Ambient Physics: Training Neural PDE Solvers with Partial Observations

Anonymous Authors[1]

Abstract

In many scientific settings, acquiring complete observations of PDE coefficients and solutions can be expensive, hazardous, or impossible. Recent diffusion-based methods can reconstruct fields given partial observations, but require complete observations for training. We introduce Ambient Physics, a framework for learning the joint distribution of coefficient-solution pairs directly from partial observations, without requiring a single complete observation. The key idea is to randomly mask a subset of already-observed measurements and supervise on them, so the model cannot distinguish "truly unobserved" from "artificially unobserved", and must produce plausible predictions everywhere. Ambient Physics achieves state-of-the-art reconstruction performance. Compared with prior diffusion-based methods, it achieves a 62.51% reduction in average overall error while using 125× fewer function evaluations. We also identify a "one-point transition": masking a single already-observed point enables learning from partial observations across architectures and measurement patterns. Ambient Physics thus enables scientific progress in settings where complete observations are unavailable.

1. Introduction

Partial differential equations are fundamental to science and engineering (Evans, 2022; Higham et al., 2015). They are mathematical models describing some physical phenomena, in which inputs are coefficients and outputs are solutions, with a wide range of applications from atmospheric modeling to automotive optimization. Problems are often categorized as either the *forward problem*: to predict the solution from a given set of coefficients, or the *inverse problem*: to make deductions about the coefficients from a given solution. In practice, measurements of coefficients and solutions are often incomplete. Climate scientists approximate atmospheric states from sparse weather stations (Lynch, 2006), and geophysicists determine properties of the Earth's subsurface from limited borehole data (Aki & Richards, 2002).

Partial observations present significant challenges. Classical numerical solvers require defined coefficients and boundary conditions, and inverse methods require regularization or strong priors; so predicting the solution from partial observations of the coefficients or making deductions about the coefficients from partial observations of the solution can elicit large uncertainties. To tackle this challenge, researchers have proposed learning-based methods to reconstruct partial observations into physically-plausible complete observations. In particular, founded on Diffusion Posterior Sampling (DPS) (Chung et al., 2022), DiffusionPDE and FunDPS make use of diffusion models to learn the joint distribution of coefficient-solution pairs $p(a, u)$, where a and u denote the coefficients and solution respectively, and then apply DPS to sample from the posterior $p(a, u|A_a a, A_u u)$, where A_a and A_u denote measurement operators, and $A_a a$ and $A_u u$ are the corresponding partial observations (Huang et al., 2024; Yao et al., 2025).

Both DiffusionPDE and FunDPS exhibit exceptional reconstruction performance. With partial observations of only 3%, these diffusion-based methods can sample physically plausible complete observations from the posterior distribution; and across diverse PDEs, achieve relative L_2 errors in the range of 2–20%. In contrast, more traditional neural PDE solvers (DeepONet (Lu et al., 2019), FNO (Li et al., 2020), PINN (Raissi et al., 2019), PINO (Li et al., 2024)) struggle with partial observations and achieve errors that are often an order of magnitude larger. Still, there is a fundamental motivational misalignment: DiffusionPDE and FunDPS require complete observations to learn the joint distribution of coefficient-solution pairs, having been pre-trained on 50,000 complete observations. For many real-world problems—especially those that would most benefit from reconstruction—acquiring 50,000 complete measurements is infeasible. Naive training directly on partial observations fails catastrophically (see Figure 3). In that setting, a diffusion model can learn only the distribution of partial observations $p(A_a a, A_u u)$, predicting accurately only at observed locations and arbitrarily elsewhere.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

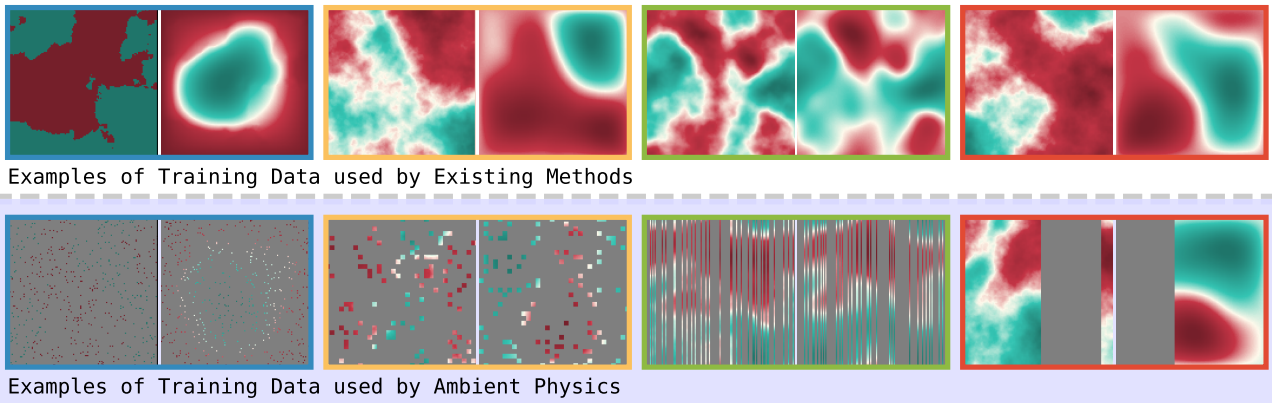

Figure 1. **Examples of Training Data used by Existing Methods vs Ambient Physics. Top:** Existing diffusion-based methods require access to complete observations to learn the joint distribution of coefficient-solution pairs. **Bottom:** Ambient Physics learns the joint distribution directly from partial observations, without requiring a single complete observation. **Left to Right:** Darcy flow with point observations, Helmholtz with patch observations, Navier-Stokes with slice observations, and Poisson with window observations. Within each panel, the left field represents the coefficients (or initial condition) and the right field represents the solution (or final state).

To address the motivational misalignment described above, we propose Ambient Physics, a framework for learning the joint distribution of coefficient-solution pairs directly from partial observations. The key idea is to randomly mask a subset of already-observed measurements and supervise on them, so that the model cannot distinguish "truly unobserved" from "artificially unobserved" and must produce physically plausible predictions everywhere to minimize error. Concretely, our contributions are as follows:

- We formulate the *reconstruction problem*, which treats both the classical forward and inverse problems as special cases.
- We show that naive training on partial observations fails catastrophically: models predict accurately only at observed locations and arbitrarily elsewhere.
- We demonstrate that, across Darcy flow, Helmholtz, Navier-Stokes, and Poisson, Ambient Physics achieves a 62.51% reduction in average overall error while using 125× fewer function evaluations., compared with prior diffusion-based methods. We emphasize that Ambient Physics is trained entirely on partial observations, whereas prior methods require complete observations.
- We demonstrate that Ambient Physics is not tied to diffusion or flow matching by adapting traditional neural PDE solvers to learn directly from partial observations.
- We identify a "one-point transition": masking a single already-observed point enables learning from partial observations across architectures and measurement patterns. We also evaluate robustness under different observation patterns, and evaluate super-resolution without any high-resolution observations.

2. Background

Partial Differential Equations. We consider partial differential equations of the form:

$$\mathcal{L}_a u = 0, \quad \text{on } \Omega,$$
$$u_{|\partial\Omega} = g,$$

where $\mathcal{L}_a$ represents a differential operator, $a : \Omega \to \mathbb{R}^n$ represents the coefficients, $u : \Omega \to \mathbb{R}^n$ represents the solution, and $g : \partial\Omega \to \mathbb{R}^{n_b}$ represents the boundary conditions. For time-dependent PDEs, we may view a as the initial condition $u(0)$ and u as the final state $u(T)$ at time T.

The forward problem is to map a given set of coefficients (or initial conditions) a to the solution u, while the inverse problem is to map the solution u to a given set of coefficients a. These problems induce corresponding operators: the forward operator $\mathcal{M}_a : a \to u$, and the inverse operator $\mathcal{M}_u : u \to a$.

Neural PDE Solvers. A neural PDE solver approximates the forward or inverse operator $\mathcal{M}$ with a learned model $\mathcal{M}^\theta$. These methods are often trained by minimizing the data loss

$$\|x_{\text{true}} - x_{\text{pred}}\|_2,$$

where x denotes the coefficients a or solution u, using data from high-fidelity numerical solvers or real-world observations, and sometimes the PDE loss which penalizes violations of the governing equation, or a combination of both. However, in many cases, incorporating the PDE loss only yields marginal improvements (Krishnapriyan et al., 2021; Wang et al., 2022), and with incomplete inputs, the required derivatives become difficult to compute (Huang et al., 2024).

Neural PDE solvers are evaluated by measuring the error between ground truth and predictions. In practice, specific requirements of the problem or task may dictate the choice of the metric. For example, climate models may prioritize long-term accuracy and stability over pointwise accuracy, while engineering designs may prioritize maximum tolerances in critical regions over average performance. Following prior work, we report the *relative L_2 error*

$$\frac{\|x_{\text{true}} - x_{\text{pred}}\|_2}{\|x_{\text{true}}\|_2}$$

for continuous-valued fields, which is particularly useful for comparing performance across fields with different magnitudes and across different PDEs. For discrete-valued fields, we report the *misclassification rate*

$$\frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\left[x_{\text{true}}^{(i)} \neq x_{\text{pred}}^{(i)}\right].$$

Neural PDE solvers are commonly grouped into two categories: physics-informed methods and operator-learning methods. Physics-Informed Neural Networks (PINNs) incorporate physical laws, typically described by partial differential equations, into the loss functions to guide training (Raissi et al., 2019). Several extensions improve parallelization and scalability: conservative PINNs (cPINNs) apply separate neural networks on subdomains and enforce flux continuity at subdomain interfaces (Jagtap et al., 2020); eXtended PINNs (XPINNs) generalize this domain decomposition approach to arbitrary domains, geometries, and PDE types (Jagtap & Karniadakis, 2020). In contrast, neural operators learn mappings between infinite-dimensional function spaces (Kovachki et al., 2023). Deep Operator Networks (DeepONets) use two sub-networks that combine via a dot product to produce predictions at queried locations (Lu et al., 2019). Fourier Neural Operators (FNOs) parameterize the solution operator via learnable integral kernels in Fourier space (Li et al., 2020). Many extensions improve expressivity and transferability: Factorized-FNO (F-FNO) uses separable spectral layers and modern residual connections and training strategies (Tran et al., 2021); Geometry-aware FNO (Geo-FNO) uses message passing to adapt operator-learning to irregular geometries (Li et al., 2023). Physics-Informed Neural Operator (PINO) incorporates PDE residuals into the loss function to bridge the gap between physics-informed and operator-learning methods (Li et al., 2024). These methods have been applied across a wide range of domains, including heat transfer (Zobeiry & Humfeld, 2021; Cai et al., 2021; Lu & Wang, 2024; Cao et al., 2024), electromagnetic fields (Baldan et al., 2023; Piao et al., 2024; Peng et al., 2022; Augenstein et al., 2023), and fluid mechanics (Jin et al., 2021; Eivazi et al., 2022; Luo et al., 2023; Azizzadenesheli et al., 2024).

Reconstruction Under Partial Observations. Several works study reconstruction under partial observations, particularly in fluid mechanics where real-world data are often collected by a limited number of probes, yielding observations only at sparse locations or along a small number of slices, and sometimes at low-resolution. For instance, transformer-based architectures have been used to reconstruct two- and three-dimensional flows from sparse locations (Zhang et al., 2025), and convolutional neural networks have been used to reconstruct three-dimensional flows from two-dimensional cross-sectional slices (Matsuo et al., 2024). More recently, diffusion models have been adopted for this problem because they generate high-quality samples and support uncertainty quantification. A number of works modify diffusion training or sampling by incorporating PDE residuals, and demonstrate applications to field reconstruction and super-resolution tasks for two- and three-dimensional flows (Yousif et al., 2021; Shu et al., 2023; Jacobsen et al., 2025; Amorós-Trepat et al., 2026). DiffusionPDE (Huang et al., 2024) learns the joint distribution of coefficient-solution pairs $p(a, u)$ and then applies Diffusion Posterior Sampling (Chung et al., 2022) to guide the denoising process using partial observations and PDE residuals, while FunDPS (Yao et al., 2025) extends this framework to function-space diffusion models. Importantly, **while these works are motivated by scientific settings in which complete observations are unavailable, the experimental setups exclusively use complete observations for training and validation**, with datasets ranging from 1,000–50,000 complete observations. This motivates methods that can learn directly from partial observations.

Diffusion Models for Inverse Problems. Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) are effective generative priors for inverse problems, enabling reconstruction by generating samples that are consistent with a learned prior and the available observations. This is often done by training an unconditional diffusion model on complete observations and then performing approximate posterior sampling by guiding the denoising process with a measurement-consistency objective, optionally augmented with additional constraints. Examples include DDRM for linear inverse problems (Kawar et al., 2022) and DPS for more general inverse problems (Chung et al., 2022). However, these methods assume access to fully observed training samples. In contrast, "ambient" methods aim to learn a generative model of the underlying data distribution given only incomplete or noisy measurements. AmbientGAN shows that learning the target distribution from partially observed training samples is possible (Bora et al., 2018), and Ambient Diffusion extends this perspective to diffusion models (Daras et al., 2023). This enables learning generative priors directly from partial observations, without requiring a single complete observation.

3. Method

The Reconstruction Problem. The PDE literature typically distinguishes between the *classical forward problem* $a \to u$ and the *classical inverse problem* $u \to a$. In this work, we propose a generalized formulation that treats both as special cases and explicitly addresses real-world partial observations.

Let $A_a : \mathbb{R}^n \to \mathbb{R}^n$ and $A_u : \mathbb{R}^n \to \mathbb{R}^n$ represent (possibly non-invertible) measurement operators (e.g., binary masks) that act on coefficients and solutions, respectively. Given partial observations $(A_a a, A_u u)$, the *reconstruction problem* is to predict a coefficient-solution pair (a, u) that agrees with the partial observations and is physically-plausible. This perspective is closely analogous to image inpainting with multiple channels and per-channel masks, as well as many other tasks in computational imaging (e.g., X-ray computed tomography).

This generalized formulation treats the classical forward and inverse problems as special cases. Let $\mathbf{I}$ denote the identity operator and $\mathbf{0}$ denote the null operator. When $A_a = \mathbf{I}$ and $A_u = \mathbf{0}$, the reconstruction problem reduces to the classical forward problem:

$$(\mathbf{I}a, \mathbf{0}u) = (a, \emptyset) \to (a, u).$$

Conversely, when $A_a = \mathbf{0}$ and $A_u = \mathbf{I}$, the reconstruction problem reduces to the classical inverse problem:

$$(\mathbf{0}a, \mathbf{I}u) = (\emptyset, u) \to (a, u).$$

It also captures settings not covered by the classical dichotomy. For example, when $A_a \notin \{\mathbf{0}, \mathbf{I}\}$ and $A_u = \mathbf{0}$, the reconstruction problem becomes *coefficient inpainting*, where the goal is to predict the coefficients given partial coefficients. More generally, both the coefficients and solution may be incomplete, i.e., $A_a, A_u \notin \{\mathbf{0}, \mathbf{I}\}$. All of these cases fall under our proposed generalized formulation. In the remainder of this work, we use "reconstruction problem" to refer to this unified task of predicting coefficient-solution pairs (a, u) given partial observations $(A_a a, A_u u)$, including the classical forward and inverse problems as special cases.

Objective. Our objective is to solve the reconstruction problem by learning the joint distribution of coefficient-solution pairs $p(a, u)$, and then, at inference, sampling from the posterior $p(a, u \mid A_a a, A_u u)$. Prior diffusion-based methods adopt this strategy, but they require access to complete observations (a, u) during training. In many scientific and engineering settings, however, only partial observations $(A_a a, A_u u)$ are available. This makes learning the joint distribution strictly more challenging: we must learn $p(a, u)$ directly from partial observations $(A_a a, A_u u)$.

Flow Matching. In this work, we adopt flow matching as our generative modeling framework (Lipman et al., 2022). Flow matching learns a transport from a simple base distribution (e.g., Gaussian noise $x_0 \sim \mathcal{N}(0, I)$) to the data distribution (e.g., $x_1 \sim p_{\text{data}}$) by sampling an interpolation between noise x_0 and data x_1 and training a neural network to predict the corresponding flow field. Flow matching also supports conditional generation (e.g., posterior sampling).

We use rectified flow (Liu et al., 2022), which has a particularly simple training and sampling procedure and typically performs competitively with more complex score-based diffusion models. While common practice for rectified flow is to parameterize the network to predict the instantaneous velocity, we parameterize our network to predict the final state x_1.

Naive Training. With only partial observations available, a natural baseline is to train a conditional flow matching model directly on the available measurements.

Let $\epsilon_a, \epsilon_u \sim \mathcal{N}(0, I)$ denote independent Gaussian noise and let $t \sim \mathcal{U}([0, 1))$ denote the flow time. Following rectified flow, one forms *mid-flow states* (i.e., noised partial observations) by linearly interpolating between noise and the measurements:

$$a_t = t \cdot (A_a a) + (1 - t) \cdot \epsilon_a,$$
$$u_t = t \cdot (A_u u) + (1 - t) \cdot \epsilon_u.$$

Then, one can train a conditional model $\mathcal{M}^\theta$ to predict the coefficient-solution pair (a, u) from the mid-flow states (a_t, u_t) while conditioning on partial observations $(A_a a, A_u u)$, masks (A_a, A_u), and time t:

$$\mathcal{M}^\theta(A_a a_t, A_u u_t, A_a a, A_u u, A_a, A_u, t) = (\hat{a}, \hat{u}).$$

Conditioning on partial observations $(A_a a, A_u u)$ is necessary so that, at inference, generated fields depend on, and agree with, the available measurements; removing it would yield unconditional samples that are not constrained to match partial observations $(A_a a, A_u u)$. Since only partial observations $(A_a a, A_u u)$ are available for training, supervision is restricted to the available measurements:

$$\mathcal{L}_{\text{meas}} = \|A_a a - A_a \hat{a}\|_2 + \|A_u u - A_u \hat{u}\|_2.$$

This naive training procedure leads to catastrophic failure: the model predicts accurately only at the available measurements and arbitrarily elsewhere (Figure 3). Since the loss $\mathcal{L}_{\text{meas}}$ penalizes errors only at observed locations while the full partial observations $(A_a a, A_u u)$ are also provided as conditioning, the loss can be minimized by simply copying the observed values into $(\hat{a}, \hat{u})$, yielding near-perfect predictions at observed locations. Everywhere else, however, the model receives no training signal. Predictions at unobserved locations do not affect $\mathcal{L}_{\text{meas}}$, so the model can make arbitrary predictions.

Ambient Physics Training. We propose Ambient Physics training to learn the joint distribution of coefficient-solution pairs directly from partial observations $(A_a a, A_u u)$. Naive training provides no training signal at unobserved locations, so to obtain physically plausible predictions everywhere, additional training signal is needed at unobserved locations. The key idea is to randomly mask a subset of already-observed measurements and supervise on these artificially masked measurements. As a result, the model cannot distinguish "truly unobserved" from "artificially unobserved" and must produce physically plausible predictions across the entire domain to minimize error.

Let $B_a : \mathbb{R}^n \to \mathbb{R}^n$ and $B_u : \mathbb{R}^n \to \mathbb{R}^n$ denote additional masking operators that further mask the partial observations. In particular, B_a and B_u are chosen so that they mask only a subset of already-observed measurements (i.e., they zero-out entries that are observed under A_a and A_u, and never reveal new ones). Consequently, the inputs $(B_a A_a a,\ B_u A_u u)$ contain strictly less information than the training observations $(A_a a,\ A_u u)$. For notational convenience, we define $\tilde{B}_x := B_x A_x$.

Let $\epsilon_a, \epsilon_u \sim \mathcal{N}(0, I)$ denote independent Gaussian noise and let $t \sim \mathcal{U}([0, 1))$ denote the flow time. Following rectified flow, we form mid-flow states by linearly interpolating between noise and the masked partial observations:

$$a_t = t \cdot (\tilde{B}_a a) + (1 - t) \cdot \epsilon_a,$$
$$u_t = t \cdot (\tilde{B}_u u) + (1 - t) \cdot \epsilon_u.$$

Then, we train a conditional model $\mathcal{M}^\theta$ to predict the clean fields (a, u) from (a_t, u_t) while conditioning on the masked partial observations $(\tilde{B}_a a, \tilde{B}_u u)$, masks $(\tilde{B}_a, \tilde{B}_u)$, and time t:

$$\mathcal{M}^\theta(\tilde{B}_a a_t, \tilde{B}_u u_t, \tilde{B}_a a, \tilde{B}_u u, \tilde{B}_a, \tilde{B}_u, t) = (\hat{a}, \hat{u}).$$

Critically, the model is given strictly less information than the training observations, but it is still supervised on all available measurements $(A_a a, A_u u)$:

$$\mathcal{L}_{\text{meas}} = \|A_a a - A_a \hat{a}\|_2 + \|A_u u - A_u \hat{u}\|_2.$$

In contrast to naive training, which can minimize $\mathcal{L}_{\text{meas}}$ by simply copying the observed values and predicting arbitrarily elsewhere, Ambient Physics withholds a random subset of the available measurements from the conditioning using (B_a, B_u). The model is still supervised on the full set of available measurements $(A_a a, A_u u)$ through $\mathcal{L}_{\text{meas}}$, so it must predict the held-out ("artificially unobserved") measurements from the remaining context. Since the model cannot distinguish truly unobserved points from artificially unobserved ones, the only way to minimize $\mathcal{L}_{\text{meas}}$ is to learn the joint distribution of coefficient-solution pairs, enabling posterior sampling of fields that are physically plausible across the entire domain.

Ambient Physics Sampling. During training, the Ambient Physics model receives as input the masked mid-flow states $(\tilde{B}_a a_t, \tilde{B}_u u_t)$ and conditions on the masked partial observations $(\tilde{B}_a a, \tilde{B}_u u)$, the corresponding masks $(\tilde{B}_a, \tilde{B}_u)$, and time t. At inference, we are given partial observations $(A_a a, A_u u)$ and our goal is to sample from the posterior $p(a, u \mid A_a a, A_u u)$. Crucially, sampling must replicate the same masking and conditioning structure used during training.

We first sample Gaussian noise $a_0, u_0 \sim \mathcal{N}(0, I)$ and masking operators (B_a, B_u) from the same distribution used during training, i.e., $B_a \sim p(B_a \mid A_a)$ and $B_u \sim p(B_u \mid A_u)$. We then numerically integrate the rectified-flow ODE from $t = 0$ to $t \approx 1$.

To do so, we discretize time as $t_k = k/K$ for $k = 0, \ldots, K$, where K is the number of sampling steps (function evaluations), and we maintain states (a_k, u_k) that approximate (a_{t_k}, u_{t_k}).

At each step k, we predict the corresponding clean fields by applying the same masking/conditioning structure used during training:

$$\mathcal{M}^\theta(\tilde{B}_a a_k, \tilde{B}_u u_k, \tilde{B}_a a, \tilde{B}_u u, \tilde{B}_a, \tilde{B}_u, t_k) = (\hat{a}_k, \hat{u}_k).$$

We then compute the rectified-flow velocity at t_k as

$$v_a(t_k) = \frac{\hat{a}_k - a_k}{(1 - t_k)},$$
$$v_u(t_k) = \frac{\hat{u}_k - u_k}{(1 - t_k)},$$

and step forward with an ODE solver (e.g., Euler or Heun; we use Euler):

$$a_{k+1} = a_k + \frac{1}{K} \cdot v_a(t_k),$$
$$u_{k+1} = u_k + \frac{1}{K} \cdot v_u(t_k).$$

The final state (a_K, u_K) is returned as a reconstruction sample $(\hat{a}, \hat{u})$.

Sampling the masking operators (B_a, B_u) introduces an additional source of randomness beyond the Gaussian noise. This allows us to draw different samples from the conditional distributions $p(a, u \mid \tilde{B}_a a, \tilde{B}_u u)$ induced by different masking operators, and can be viewed as sampling from a mixture over masks. In particular,

$$p(a, u \mid A_a a, A_u u) \approx \mathbb{E}_{B_a, B_u}\left[p(a, u \mid \tilde{B}_a a, \tilde{B}_u u)\right].$$

This can improve robustness and provide a practical way to quantify uncertainty. In practice, we find that a single mask draw and a single reconstruction sample from $p(a, u \mid \tilde{B}_a a, \tilde{B}_u u)$ performs well enough.

	Partial Obs. Training	NFE	Darcy Flow		Helmholtz		Navier-Stokes		Poisson	
			Coefficient	Solution	Coefficient	Solution	Coefficient	Solution	Coefficient	Solution
DeepONet	✗	–	41.1%	38.3%	132.8%	123.1%	97.2%	103.2%	105.8%	155.5%
FNO	✗	–	49.3%	28.2%	218.2%	98.2%	96.0%	101.4%	232.7%	100.9%
PINN	✗	–	59.7%	48.8%	160.0%	142.3%	146.8%	142.7%	130.0%	128.1%
PINO	✗	–	49.2%	35.2%	216.9%	106.5%	96.0%	101.4%	231.9%	107.1%
DiffusionPDE	✗	2000	7.87%	6.07%	19.07%	12.64%	9.63%	3.78%	21.10%	4.88%
FunDPS	✗	200	6.78%	2.88%	20.07%	2.20%	9.87%	3.99%	24.04%	2.04%
		500	5.18%	2.49%	17.16%	2.13%	8.48%	3.32%	20.47%	1.99%
Ambient Flow	✓	1	2.08%	0.76%	7.73%	0.47%	6.41%	0.98%	7.86%	0.48%
		4	**2.07%**	**0.74%**	**7.67%**	**0.44%**	**6.38%**	**0.96%**	**7.81%**	**0.45%**
		16	**2.07%**	**0.74%**	**7.67%**	**0.44%**	**6.38%**	**0.96%**	**7.81%**	**0.45%**

Table 1. **Reconstruction Performance Under Partial Observations.** Relative L_2 error (%) between ground truth and reconstructed coefficients ($\hat{a}$) and solutions ($\hat{u}$) from partial observations $(A_a a, A_u u)$ with 3% uniformly random measurements. *Note: In prior work these are reported as "Inverse" for coefficients and "Forward" for solutions.* "Partial Obs. Training" indicates whether training uses partial observations (✓) or complete observations (✗); "NFE" is the number of function evaluations. Across all PDEs and both fields, Ambient Flow achieves the lowest relative L_2 error despite training only on partial observations and sampling with just 1–16 NFE.

4. Results

Reconstruction Under Partial Observations. On the reconstruction problem, we compare traditional neural PDE solvers and prior state-of-the-art diffusion-based methods against Ambient Flow—our instantiation of Ambient Physics. The fundamental question is *whether Ambient Physics can learn the joint distribution of coefficient-solution pairs directly from partial observations.*

In this experimental setup, we are given partial observations $(A_a a, A_u u)$, where A_a and A_u denote independently sampled binary masks that retain only 3% uniformly random measurements. The goal is to predict a coefficient-solution pair (a, u) that agrees with the partial observations and is physically plausible.

We consider four diverse PDEs: Darcy flow, Helmholtz, Navier-Stokes, and Poisson. Following prior work, reconstruction performance is measured using the relative L_2 error between ground truth and prediction (reported as a percentage), and computed over the entire field for each coefficient/solution and each test example, and then averaged across all examples.

We compare against traditional neural PDE solvers (DeepONet (Lu et al., 2019), FNO (Li et al., 2020), PINN (Raissi et al., 2019), PINO (Li et al., 2024)), which learn directional mappings and therefore require two separate models per PDE. We also include diffusion-based methods (DiffusionPDE (Huang et al., 2024), FunDPS (Yao et al., 2025)), which learn the joint distribution of coefficient-solution pairs and then perform posterior sampling using the available partial observations. Importantly, all baselines require access to complete observations to learn either the directional mappings or the joint distribution. In contrast, Ambient Flow does not require a single complete observation and can learn the joint distribution directly from partial observations.

In Ambient Flow, we follow the Ambient Physics procedure from section 3. Starting from partial observations $(A_a a, A_u u)$, we sample additional masks (B_a, B_u) that withhold one-third of the observed points. Thus, the model is conditioned on $(B_a A_a a,\ B_u A_u u)$, which retain only 2% uniformly random measurements overall, while we supervise on all available measurements $(A_a a, A_u u)$. We use the UNet of Ho et al. (2020) and sample with an Euler scheme. Implementation details are provided in Appendix A.

Table 5 summarizes reconstruction performance across PDEs and fields. Traditional neural PDE solvers perform poorly, with relative L_2 errors ranging from 28.2% to 232.7%—as expected, since these solvers are not designed for reconstruction, particularly under extreme sparsity (3% uniformly random measurements). Diffusion-based methods perform substantially better, with errors on reconstructed coefficients ranging from 5.18% to 24.04% and on reconstructed solutions ranging from 1.99% to 12.64%.

Across all PDEs and both fields, Ambient Flow achieves state-of-the-art reconstruction performance, with errors on coefficients ranging from 2.07% to 7.86% and errors on solutions ranging from 0.44% to 0.96%. Compared to the prior state-of-the-art method (FunDPS at 500 NFE), Ambient Flow achieves a 62.51% reduction in average overall error while using 125× fewer function evaluations, despite being trained directly on partial observations. Specifically, it reduces error by 65.16% on Darcy flow, 67.32% on Helmholtz, 47.92% on Navier–Stokes, and 69.62% on Poisson, with reductions of 50.49% on coefficients and 74.52% on solutions.

These results address the fundamental question: *Ambient Physics can learn the joint distribution of coefficient-solution pairs directly from partial observations*, and it achieves state-of-the-art reconstruction performance.

	Partial Obs. Training		Darcy Flow		Helmholtz		Navier-Stokes		Poisson	
			Coefficient	Solution	Coefficient	Solution	Coefficient	Solution	Coefficient	Solution
DeepONet	✗	–	41.1%	38.3%	132.8%	123.1%	97.2%	103.2%	105.8%	155.5%
FNO	✗	–	49.3%	28.2%	218.2%	98.2%	96.0%	101.4%	232.7%	100.9%
PINN	✗	–	59.7%	48.8%	160.0%	142.3%	146.8%	142.7%	130.0%	128.1%
PINO	✗	–	49.2%	35.2%	216.9%	106.5%	96.0%	101.4%	231.9%	107.1%
		–	6.79%	24.90%	22.86%	5.85%	25.78%	45.92%	22.16%	5.76%
Ambient FNO	✓	+Mask Conditioning	6.11%	27.20%	20.69%	5.35%	26.99%	41.93%	21.62%	7.15%
		+Joint Dist. Modeling	**5.38%**	**2.56%**	**16.57%**	**2.86%**	**8.84%**	**2.12%**	**17.20%**	**2.99%**
		–	2.55%	2.59%	9.16%	2.00%	8.25%	3.57%	9.65%	1.94%
Ambient UNet	✓	+Mask Conditioning	**2.21%**	2.57%	**7.97%**	1.99%	**7.35%**	3.36%	**8.32%**	1.87%
		+Joint Dist. Modeling	2.46%	**0.64%**	10.81%	**0.49%**	8.33%	**1.25%**	10.53%	**0.48%**

Table 2. **Reconstruction Performance Under Partial Observations.** Relative L_2 error (%) between ground truth and reconstructed coefficients ($\hat{a}$) and solutions ($\hat{u}$) from partial observations ($A_a a$, $A_u u$) with 3% uniformly random measurements. *Note: In prior work these are reported as "Inverse" for coefficients and "Forward" for solutions.* "Partial Obs. Training" indicates whether training uses partial observations (✓) or complete observations (✗). Red entries indicate settings that exhibit visible artifacts (Figure 2). Ambient FNO and Ambient UNet perform competitively to diffusion-based methods despite training only on partial observations.

Beyond Diffusion and Flow Matching. Prior work observed that DeepONet and FNO perform poorly with sparse inputs (Huang et al., 2024; Yao et al., 2025), and therefore use complete observations during training and partial observations only at inference. We ask *whether Ambient Physics can adapt traditional neural PDE solvers to learn directly from partial observations.*

We consider the same experimental setup as before and start from an FNO backbone, which we adapt for reconstruction under partial observations. The goal is to learn directional reconstruction maps, $A_a a \to u$ and $A_u u \to a$, while supervising only on the available measurements. This objective can be viewed as an instance of Ambient Physics training: the model is conditioned on masked partial observations $(B_a A_a a, B_u A_u u)$ and supervised on the available measurements $(A_a a, A_u u)$. For $A_a a \to u$, we effectively set $B_a = I$ and $B_u = 0$, and for $A_u u \to a$, we set $B_a = 0$ and $B_u = I$. We refer to this model as Ambient FNO.

Table 2 summarizes reconstruction performance across PDEs and fields. Compared to a vanilla FNO, Ambient FNO achieves a 74.27% reduction in average overall error despite training only on partial observations. However, the reconstructed solutions $\hat{u}$ occasionally exhibit visible artifacts: Darcy flow solutions can appear spatially shifted, and Navier-Stokes final states can resemble blurred initial conditions (see Figure 2). We test two modifications: `+Mask Conditioning`, which yields minimal improvement, and `+Joint Dist. Modeling`, where B_a and B_u mask only a subset of already-observed values ($B_a, B_u \notin \{I, 0\}$), which yields substantial improvements and eliminates visible artifacts. Compared to DiffusionPDE, Ambient FNO `+Joint Dist. Modeling` achieves a 36.16% reduction in average overall error and is competitive with FunDPS at 200 NFE. We repeat these ablations using the UNet of Ho et al. (2020): compared to the prior state-of-the-art method (FunDPS at 500 NFE), Ambient UNet achieves a 53.76% reduction in average overall error.

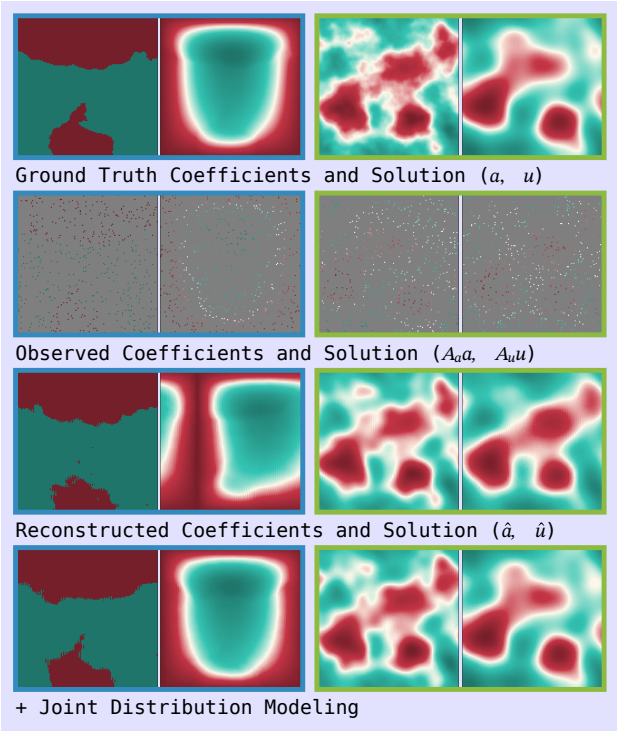

Figure 2. **Examples of Ambient FNO. Reconstructions From Partial Observations. Top to Bottom:** Ground truth coefficients and solution (a, u), partial observations $(A_a a, A_u u)$ with 3% uniformly random measurements, Ambient FNO reconstructed coefficients and solutions $(\hat{a}, \hat{u})$, Ambient FNO `+Joint Dist. Modeling` reconstructed coefficients and solutions $(\hat{a}, \hat{u})$. **Left:** Darcy flow. **Right:** Navier-Stokes. Within each panel, the left field represents the coefficients (or initial condition) and the right field represents the solution (or final state).

These results address our question: *Ambient Physics can adapt traditional neural PDE solvers to learn directly from partial observations,* and it is not tied to diffusion or flow matching.

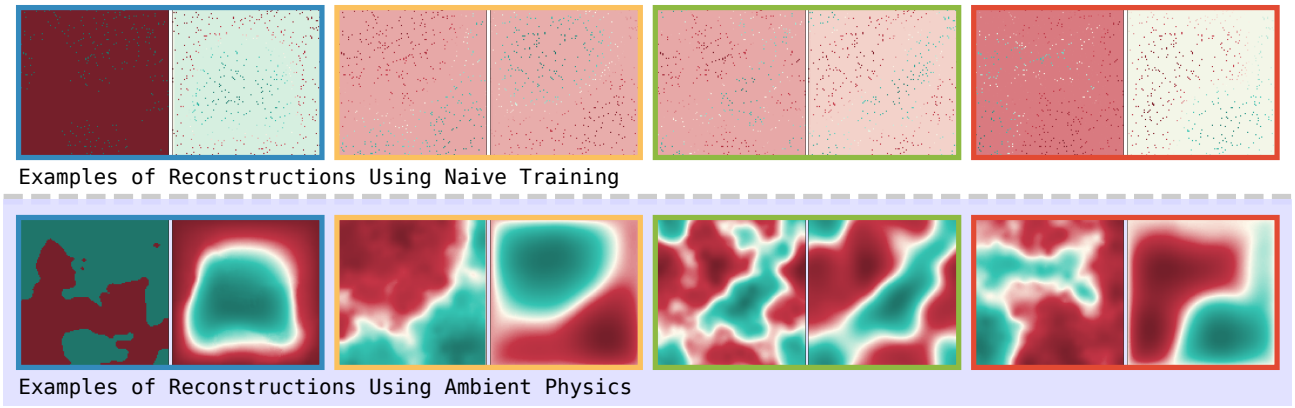

Figure 3. **Naive Training vs Ambient Physics Predictions from Partial Observations. Top:** Naively training on partial observations yields a model that predicts accurately only at observed locations and arbitrarily elsewhere. **Bottom:** Ambient Physics, *trained on the same partial observations,* yields a model that predicts physically-plausible fields everywhere. **Left to Right:** Darcy flow with point observations, Helmholtz with patch observations, Navier-Stokes with slice observations, and Poisson with window observations. Within each panel, the left field represents the coefficients (or initial condition) and the right field represents the solution (or final state).

One-Point Transition to Ambient Physics. In Ambient Physics training (section 3), we randomly mask a subset of already-observed measurements but still supervise on all available measurements. We ask how large this subset must be. We consider the same experimental setup as before. Starting from naive training ($B = I$), we sample masking operators that withhold $\{0, 1, 2, 4, 8, 16, 32, 64, 128, 256\}$ observed points. Figure 4 shows a sharp threshold: masking a single observed point yields an $\sim 10\times$ reduction in coefficient error and an $\sim 100\times$ reduction in solution error; performance improves up to 128 extra masked points. This "one-point transition" from naive to Ambient Physics training, under an otherwise identical setup, provides evidence that the improvements are primarily attributable to the Ambient Physics framework, rather than incidental choices such as architecture or optimization (see Appendix C).

Robustness Across Patterns and Super-Resolution. Thus far, we have studied Ambient Physics under 3% uniformly random measurements. In many scientific settings, measurements are structured (e.g., localized around probes, collected along slices, or restricted to a region). Here, we study Darcy flow under structured observation patterns—*patch*, *column*, and *window* measurements—across multiple sparsity levels. Across all patterns, we show that Ambient Physics accurately reconstructs both coefficients and solutions, and that its reconstruction performance is not limited to uniformly random sampling (see Appendix D).

In addition, we can treat low-resolution measurements as partial observations of an underlying high-resolution field. We study this setting, and show that Ambient Physics can super-resolution a 32×32 measurement into a 128×128 reconstruction, learning directly from low-resolution measurements, without a single 128×128 measurement.

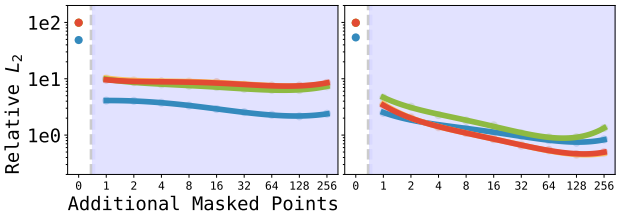

Figure 4. **One-Point Transition to Ambient Physics.** Relative L_2 error (log scale) versus the number of additional masked already-observed points used during training, across Darcy flow, Helmholtz, Navier-Stokes, and Poisson. **Left:** Coefficient error. **Right:** Solution error. The point at 0 additional masked points corresponds to naive training ($B = I$); masking even a single already-observed point leads to a $\sim$ 10–100$\times$ reduction in error.

5. Conclusions

We introduced Ambient Physics, a framework for learning the joint distribution of coefficient–solution pairs $p(a, u)$ directly from partial observations, without requiring a single fully observed training example. Across four diverse PDE, our instantiation Ambient Flow achieves state-of-the-art reconstruction performance under extreme sparsity while reducing the number of function evaluations by orders of magnitude. We further showed that Ambient Physics is not tied to diffusion or flow matching, and identified a "one-point transition" in which masking a single already-observed point is sufficient to move from catastrophic failure to effective learning.

Future work can apply Ambient Physics to real-world scientific and engineering settings where complete observations are unavailable, and explore training a single model that generalizes across diverse measurement patterns and resolutions.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Aki, K. and Richards, P. *Quantitative seismology*. MIT Press, 2002.

Amorós-Trepat, M., Medrano-Navarro, L., Liu, Q., Guastoni, L., and Thuerey, N. Guiding diffusion models to reconstruct flow fields from sparse data. *Physics of Fluids*, 38(1), 2026.

Augenstein, Y., Repan, T., and Rockstuhl, C. Neural operator-based surrogate solver for free-form electromagnetic inverse design. *ACS Photonics*, 10(5):1547–1557, 2023.

Azizzadenesheli, K., Kovachki, N., Li, Z., Liu-Schiaffini, M., Kossaifi, J., and Anandkumar, A. Neural operators for accelerating scientific simulations and design. *Nature Reviews Physics*, 6(5):320–328, 2024.

Baldan, M., Di Barba, P., and Lowther, D. A. Physics-informed neural networks for inverse electromagnetic problems. *IEEE Transactions on Magnetics*, 59(5):1–5, 2023.

Bora, A., Price, E., and Dimakis, A. G. Ambientgan: Generative models from lossy measurements. In *International conference on learning representations*, 2018.

Cai, S., Wang, Z., Wang, S., Perdikaris, P., and Karniadakis, G. E. Physics-informed neural networks for heat transfer problems. *Journal of Heat Transfer*, 143(6):060801, 2021.

Cao, Q., Goswami, S., and Karniadakis, G. E. Laplace neural operator for solving differential equations. *Nature Machine Intelligence*, 6(6):631–640, 2024.

Chung, H., Kim, J., Mccann, M. T., Klasky, M. L., and Ye, J. C. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022.

Daras, G., Shah, K., Dagan, Y., Gollakota, A., Dimakis, A., and Klivans, A. Ambient diffusion: Learning clean distributions from corrupted data. *Advances in Neural Information Processing Systems*, 36:288–313, 2023.

Eivazi, H., Tahani, M., Schlatter, P., and Vinuesa, R. Physics-informed neural networks for solving reynolds-averaged navier–stokes equations. *Physics of Fluids*, 34(7), 2022.

Evans, L. C. *Partial differential equations*, volume 19. American mathematical society, 2022.

Hendrycks, D. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Higham, N. J., Dennis, M. R., Glendinning, P., Martin, P. A., Santosa, F., and Tanner, J. *Princeton companion to applied mathematics*. Princeton University Press, 2015.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Huang, J., Yang, G., Wang, Z., and Park, J. J. Diffusionpde: Generative pde-solving under partial observation. *Advances in Neural Information Processing Systems*, 37: 130291–130323, 2024.

Jacobsen, C., Zhuang, Y., and Duraisamy, K. Cocogen: Physically consistent and conditioned score-based generative models for forward and inverse problems. *SIAM Journal on Scientific Computing*, 47(2):C399–C425, 2025.

Jagtap, A. D. and Karniadakis, G. E. Extended physics-informed neural networks (xpinns): A generalized space-time domain decomposition based deep learning framework for nonlinear partial differential equations. *Communications in Computational Physics*, 28(5), 2020.

Jagtap, A. D., Kharazmi, E., and Karniadakis, G. E. Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering*, 365:113028, 2020.

Jin, X., Cai, S., Li, H., and Karniadakis, G. E. Nsfnets (navier-stokes flow nets): Physics-informed neural networks for the incompressible navier-stokes equations. *Journal of Computational Physics*, 426:109951, 2021.

Kawar, B., Elad, M., Ermon, S., and Song, J. Denoising diffusion restoration models. *Advances in neural information processing systems*, 35:23593–23606, 2022.

Kossaifi, J., Kovachki, N., Azizzadenesheli, K., and Anandkumar, A. Multi-grid tensorized fourier neural operator for high-resolution pdes. *arXiv preprint arXiv:2310.00120*, 2023.

Kovachki, N., Li, Z., Liu, B., Azizzadenesheli, K., Bhattacharya, K., Stuart, A., and Anandkumar, A. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

Krishnapriyan, A., Gholami, A., Zhe, S., Kirby, R., and Mahoney, M. W. Characterizing possible failure modes

in physics-informed neural networks. *Advances in neural information processing systems*, 34:26548–26560, 2021.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

Li, Z., Huang, D. Z., Liu, B., and Anandkumar, A. Fourier neural operator with learned deformations for pdes on general geometries. *Journal of Machine Learning Research*, 24(388):1–26, 2023.

Li, Z., Zheng, H., Kovachki, N., Jin, D., Chen, H., Liu, B., Azizzadenesheli, K., and Anandkumar, A. Physics-informed neural operator for learning partial differential equations. *ACM/IMS Journal of Data Science*, 1(3):1–27, 2024.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Liu, X., Gong, C., and Liu, Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Lu, L., Jin, P., and Karniadakis, G. E. Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *arXiv preprint arXiv:1910.03193*, 2019.

Lu, X. and Wang, Y. Surrogate modeling for radiative heat transfer using physics-informed deep neural operator networks. *Proceedings of the Combustion Institute*, 40 (1-4):105282, 2024.

Luo, Y., Chen, Y., and Zhang, Z. Cfdbench: A large-scale benchmark for machine learning methods in fluid dynamics. *arXiv preprint arXiv:2310.05963*, 2023.

Lynch, P. *The emergence of numerical weather prediction: Richardson's dream*. Cambridge University Press, 2006.

Matsuo, M., Fukami, K., Nakamura, T., Morimoto, M., and Fukagata, K. Reconstructing three-dimensional bluff body wake from sectional flow fields with convolutional neural networks. *SN Computer Science*, 5(3):306, 2024.

Peng, Z., Yang, B., Xu, Y., Wang, F., Liu, L., and Zhang, Y. Rapid surrogate modeling of electromagnetic data in frequency domain using neural operator. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–12, 2022.

Piao, S., Gu, H., Wang, A., and Qin, P. A domain-adaptive physics-informed neural network for inverse problems of maxwell's equations in heterogeneous media. *IEEE Antennas and Wireless Propagation Letters*, 23(10):2905–2909, 2024.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

Shu, D., Li, Z., and Farimani, A. B. A physics-informed diffusion model for high-fidelity flow field reconstruction. *Journal of Computational Physics*, 478:111972, 2023.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. pmlr, 2015.

Tran, A., Mathews, A., Xie, L., and Ong, C. S. Factorized fourier neural operators. *arXiv preprint arXiv:2111.13802*, 2021.

Wang, S., Yu, X., and Perdikaris, P. When and why pinns fail to train: A neural tangent kernel perspective. *Journal of Computational Physics*, 449:110768, 2022.

Yao, J., Mammadov, A., Berner, J., Kerrigan, G., Ye, J. C., Azizzadenesheli, K., and Anandkumar, A. Guided diffusion sampling on function spaces with applications to pdes. *arXiv preprint arXiv:2505.17004*, 2025.

Yousif, M. Z., Yu, L., and Lim, H.-C. High-fidelity reconstruction of turbulent flow from spatially limited data using enhanced super-resolution generative adversarial network. *Physics of Fluids*, 33(12), 2021.

Zhang, Q., Krotov, D., and Karniadakis, G. E. Operator learning for reconstructing flow fields from sparse measurements: an energy transformer approach. *Journal of Computational Physics*, pp. 114148, 2025.

Zobeiry, N. and Humfeld, K. D. A physics-informed machine learning approach for solving heat transfer equation in advanced manufacturing and engineering applications. *Engineering Applications of Artificial Intelligence*, 101: 104232, 2021.

A. Implementation Details

Hardware Details. All models were trained on NVIDIA H100 GPUs. In total, we trained roughly 200 models for the main results and roughly 100 models for early-stage experimentation (e.g., preliminary exploration and hyperparameter tuning), for a total of ∼300 models. Across experiments, the average training time was approximately 250–300 minutes per model (including data loading and evaluation under our standard experimental setup).

Data Details. We use the same PDE datasets as DiffusionPDE (Huang et al., 2024) and FunDPS (Yao et al., 2025): Darcy flow, Helmholtz, unbounded Navier–Stokes, and Poisson. For each PDE, we use 50,000 training samples and 1,000 test samples at a spatial resolution of 128×128.

DiffusionPDE and FunDPS also include a bounded Navier–Stokes dataset; however, we found that approximately 10% of its samples contain non-physical artifacts, and we therefore deemed it unsuitable for our evaluation.

We emphasize that Ambient Physics models never receive fully observed fields as inputs during training or inference. Instead, partial observations $(A_a a, A_u u)$ are generated in the data loader by applying the corresponding measurement operators. In our implementation, the observation pattern is fixed per sample throughout training (i.e., from one epoch to the next), so unobserved locations remain unobserved for that sample.

Architecture Details. For Ambient Flow and Ambient UNet, we use the UNet backbone from DDPM (Ho et al., 2020). For Ambient FNO, we use a modern Fourier Neural Operator (FNO) backbone based on Kossaifi et al. (2023). Architecture hyperparameters are listed in Table 3.

Optimization Details. For all models, we use the AdamW optimizer (Loshchilov & Hutter, 2017). Optimization hyperparameters are listed in Table 3.

Hyperparameter	Ambient Flow	Ambient UNet	Ambient FNO
Architecture			
`fourier_channels`	64	–	–
`hidden_channels`	192	64	64
`ch_mult`	[1, 2, 3, 4]	[1, 2, 3, 4]	–
`is_attn`	[`False`, `False`, `True`, `True`]	[`False`, `False`, `False`, `True`]	–
`md_attn`	`True`	`True`	–
`dropout`	0.1	0.0	–
`attn_heads`	4	4	–
`attn_head_channels`	32	32	–
`depth`	–	–	8
`fft_modes`	–	–	16
`activation`	–	–	`GELU` (Hendrycks, 2016)
Optimization			
`epochs`	25	100	100
`clip_grad_max_norm`	1.0	1.0	1.0
`mixed_precision`	`True`	`False`	`False`
`lr`	1.0×10^{-4}	1.0×10^{-4}	1.0×10^{-4}
`weight_decay`	1.0×10^{-5}	1.0×10^{-5}	1.0×10^{-5}
`betas`	[0.9, 0.95]	[0.9, 0.999]	[0.9, 0.999]
`eps`	1.0×10^{-8}	1.0×10^{-8}	1.0×10^{-8}

Table 3. Hyperparameters used by Ambient Flow, Ambient UNet, and Ambient FNO.

B. Partial Differential Equations

Darcy Flow Equation. The Darcy flow equation is a second-order elliptic partial differential equation used to model the flow of viscous incompressible fluid through a porous medium. Its two-dimensional variant can be expressed as:

$$-\nabla \cdot (a(\mathbf{x}) \nabla u(\mathbf{x})) = f(\mathbf{x}), \quad \mathbf{x} \in (0,1)^2,$$

with zero boundary conditions $u|_{\partial\Omega} = 0$. Here, the coefficient $a(\mathbf{x})$ represents the spatially varying permeability of the medium and typically takes binary values to model heterogeneous materials, while $f(\mathbf{x})$ (often set to a constant) represents the forcing term. The binary nature of $a(\mathbf{x})$ introduces sharp discontinuities in permeability, and learning-based methods must represent these discontinuities while keeping the solution field physically consistent.

Inhomogeneous Helmholtz Equation. The inhomogeneous Helmholtz equation is a second-order elliptic partial differential equation used to model time-harmonic wave propagation in heterogeneous media. Its two-dimensional variant can be expressed as:

$$\nabla^2 u(\mathbf{x}) + k^2 u(\mathbf{x}) = a(\mathbf{x}), \quad \mathbf{x} \in (0,1)^2,$$

with zero boundary conditions $u|_{\partial\Omega} = 0$. Here, k denotes the wave number (typically set to $k = 1$), and $a(\mathbf{x})$ represents the spatially varying source term, often modeled as a piecewise-constant function. The Laplacian term $\nabla^2 u$ governs diffusion and the $k^2 u$ term introduces oscillations, which can yield interference patterns and local resonances. When $k = 0$, the equation reduces to the Poisson equation.

Unbounded Incompressible Navier-Stokes Equation. The incompressible Navier-Stokes equations are a system of nonlinear partial differential equations that govern the dynamics of viscous fluid flow. For the unbounded velocity formulation, the two-dimensional variant can be expressed as:

$$\begin{aligned} \partial_t w(\mathbf{x},\tau) + \mathbf{v}(\mathbf{x},\tau) \cdot \nabla w(\mathbf{x},\tau) &= \nu \Delta w(\mathbf{x},\tau) + q(\mathbf{x}), \quad \mathbf{x} \in (0,1)^2,\ \tau \in (0,T], \\ \nabla \cdot \mathbf{v}(\mathbf{x},\tau) &= 0, \end{aligned}$$

where $w = \nabla \times \mathbf{v}$ is the vorticity, $\mathbf{v}(\mathbf{x},\tau)$ is the velocity field, $q(\mathbf{x})$ is a fixed force field, and ν is the kinematic viscosity. With $\nu = 10^{-3}$, the corresponding Reynolds number $Re = 1/\nu = 1000$ places the flow in the turbulent regime, where nonlinear advection and viscous diffusion generate vortices, shear layers, and energy cascades. The divergence-free constraint $\nabla \cdot \mathbf{v} = 0$ enforces incompressibility, coupling the velocity and vorticity fields through the stream function. Learning-based methods must model the time evolution while respecting these constraints.

Poisson Equation. The Poisson equation is a second-order elliptic partial differential equation that describes steady-state diffusion processes. Its two-dimensional variant can be expressed as:

$$\nabla^2 u(\mathbf{x}) = a(\mathbf{x}), \quad \mathbf{x} \in (0,1)^2,$$

with homogeneous Dirichlet boundary conditions $u|_{\partial\Omega} = 0$. Here, $a(\mathbf{x})$ represents the source term, which often exhibits piecewise-constant structure to model localized sources or sinks in heterogeneous domains. Although the solution field is typically smooth, discontinuities in $a(\mathbf{x})$ influence $u(\mathbf{x})$ through the elliptic operator, and models must respect the boundary conditions.

C. One-Point Transition to Ambient Physics

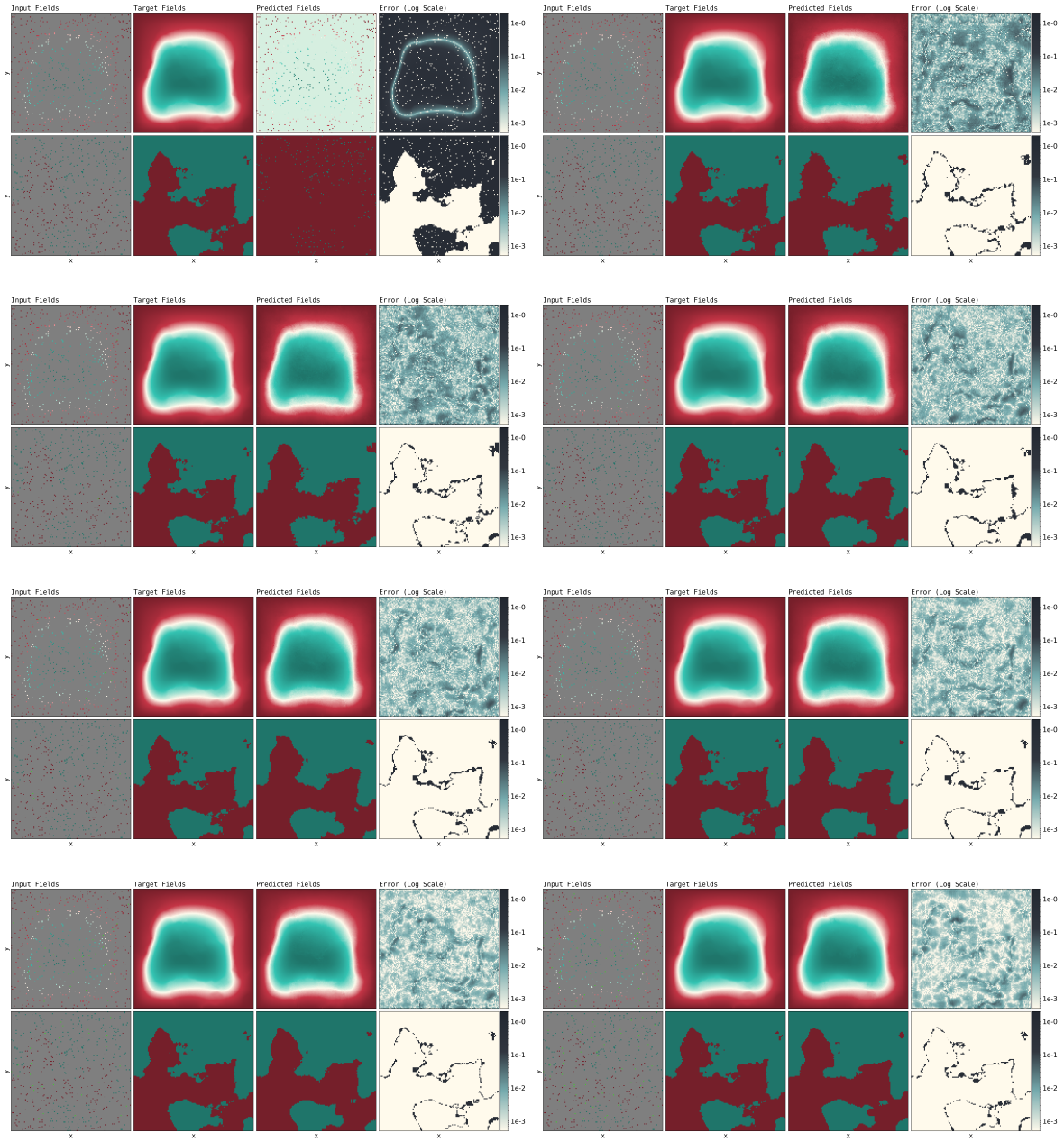

Figure 5. **Example of "One-Point Transition" (Darcy Flow).** Each subplot shows an Ambient Flow reconstruction from partial observations (3% uniformly random points), varying the number of additional masked (already-observed) points used during training (increasing across the grid; top-left: 0, which corresponds to naive training). **Within Each Subplot: Top:** Solution field. **Bottom:** Coefficient Field. **Left to Right:** Input Fields, Target Fields, Reconstructed Fields, Error (Log Scale). Within each input panel, gray pixels represent unobserved locations and green pixels represent masked measurements withheld from the model.

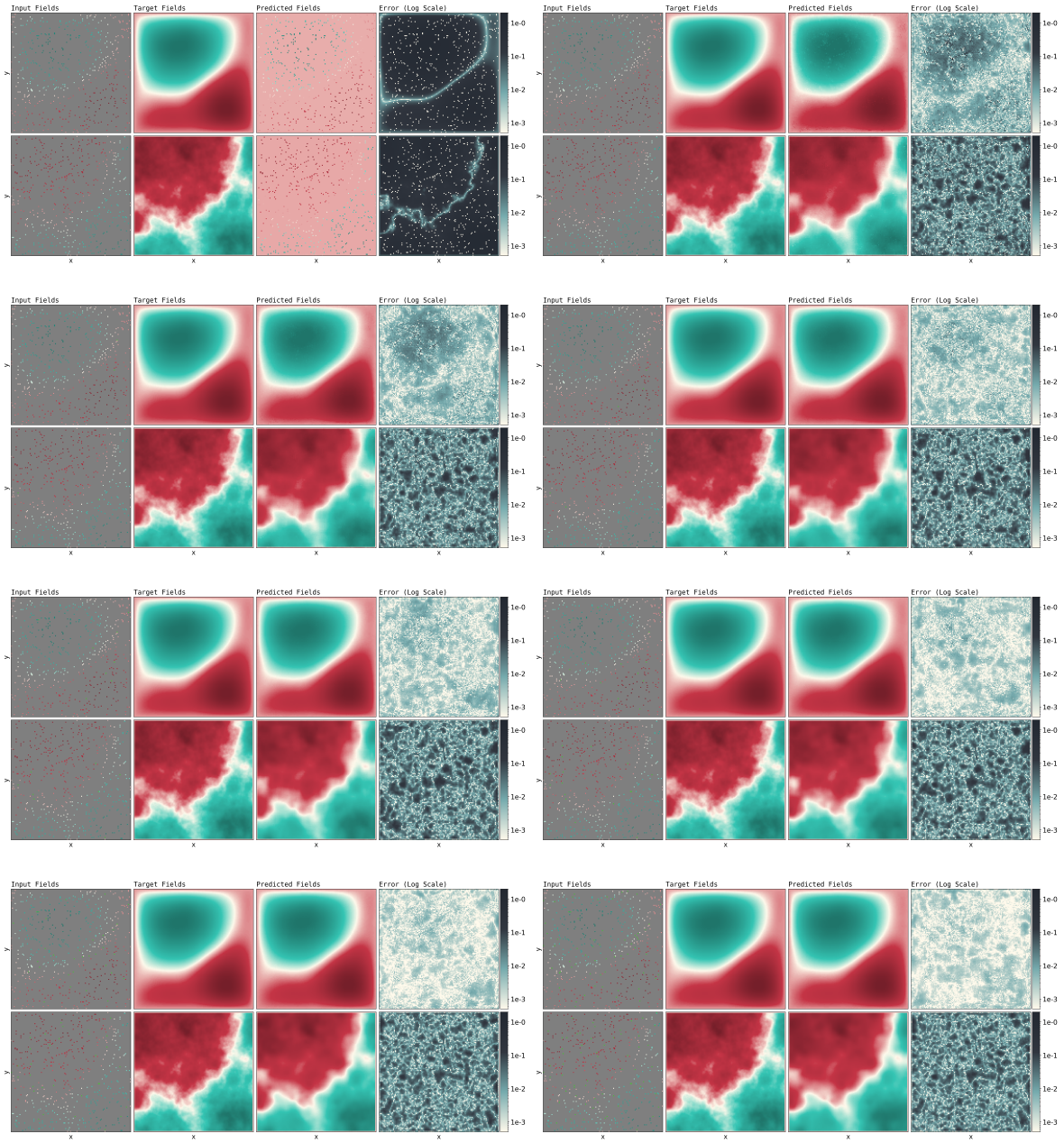

Figure 6. **Example of "One-Point Transition" (Helmholtz).** Each subplot shows an Ambient Flow reconstruction from partial observations (3% uniformly random points), varying the number of additional masked (already-observed) points used during training (increasing across the grid; top-left: 0, which corresponds to naive training). **Within Each Subplot: Top:** Solution field. **Bottom:** Coefficient Field. **Left to Right:** Input Fields, Target Fields, Reconstructed Fields, Error (Log Scale). Within each input panel, gray pixels represent unobserved locations and green pixels represent masked measurements withheld from the model.

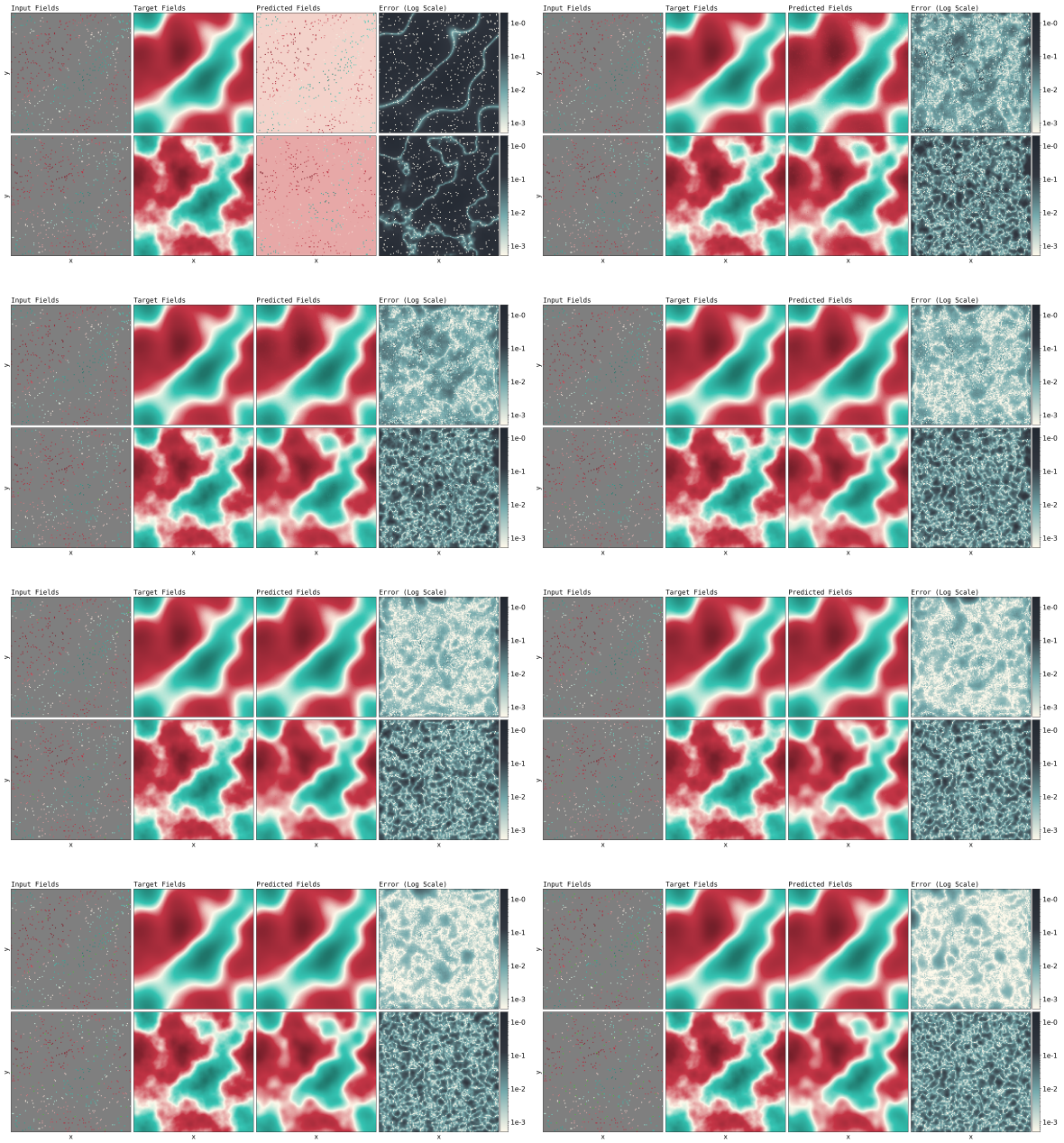

Figure 7. **Example of "One-Point Transition" (Navier-Stokes).** Each subplot shows an Ambient Flow reconstruction from partial observations (3% uniformly random points), varying the number of additional masked (already-observed) points used during training (increasing across the grid; top-left: 0, which corresponds to naive training). **Within Each Subplot: Top:** Solution field. **Bottom:** Coefficient Field. **Left to Right:** Input Fields, Target Fields, Reconstructed Fields, Error (Log Scale). Within each input panel, gray pixels represent unobserved locations and green pixels represent masked measurements withheld from the model.

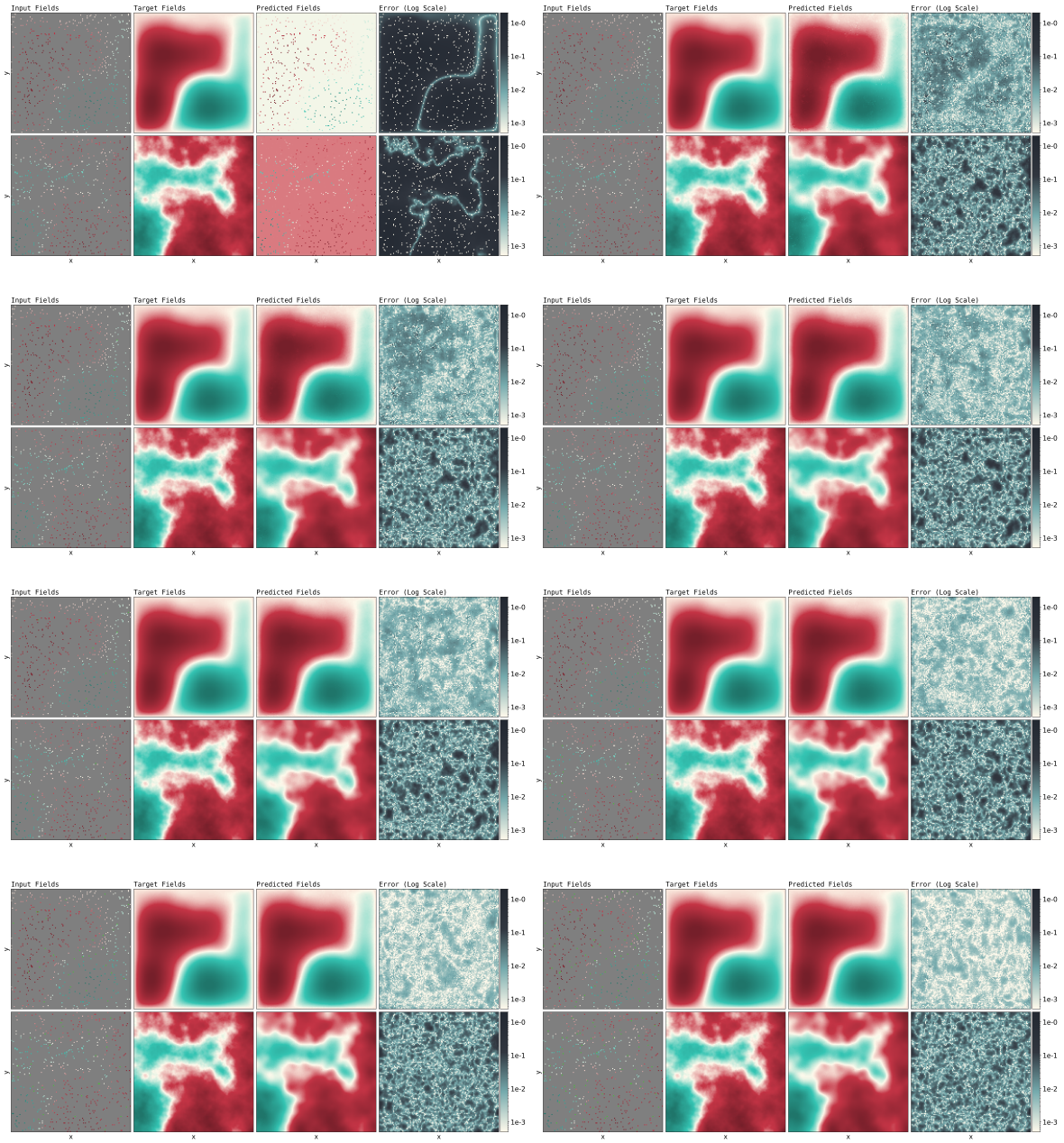

Figure 8. **Example of "One-Point Transition" (Poisson).** Each subplot shows an Ambient Flow reconstruction from partial observations (3% uniformly random points), varying the number of additional masked (already-observed) points used during training (increasing across the grid; top-left: 0, which corresponds to naive training). **Within Each Subplot: Top:** Solution field. **Bottom:** Coefficient Field. **Left to Right:** Input Fields, Target Fields, Reconstructed Fields, Error (Log Scale). Within each input panel, gray pixels represent unobserved locations and green pixels represent masked measurements withheld from the model.

D. Robustness Across Patterns and Super-Resolution

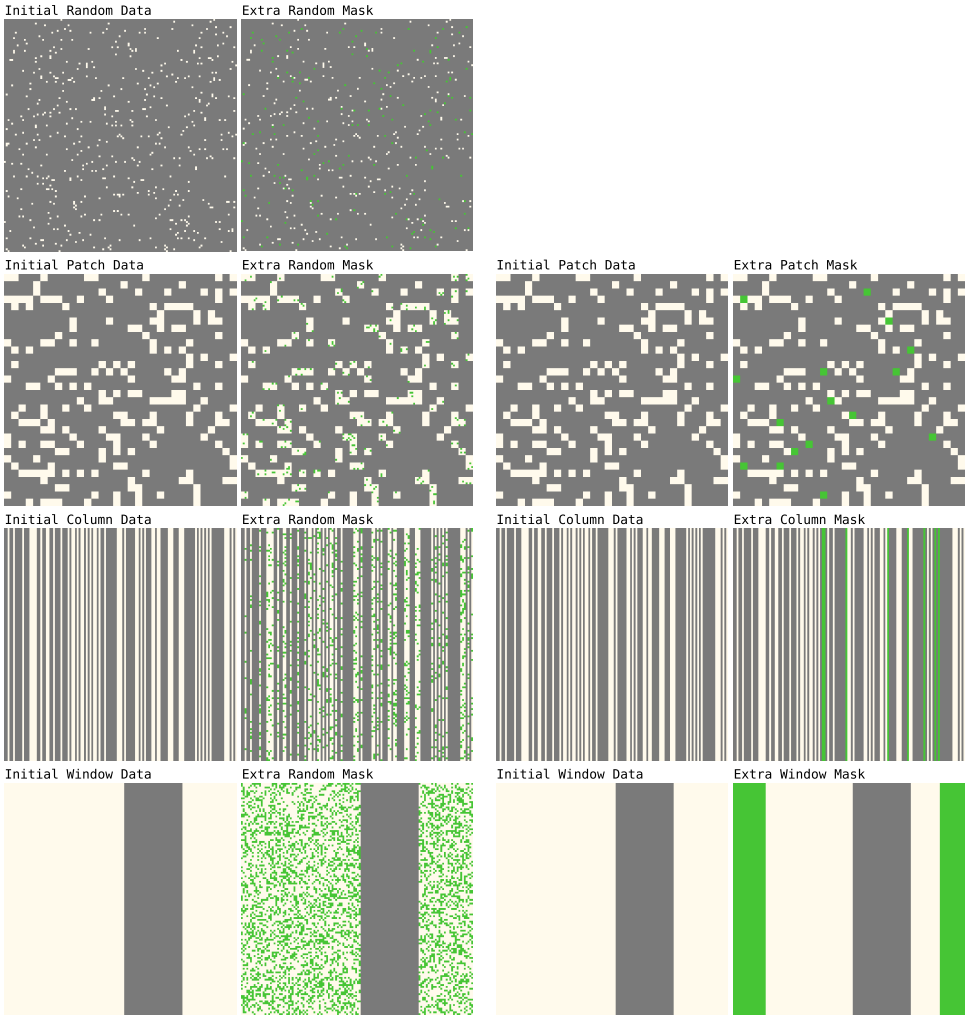

Figure 9. **Measurement patterns and extra masking operators.** Visualization of measurement patterns A and additional masking operators B used during Ambient Physics training. Additional masking B may match the measurement pattern (pattern-matched) or differ from it (mismatched); when mismatched, stronger additional masking is typically required to achieve similar training effects. **Top to bottom:** Random, Patch, Column, Window. Gray pixels represent unobserved locations and green pixels represent masked measurements withheld from the model.

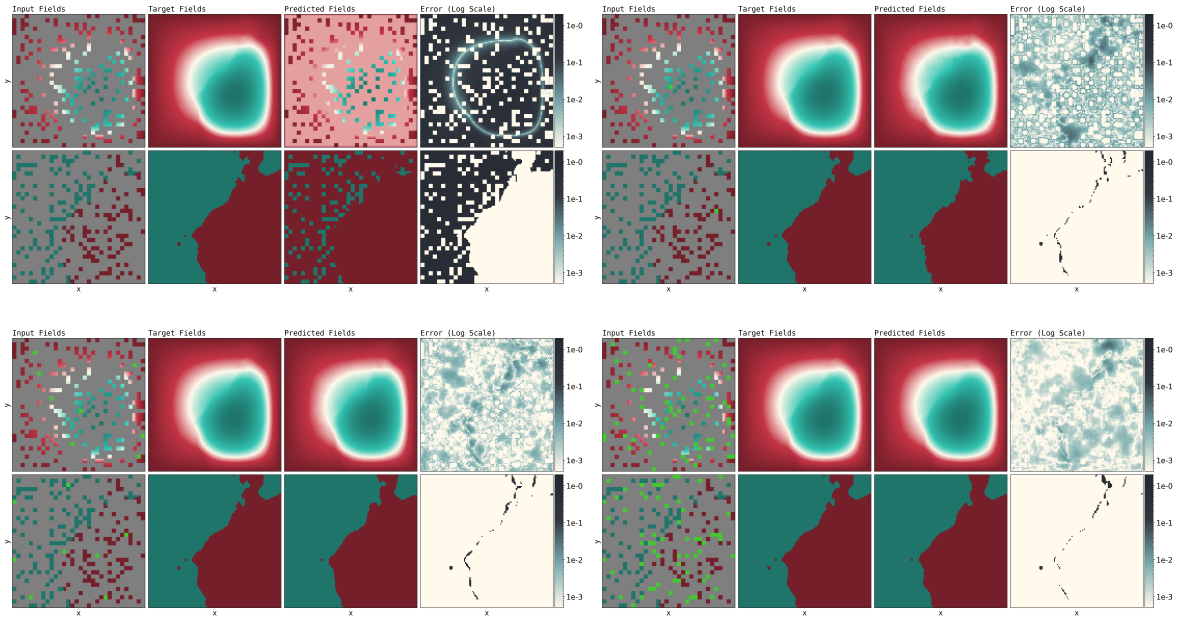

Figure 10. **Example of Patch Measurements (Darcy Flow).** Each subplot shows an Ambient Flow reconstruction from partial observations (20% uniformly random patches), varying the number of additional masked (already-observed) patches used during training (increasing across the grid; top-left: 0, which corresponds to naive training). **Within Each Subplot: Top:** Solution field. **Bottom:** Coefficient Field. **Left to Right:** Input Fields, Target Fields, Reconstructed Fields, Error (Log Scale). Within each input panel, gray pixels represent unobserved locations and green pixels represent masked measurements withheld from the model.

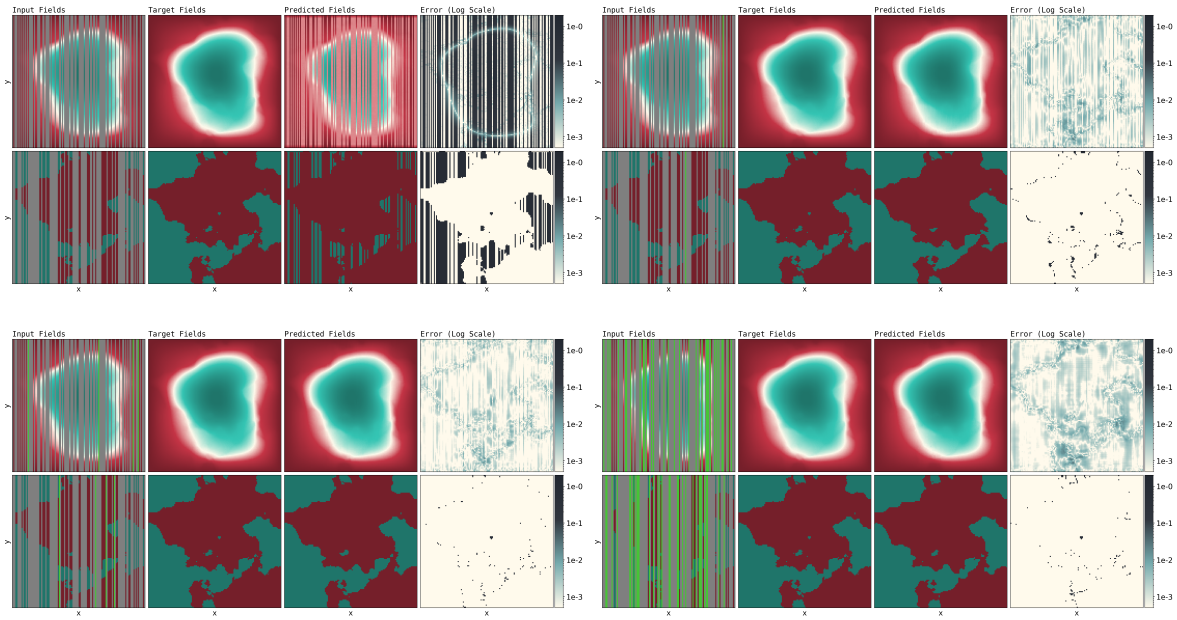

Figure 11. **Example of Column Measurements (Darcy Flow).** Each subplot shows an Ambient Flow reconstruction from partial observations (40% uniformly random columns), varying the number of additional masked (already-observed) columns used during training (increasing across the grid; top-left: 0, which corresponds to naive training). **Within Each Subplot: Top:** Solution field. **Bottom:** Coefficient Field. **Left to Right:** Input Fields, Target Fields, Reconstructed Fields, Error (Log Scale). Within each input panel, gray pixels represent unobserved locations and green pixels represent masked measurements withheld from the model.

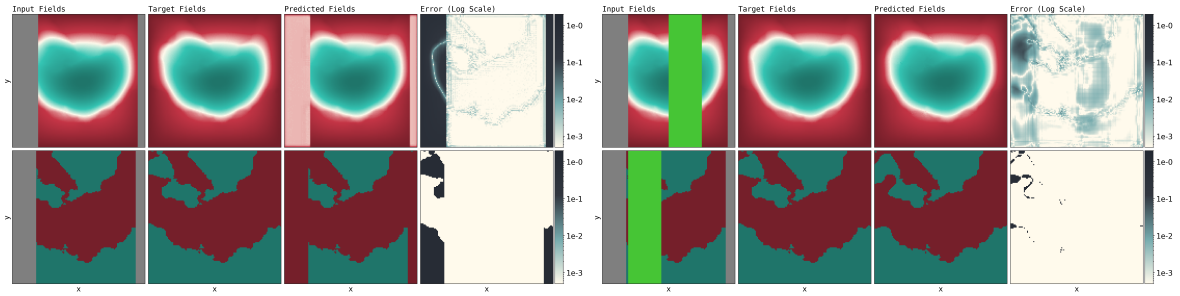

Figure 12. **Example of Window Measurements (Darcy Flow).** Each subplot shows an Ambient Flow reconstruction from partial observations (75% contiguous window). **Left:** No additional mask (naive training). **Right:** One additional mask.**Within Each Subplot: Top:** Solution field. **Bottom:** Coefficient Field. **Left to Right:** Input Fields, Target Fields, Reconstructed Fields, Error (Log Scale). Within each input panel, gray pixels represent unobserved locations and green pixels represent masked measurements withheld from the model.

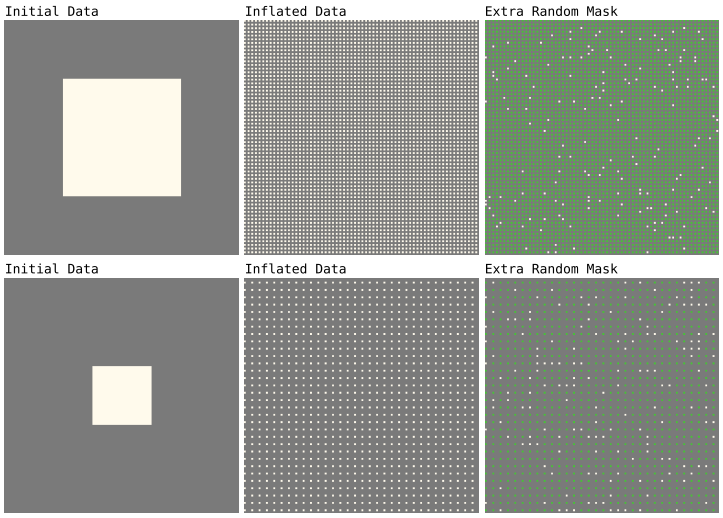

Figure 13. **Low-Resolution Observations as Partial Measurements for Super-Resolution.** A low-resolution measurement can be interpreted as a partially observed high-resolution field by inflating (upsampling) the observed values onto a higher-resolution (possibly shifted) grid. However, the inflated observations lie on a regular checkerboard-like lattice; if supervised only on these fixed locations, a model can learn to be accurate on the lattice while predicting unreliably elsewhere. We therefore apply an additional random mask over the inflated grid. Crucially, this mask must be sufficiently strong to hide the lattice structure, so the supervised locations no longer reveal a fixed checkerboard pattern and the model is encouraged to produce coherent high-resolution predictions across the domain. Ambient Physics can then learn a generative prior over high-resolution fields directly from low-resolution observations and perform super-resolution (see Table 4).

Initial Resolution	Extra Masked	Total Unobserved	Darcy Flow		Helmholtz		Navier-Stokes		Poisson	
			Coefficients	Solutions	Coefficients	Solutions	Coefficients	Solutions	Coefficients	Solutions
64×64	22.00%	97.00%	3.57%	2.20%	9.55%	19.34%	8.23%	2.28%	10.63%	37.20%
	23.00%	98.00%	**1.55%**	1.57%	**8.67%**	6.04%	**7.40%**	**1.71%**	**9.77%**	3.65%
	24.00%	99.00%	2.26%	**1.13%**	10.02%	**1.60%**	9.32%	2.48%	11.39%	**1.47%**
	24.50%	99.50%	3.43%	1.36%	14.29%	3.06%	14.82%	6.28%	14.85%	1.59%
32×32	3.25%	97.00%	1.72%	5.31%	24.74%	16.22%	14.41%	5.40%	27.33%	23.96%
	4.25%	98.00%	3.05%	3.17%	14.25%	6.00%	**8.77%**	1.50%	14.12%	5.83%
	5.25%	99.00%	**2.39%**	**1.82%**	**10.82%**	2.34%	9.05%	**1.46%**	**11.27%**	2.29%
	5.75%	99.50%	3.48%	1.91%	13.73%	**1.92%**	14.38%	2.51%	13.89%	**2.05%**

Table 4. **Super-resolution ablation on additional random masking.** Relative L_2 error (%) for coefficient/solution reconstruction after inflating low-resolution observations and applying an extra random mask. Intermediate masking (about 98%–99% total unobserved) performs best, consistent with the need to obscure the checkerboard observation lattice.

	Observed Measurements	Darcy Flow		Helmholtz		Navier-Stokes		Poisson	
		Coefficient	Solution	Coefficient	Solution	Coefficient	Solution	Coefficient	Solution
Ambient Flow	25%	1.16%	0.25%	2.77%	0.18%	1.79%	0.23%	2.70%	0.19%
	20%	1.23%	0.27%	3.14%	0.19%	2.10%	0.24%	3.10%	0.20%
	15%	1.28%	0.31%	3.68%	0.23%	2.52%	0.27%	3.61%	0.23%
	10%	1.34%	0.38%	4.59%	0.27%	3.16%	0.34%	4.61%	0.28%
	5%	1.65%	0.54%	6.39%	0.39%	4.64%	0.52%	6.47%	0.38%
	3%	2.07%	0.74%	7.67%	0.44%	6.38%	0.96%	7.81%	0.45%

Table 5. **Reconstruction Performance Under Partial Observations.** Relative L_2 error (%) between ground truth and reconstructed coefficients ($\hat{a}$) and solutions ($\hat{u}$) from partial observations $(A_a a, A_u u)$ with uniformly random measurements and a varying percentage of observed measurements. *Note: In prior work these are reported as "Inverse" for coefficients and "Forward" for solutions.* "Partial Obs. Training" indicates whether training uses partial observations (✓) or complete observations (✗); "NFE" is the number of function evaluations.

	Ambient Diffusion	**Incomplete Physics, Complete Dynamics**	**Ambient Physics**
Corruption Pattern	Pixels, Patches (boxes)	Pixels, Patches (boxes)	Pixels, Patches (boxes), Columns, Windows, Checkerboard
Super-Resolution	No	No	Yes
Model	Diffusion Only	Diffusion Only	Flow Matching + deterministic neural PDE solvers
PDE-Domain Baselines	No	No	Yes
Naive Training Failure	No	Yes	No
Reproducibility	On CelebA, CIFAR-10, AFHQ	Limited	Full (multiple datasets, nD architectures, 10 evaluation metrics, configs for all experiments)

Table 6. Caption

E. Temp