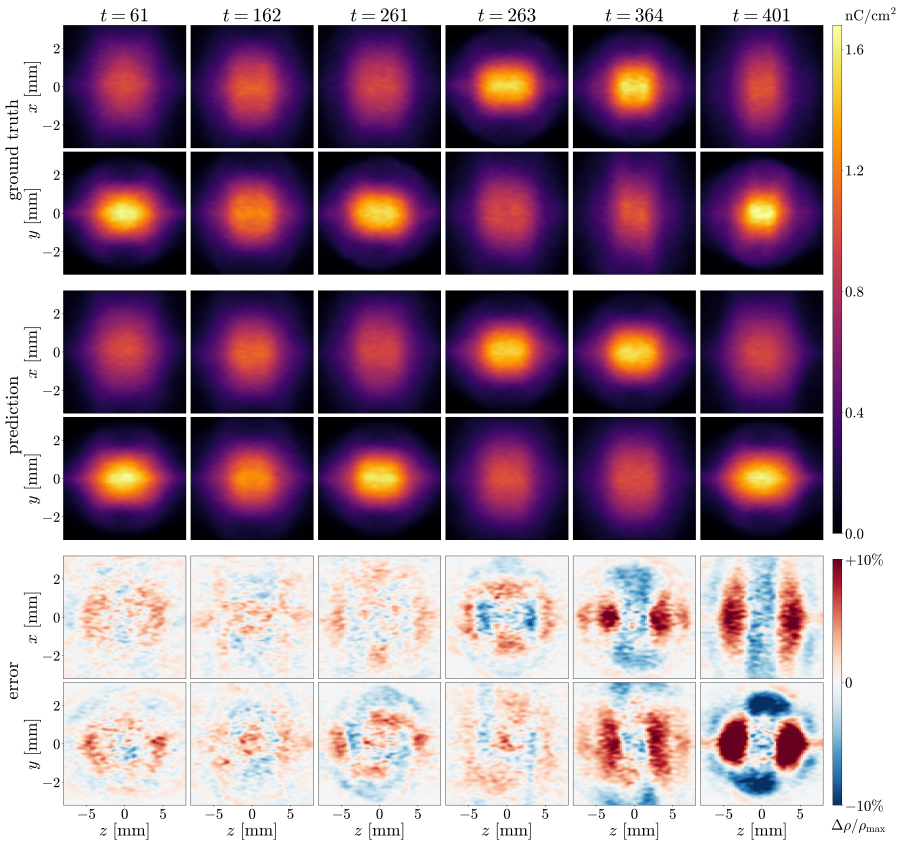

Figure 19: Bunched-beam Transport Run 343: longitudinal beam evolution. Ground-truth and predicted longitudinal projections are compared over the rollout, with relative density errors shown below.

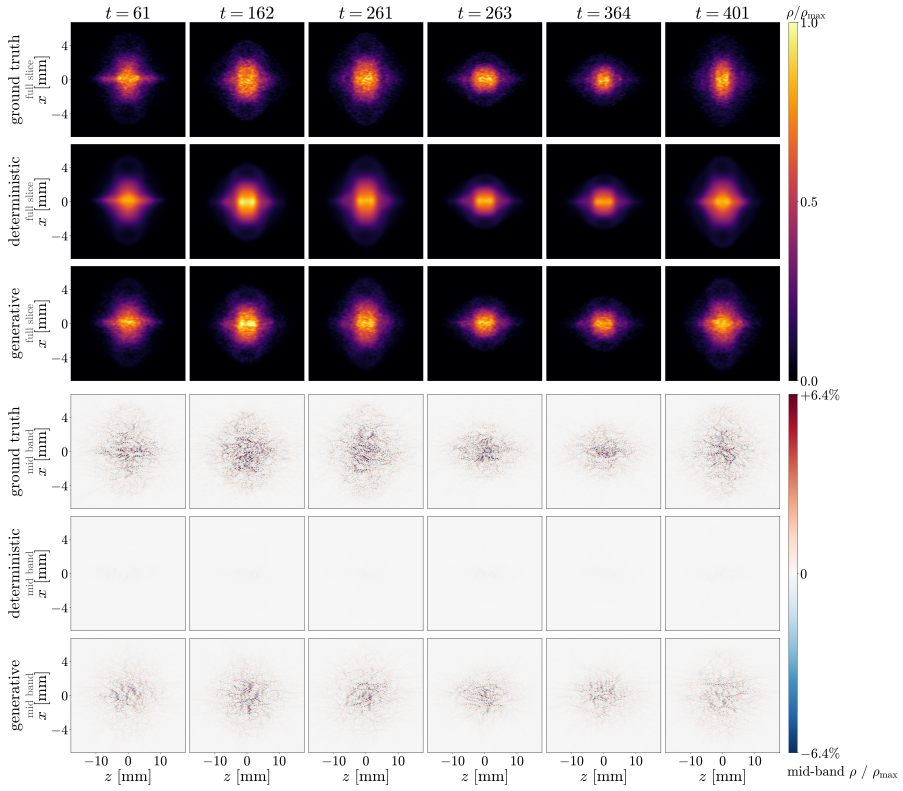

Figure 20: Bunched-beam Transport Run 343: fine-scale structure in the $y = 0$ longitudinal slice. The deterministic prediction captures the bulk density evolution, while generative refinement recovers the realization-dependent mid-band structure observed in WarpX.

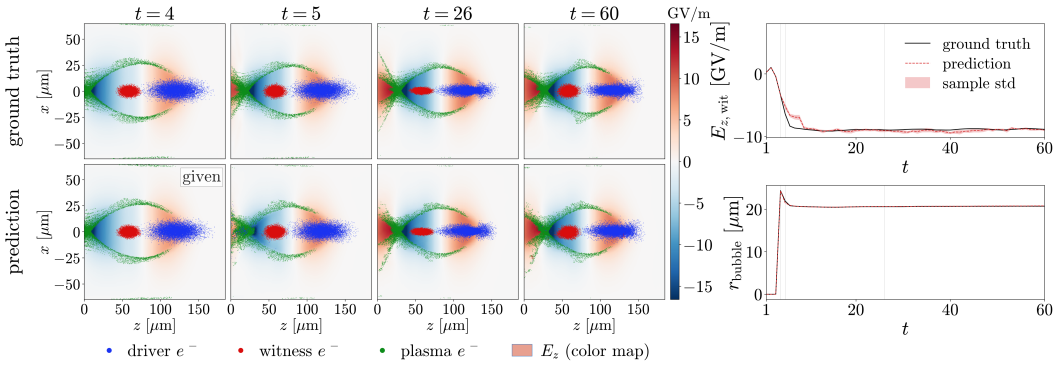

Figure 21: PWFA Run 312: wake evolution in the x–z plane together with trajectory-level observables. The color map shows E_z; particles are overlaid by species. Right: predicted and reference observables with the ensemble sample deviation.

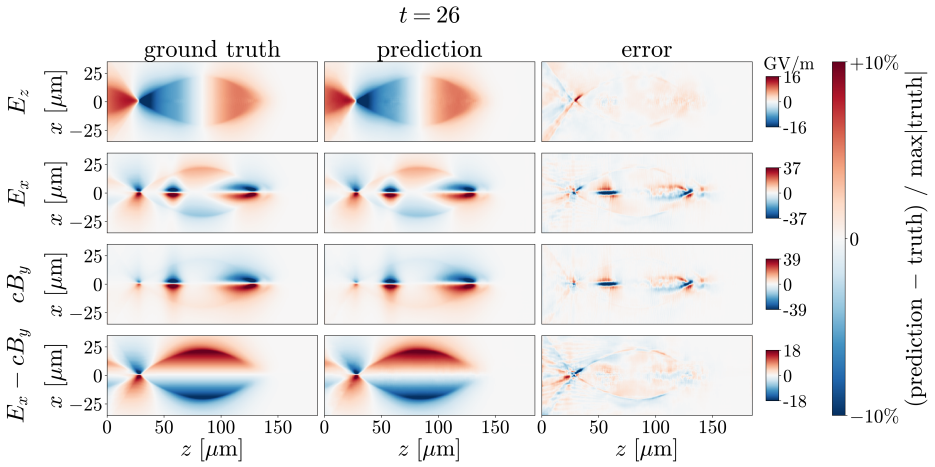

Figure 22: PWFA Run 312: detailed comparison at $t = 25$ for representative field quantities. Left to right: ground truth, prediction, and error.

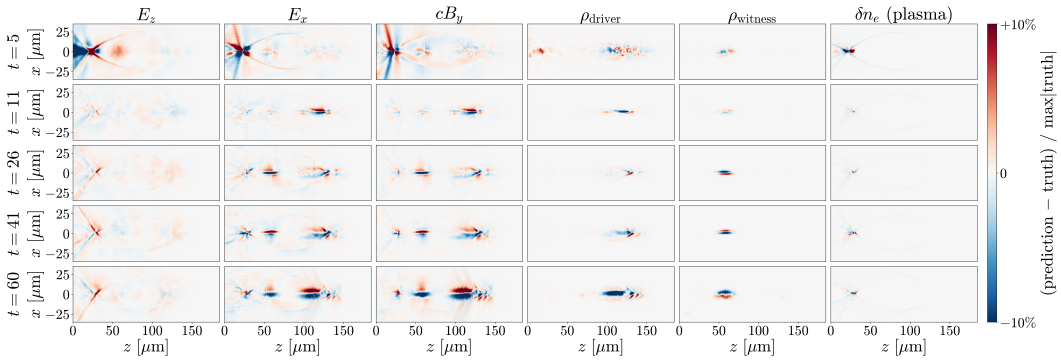

Figure 23: PWFA Run 312: relative-error maps for representative channels at selected times. Colors show $(\text{prediction} - \text{truth})/\max|\text{truth}|$.

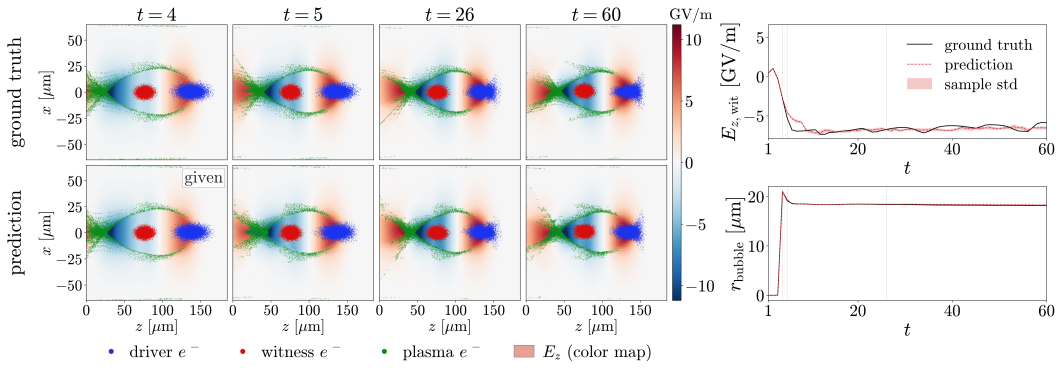

Figure 24: PWFA Run 355: wake evolution in the x–z plane together with trajectory-level observables. The color map shows E_z; particles are overlaid by species. Right: predicted and reference observables with the ensemble sample deviation.

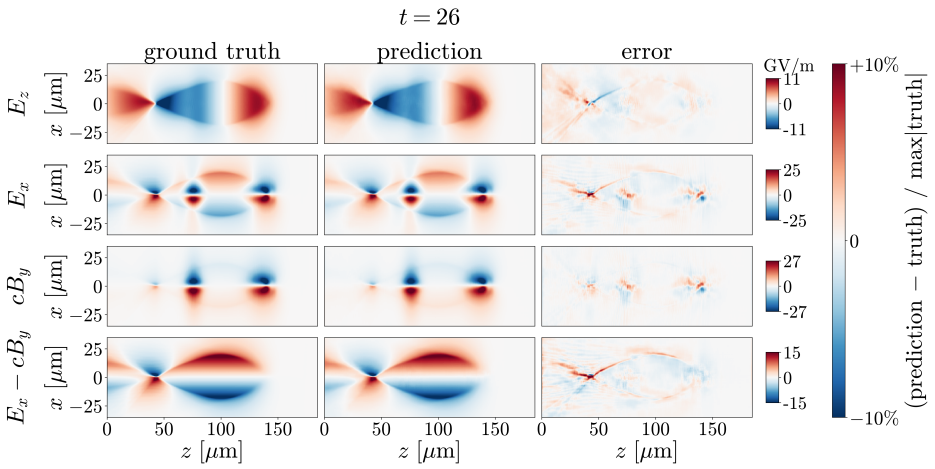

Figure 25: PWFA Run 355: detailed comparison at $t = 25$ for representative field quantities. Left to right: ground truth, prediction, and error.

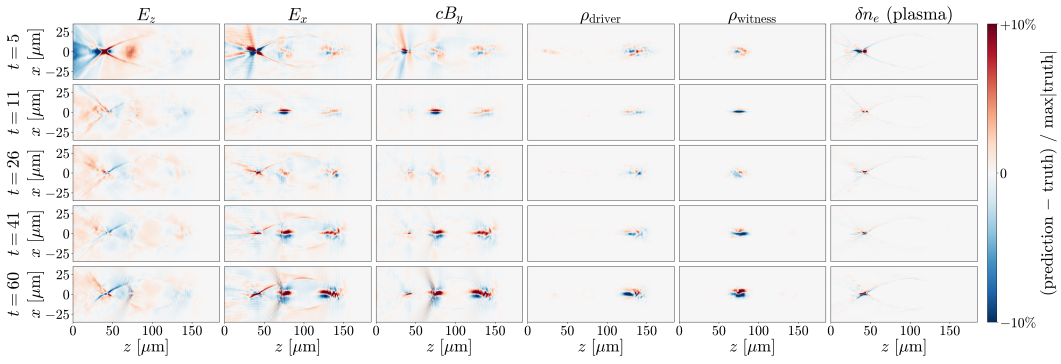

Figure 26: PWFA Run 355: relative-error maps for representative channels at selected times. Colors show $(\text{prediction} - \text{truth})/\max|\text{truth}|$.

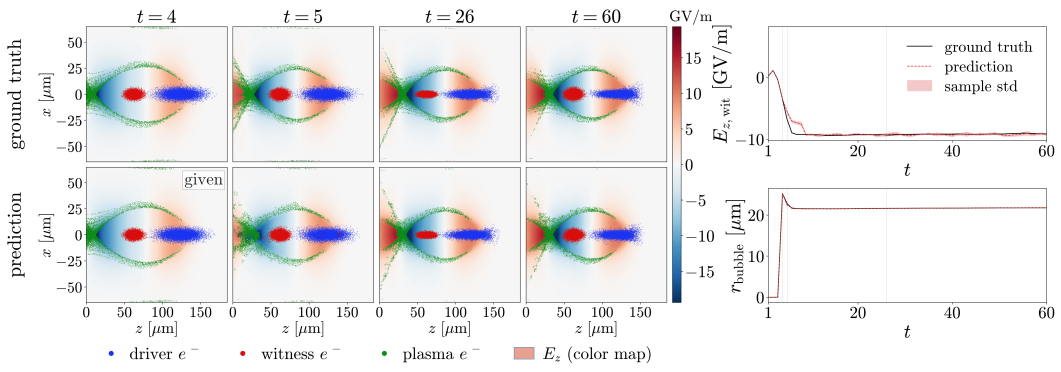

Figure 27: PWFA Run 271: wake evolution in the x–z plane together with trajectory-level observables. The color map shows E_z; particles are overlaid by species. Right: predicted and reference observables with the ensemble sample deviation.

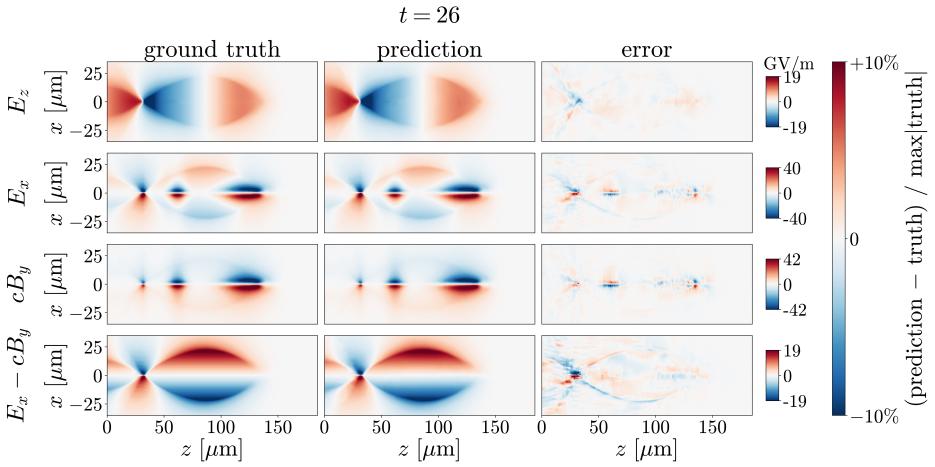

Figure 28: PWFA Run 271: detailed comparison at $t = 25$ for representative field quantities. Left to right: ground truth, prediction, and error.

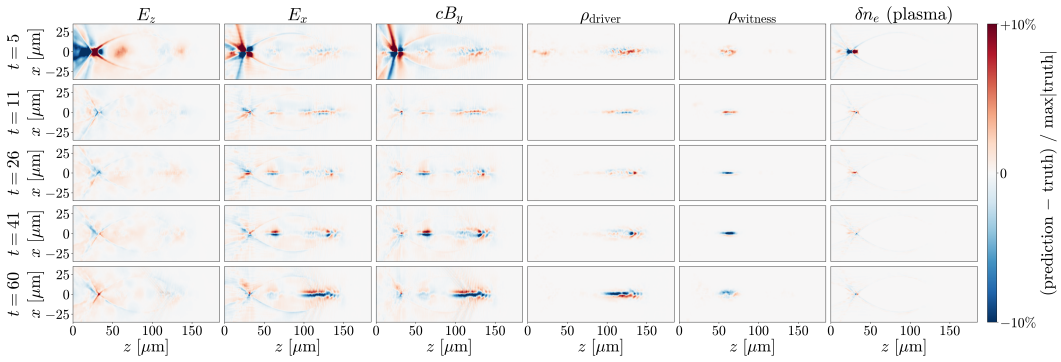

Figure 29: PWFA Run 271: relative-error maps for representative channels at selected times. Colors show $(\text{prediction} - \text{truth})/\max|\text{truth}|$.

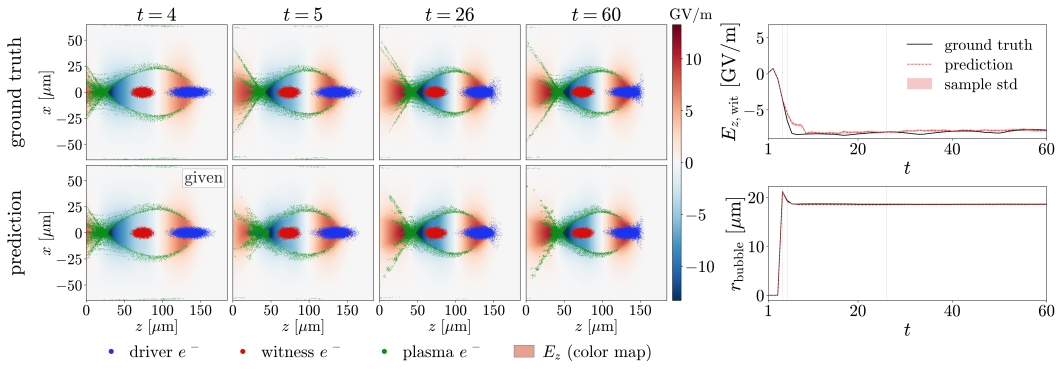

Figure 30: PWFA Run 27: wake evolution in the x–z plane together with trajectory-level observables. The color map shows E_z; particles are overlaid by species. Right: predicted and reference observables with the ensemble sample deviation.

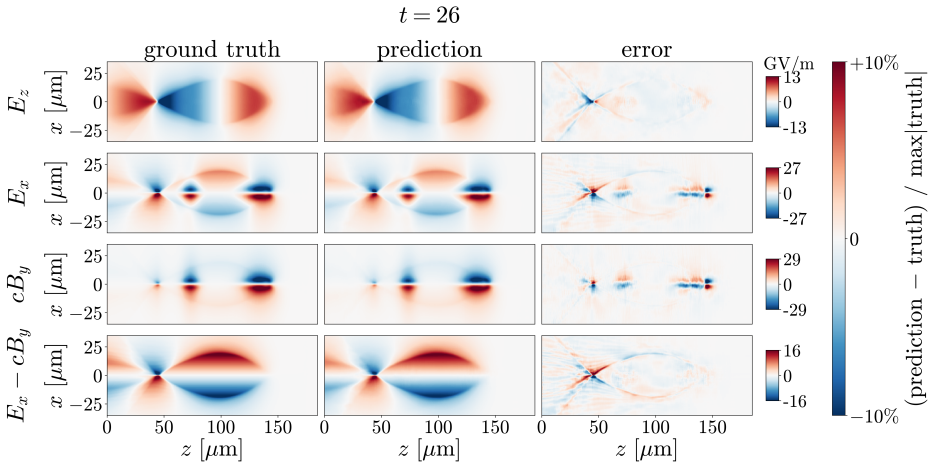

Figure 31: PWFA Run 27: detailed comparison at $t = 25$ for representative field quantities. Left to right: ground truth, prediction, and error.

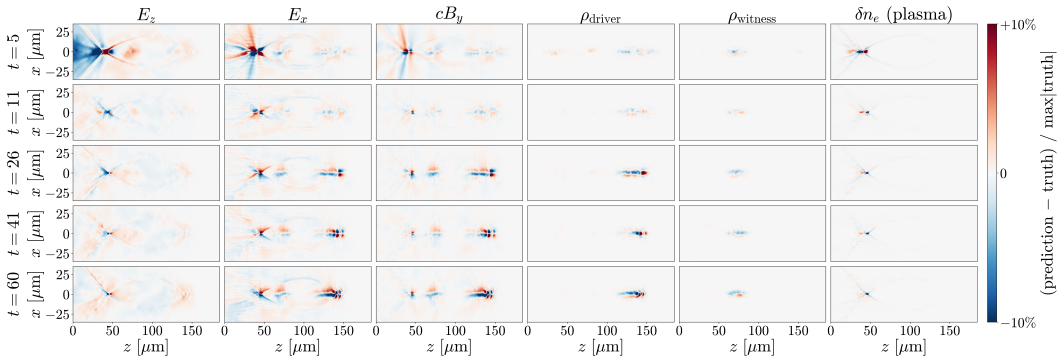

Figure 32: PWFA Run 27: relative-error maps for representative channels at selected times. Colors show $(\text{prediction} - \text{truth})/\max|\text{truth}|$.

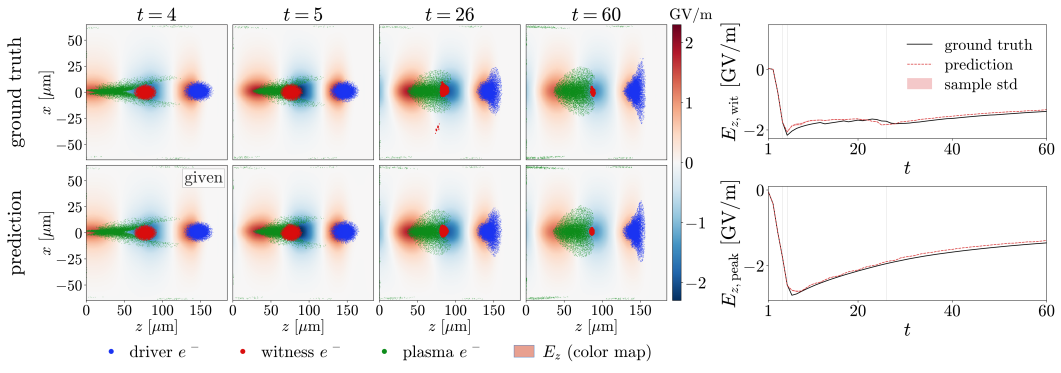

Figure 33: PWFA Run 118: wake evolution in the x–z plane together with trajectory-level observables. The color map shows E_z; particles are overlaid by species. Right: predicted and reference observables with the ensemble sample deviation.

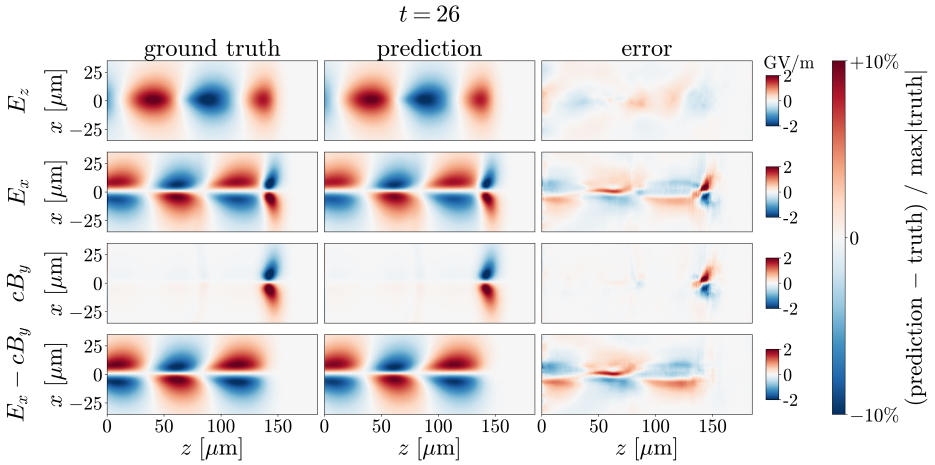

Figure 34: PWFA Run 118: detailed comparison at $t = 25$ for representative field quantities. Left to right: ground truth, prediction, and error.

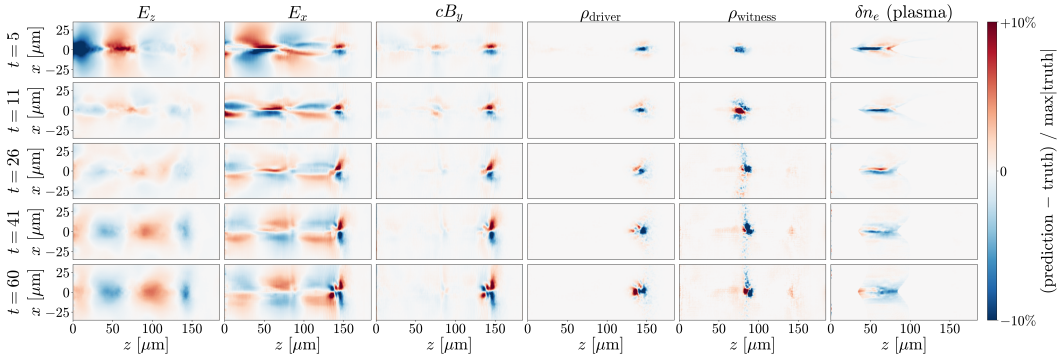

Figure 35: PWFA Run 118: relative-error maps for representative channels at selected times. Colors show $(\text{prediction} - \text{truth}) / \max|\text{truth}|$.