ADAPTIVE SDE INTERPOLANTS FOR CALIBRATED PROBABILISTIC PDE FORECASTING

Jorge Mifsut Benet[1,*], **Armand Kassaï Koupaï**[1], **Ramon Daniel Regueiro-Espino**[1], **Nicolas Baskiotis**[1], **Patrick Gallinari**[1,2]
[1] Sorbonne Université, CNRS, ISIR, 75005 Paris, France
[2] Criteo AI Lab, Paris, France

ABSTRACT

Neural surrogate models for PDEs increasingly rely on generative methods to produce predictive ensembles, but existing approaches typically employ fixed noising schedules, leading to poorly calibrated uncertainty, especially under autoregressive rollouts and limited inference budgets. We propose a framework for calibrated probabilistic PDE forecasting that learns an adaptive latent stochastic transport between consecutive time steps. We model the transition between latent VAE posteriors using a learned Itô SDE with state-dependent drift and diffusion, trained via closed-form conditional supervision derived from a Gaussian stochastic interpolant, and regularized with an energy distance to improve distributional fidelity. This formulation enables simulation-free training and allows the diffusion term to adapt to the local dynamical regime. Across several PDE benchmarks, our method consistently improves calibration in short-to-intermediate rollouts compared to fixed-schedule generative baselines, while maintaining comparable mean accuracy under matched compute-constrained inference budgets.

1 INTRODUCTION

Neural surrogates for partial differential equations (PDEs) produce predictions that can degrade under distribution shift (Koupaï et al., 2024); and small one-step errors can accumulate when forecasts are iterated forward in time (Kovachki et al., 2024; Lippe et al., 2023). These effects are particularly present in chaotic regimes. For this reason, predictions should be accompanied by well calibrated uncertainty estimates, that indicate when a surrogate is reliable. A way to address uncertainty quantification is to have methods that parameterize a predictive probability distribution within a single model (Bülte et al., 2025).

Generative models provide a route to model probabilistic distributions via sampling. The success of diffusion models (Ho et al., 2020; Song et al., 2021) and of their continuous-time counterparts, flow matching (Lipman et al., 2023) and stochastic interpolation (Albergo et al., 2023), has motivated their adoption for probabilistic PDE forecasting and operator learning (Rühling Cachay et al., 2023; Price et al., 2024; Li et al., 2025; Zhou et al., 2025). However, generative models are trained to match the data distribution rather than to produce well-calibrated predictive intervals (Sajjadi et al., 2018). The issue is amplified in conditional generation, where the model may rely too much on the conditioning input state, and quickly collapse toward a conditional mean, especially in a few-step settings (Dhariwal & Nichol, 2021; Ho & Salimans, 2021; Bortoli et al., 2025). This often results in overconfident generative surrogates.

Physical systems can showcase a diversity of regimes and the appropriate level and structure of predictive uncertainty can therefore change substantially with the system state. Despite this, classical stochastic interpolants (SI) cannot express such heteroscedastic, regime-dependent behavior, due to their fixed diffusion coefficient (Albergo et al., 2023). As a result, the induced conditional distribution is misspecified and the forecasts become systematically miscalibrated. We propose to augment classical SI with an adaptive diffusion term that depends on the system state, leading to an improved modeling of the uncertainty.

[*]Corresponding author: mifsutbenet[at]isir.upmc.fr

To instantiate this idea, we introduce a framework for calibrated probabilistic transport between the latent posteriors of consecutive PDE states. Leveraging a Variational Autoencoder (VAE), we model the transition between consecutive latent time steps via a learned Itô SDE. Crucially, unlike standard approaches, we parameterize both the drift and diffusion terms with neural networks.

2 METHOD

Our method learns the drift and state-dependent diffusion coefficient of an Itô SDE over an auxiliary time $\tau \in [0, 1]$ that transports samples between VAE posteriors of consecutive PDE states.

2.1 OBJECTIVE: LATENT POSTERIOR TRANSPORT VIA A STOCHASTIC BRIDGE

A common setting for neural PDE surrogates is to learn the time evolution model in a reduced latent space, where PDE states are first compressed into a lower-dimensional representation. We use a VAE, where the encoder E_ϕ maps PDE solution fields $u \in \mathbb{R}^{d_u}$ to a diagonal Gaussian posterior

$$q_\phi(z|u) = \mathcal{N}(\mu_\phi(u), \sigma^2_\phi(u)), \quad z \in \mathbb{R}^{d_z},\ d_z \ll d_u,$$

and the decoder D_ϕ defines a conditional likelihood $p_\phi(u|z)$.

Encoder posterior as target. Since z_{t+1} is unobserved, the only supervision available for the next latent state comes from the encoder posterior $q_\phi(z_{t+1} \mid u_{t+1})$. Combined with the data dynamics, this induces a target distribution over z_{t+1} given u_t, obtained by averaging the encoder posterior over likely next states. Thus, the optimal one-step predictor matches the encoder-induced conditional latent transition $\tilde{p}(z_{t+1} \mid u_t)$.

Continuous-time stochastic bridge . We realize this conditional transition as a continuous-time stochastic bridge over an auxiliary interpolation time $\tau \in [0, 1]$. We model the transition between consecutive posteriors via an Itô SDE:

$$\mathrm{d}z_\tau = f_\theta(z_\tau, \tau)\,\mathrm{d}\tau + g_{\theta'}(z_\tau, \tau)\,\mathrm{d}W_\tau, \tag{1}$$

where W_τ is a standard Wiener process. The drift f_θ and the diagonal state-dependent diffusion amplitude $g_{\theta'}$ are parameterized by neural networks. Unlike classical SI that prescribe a global diffusion schedule, we learn both drift and diffusion directly from data, and the interpolation endpoints correspond to consecutive latent posteriors.

2.2 GAUSSIAN STOCHASTIC INTERPOLANT AND ANALYTIC CONDITIONAL TARGETS

We construct a stochastic interpolant between independent endpoint samples $z_0 \sim q_\phi(z_t|u_t)$ and $z_1 \sim q_\phi(z_{t+1}|u_{t+1})$:

$$z_\tau = (1-\tau)z_0 + \tau z_1 + \sqrt{\gamma(\tau)}\,\xi, \qquad \xi \sim \mathcal{N}(0, I), \tag{2}$$

where $\gamma(\tau) = \sigma_{\text{scale}}\tau(1-\tau)$ and σ_{scale} is a tunable hyperparameter.

Learning the diffusion from endpoints alone is underdetermined and naïve objectives may admit degenerate solutions or poor conditioning, especially near $\tau \simeq 1$ where the remaining horizon $\Delta\tau = 1 - \tau$ is small. The most direct remedy would be pathwise supervision by integrating equation 1 to $\tau = 1$, but this requires differentiating through an SDE solver.

Instead, we supervise equation 1 using a *local one-step kernel* from (z_τ, τ) to the endpoint. We approximate the remaining transition $\tau \to 1$ by

$$\hat{z}_1 \mid z_\tau = z_\tau + \Delta\tau\, f_\theta(z_\tau, \tau) + \sqrt{\Delta\tau}\, g_{\theta'}(z_\tau, \tau) \odot \xi, \quad \xi \sim \mathcal{N}(0, I). \tag{3}$$

Under the above setup, let $\mu_{1|\tau}$ and $\Sigma_{1|\tau}$ denote the conditional mean and covariance of z_1 given z_τ. Then the conditional moments are given explicitly by:

$$\mu_{1|\tau} = \mu_1 + \tau\Sigma_1\Sigma_\tau^{-1}(z_\tau - \mu_\tau), \quad \Sigma_{1|\tau} = \Sigma_1 - \tau^2\Sigma_1\Sigma_\tau^{-1}\Sigma_1,$$

where $\mu_\tau = (1-\tau)\mu_0 + \tau\mu_1$ and $\Sigma_\tau = (1-\tau)^2\Sigma_0 + \tau^2\Sigma_1 + \gamma(\tau)I$. This follows from standard Gaussian conditioning. We use $(\mu_{\text{target}}, \Sigma_{\text{target}}) = (\mu_{1|\tau}, \Sigma_{1|\tau})$ as exact conditional targets at any τ. We match these conditional moments to the analytic targets from equation 2.2:

$$\mathcal{L}_{\text{drift}} = \mathbb{E}_{\tau,z_\tau}\left[\left\|\Delta\tau\, f_\theta(z_\tau,\tau) - (\mu_{\text{target}} - z_\tau)\right\|_2^2\right], \quad \mathcal{L}_{\text{var}} = \mathbb{E}_{\tau,z_\tau}\left[\left\|\Delta\tau\, g_{\theta'}(z_\tau,\tau)^2 - \Sigma_{\text{target}}\right\|_2^2\right]. \tag{4}$$

2.3 DISTRIBUTIONAL REGULARIZATION VIA ENERGY DISTANCE

To penalize distributional mismatch without integrating the SDE, we add an ED term between samples from the one-step kernel equation 3 and samples from the target conditional Gaussian $\mathcal{N}(\mu_{\text{target}}, \Sigma_{\text{target}})$:

$$\mathcal{L}_{\text{ED}} = \mathbb{E}_{\tau, z_\tau}\Big[\,\text{ED}\left(\{\hat{z}_1^{(k)}\}_{k=1}^{K}, \{\hat{z}_{1,\text{target}}^{(\ell)}\}_{\ell=1}^{L}\right)\Big], \tag{5}$$

Our final objective is

$$\mathcal{L} = \mathcal{L}_{\text{drift}} + \lambda_{\text{var}}\mathcal{L}_{\text{var}} + \lambda_{\text{ED}}\mathcal{L}_{\text{ED}}, \tag{6}$$

where λ_{var} and λ_{ED} are ramped up from 0 after a drift warm-up stage.

2.4 INFERENCE BY SDE INTEGRATION

At inference time, given u_t, we draw $z_0 \sim q_\phi(z_t \mid u_t)$ and integrate equation 1 over $\tau \in [0, 1]$ to obtain samples of z_{t+1}. Using Euler–Maruyama (EM) with $\Delta\tau = 1/N$:

$$z_{n+1}^{(k)} = z_n^{(k)} + \Delta\tau\, f_\theta(z_n^{(k)}, \tau_n) + \sqrt{\Delta\tau}\, g_{\theta'}(z_n^{(k)}, \tau_n) \odot \xi_n, \tag{7}$$

for $n = 0, \ldots, N-1$ and $\tau_n = n\Delta\tau$, with $\xi_n \sim \mathcal{N}(0, I)$. Decoding the ensemble yields a predictive distribution for u_{t+1}, and we repeat the procedure autoregressively for multi-step forecasting.

3 EXPERIMENTS

We evaluate our method's ability to produce calibrated predictive distributions under compute-constrained, few-step inference. We compare against standard generative baselines on two PDE datasets using four metrics capturing mean accuracy, distributional quality, calibration, and spread.

3.1 EXPERIMENTAL SETTING

Datasets We use 2D trajectory datasets generated with APEBench (Koehler et al., 2024) using the suite defaults. We consider Kolmogorov Flow and Kuramoto–Sivashinsky (KS) equations.

Baselines We compare our method to standard conditional generative baselines in latent space. All methods share the same trained VAE, conditioning, and latent-processor backbone, and differ only in the auxiliary-time construction, training loss, and sampler steps (from now number of forward evaluations (NFE)). **Diffusion (DDIM).** Discrete noising with $N{=}100$ steps; trained to predict added Gaussian noise. We sample with DDIM (Song et al., 2020) using a chosen number of denoising steps. **Flow Matching (FM).** Noise-to-data conditional FM (Lipman et al., 2023) with linear interpolant; trained by ℓ_2 velocity matching. We sample by Euler ODE integration with chosen NFE. **Stochastic Interpolants (SI).** Continuous-time adaptation of the implementation from (Zhou et al., 2025); trained with the standard SI drift-matching loss. We sample with EM with chosen NFE.

Metrics We report Mean Squared Error (MSE) for mean accuracy, Continuous Ranked Probability Score (CRPS) as a strictly proper score for distributional quality, Root Mean Squared Calibration Error (RMSCE) to quantify calibration, and Spread-Skill ratio (SSR) as a complementary diagnostic to detect over/under-dispersion. We report $|\log(\text{SSR})|$ so that lower values indicate better performance, for clarity. All metrics are computed in physical space and averaged over spatial grid points, channels, batch elements, and over the reported horizon range.

Architecture and experimental setting. **VAE:** All methods operate in the same latent space given by a Deep Compression Autoencoder (DCAE) (Chen et al., 2025) trained with a standard β-VAE objective (Higgins et al., 2017). We train one VAE instance for each dataset **Processor:** Our model and all baselines share a Transformer trunk P_Θ (DiT-based (Peebles & Xie, 2023)). The trunk's output branches into a a drift head f_θ and a diffusion head $g_{\theta'}$. The drift head consists in a final per-token projection followed by an unpatchify operation. The diffusion head uses the same decoder, preceded by an additional transformer block, and is only instantiated for our method.

Table 1: Interval evaluation (steps 1–20) at NFE=5. Mean ± std over the test set (lower is better). MSE/CRPS/RMSCE are scaled by $10^3/10^2/10^2$ (reported in $10^{-3}/10^{-2}/10^{-2}$ units). **Best** and second-best per dataset/metric are highlighted (green/blue).

Dataset	Model	MSE (10^{-3}) (↓)	CRPS (10^{-2}) (↓)	RMSCE (10^{-2}) (↓)	$\lvert\log(\mathrm{SSR})\rvert$ (↓)
Kolmogorov	DDIM	178 ± 4	18.84 ± 0.14	9.52 ± 0.05	0.628 ± 0.005
	FM	**107 ± 3**	16.33 ± 0.15	20.33 ± 0.03	1.494 ± 0.006
	SI	220 ± 5	22.43 ± 0.14	10.56 ± 0.06	0.653 ± 0.005
	Ours	120 ± 2	**14.53 ± 0.12**	**3.95 ± 0.06**	**0.268 ± 0.006**
KS	DDIM	33.88 ± 0.14	11.12 ± 0.02	15.051 ± 0.015	0.8773 ± 0.0014
	FM	9.138 ± 0.015	5.152 ± 0.005	5.483 ± 0.010	0.2984 ± 0.0007
	SI	9.604 ± 0.009	5.220 ± 0.003	3.378 ± 0.009	0.1417 ± 0.0003
	Ours	**7.824 ± 0.012**	**4.760 ± 0.004**	**2.509 ± 0.007**	**0.0209 ± 0.0007**

Training protocol. We train in a teacher-forcing regime using randomly sampled consecutive pairs (u_t, u_{t+1}) from training trajectories. For objectives that sample an interpolation time, we draw $\tau \sim \mathcal{U}(\tau_{\min}, 1 - \tau_{\min})$ with $\tau_{\min} > 0$ to avoid numerical issues near the endpoints.

Evaluation protocol and compute budget. We evaluate probabilistic forecasts under autoregressive rollout with a fixed ensemble size $K = 16$ and emphasize compute-constrained sampling with a small NFE=5.

3.2 Performance of predictive distributions

Table 1 reports interval metrics over rollout steps 1-20 across both PDE datasets at fixed compute budget (NFE=5). Our method consistently achieves the strongest uncertainty quality, with the lowest calibration errors (RMSCE) and spread–skill ratio $\lvert\log(\mathrm{SSR})\rvert$, while remaining competitive in mean accuracy (MSE) and CRPS. These results match the intended purpose of the method.

Flow matching is consistently strong in mean accuracy, but it is systematically miscalibrated. Its large $\lvert\log(\mathrm{SSR})\rvert$ values, together large RMSCE, indicate under-dispersed predictive ensembles, i.e., the model concentrates too much probability mass around the ensemble mean. This aligns with prior observations that flow-matching-style training can behave primarily as a conditional mean matcher in low-step forecasting regimes, rather than faithfully representing the full predictive distribution.

By contrast, the remaining baselines do not exhibit a uniform behavior across scenarios. DDIM and SI show mixed behavior across PDEs, suggesting that fixed schedules/noise parameterizations interact strongly with the dataset's intrinsic stochasticity and error growth.

4 Conclusion

We introduced a probabilistic neural PDE surrogate that learns a drift and a state-dependent diffusion coefficient of a latent Itô stochastic bridge that transports between consecutive VAE encoder posteriors. In contrast to fixed-schedule diffusion, flow-matching, or classical stochastic interpolants, this allows our method to have a learnable uncertainty mechanism. Across several PDE benchmarks and under matched compute-constrained inference budgets, we produce predictive ensembles with substantially improved calibration and spread–skill agreement, while remaining competitive in mean accuracy.

We acknowledge as limitation that our method's uncertainty calibration learning leverages the VAE posterior, yet standard β-VAEs are not optimized for calibrated UQ under shift or chaotic rollouts. A possible next step is developing UQ-aware VAEs. Additionally, error accumulation under autoregressive rollout is favored by the teacher forcing, one-step setting. This motivates future work to develop multi-step training strategies to learn uncertainty growth under compounding error.

ACKNOWLEDGEMENTS

We acknowledge the financial support provided by DL4CLIM (ANR-19-CHIA-0018-01), DEEPNUM (ANR-21-CE23-0017-02), PHLUSIM (ANR-23-CE23-0025-02), MELISSA (ANR-24-CE23-7140-01), PEPR Sharp (ANR-23-PEIA0008, ANR, FRANCE 2030), and cluster PostGenAI@Paris.

This project was provided with computer and storage resources by GENCI at IDRIS thanks to the grants 2025-AD011015481, and 2025-A0191016890 on the supercomputer Jean Zay's V100/A100/H100 partitions.

REFERENCES

Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions, 2023. URL https://arxiv.org/abs/2303.08797.

Valentin De Bortoli, Alexandre Galashov, J Swaroop Guntupalli, Guangyao Zhou, Kevin Patrick Murphy, Arthur Gretton, and Arnaud Doucet. Distributional diffusion models with scoring rules. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=N82967FcVK.

Christopher Bülte, Philipp Scholl, and Gitta Kutyniok. Probabilistic neural operators for functional uncertainty quantification, 2025. URL https://arxiv.org/abs/2502.12902.

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.

Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat GANs on image synthesis. 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=AAWuCvzaVt.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. β-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Sy2fzU9gl.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL https://openreview.net/forum?id=qw8AKxfYbI.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

Felix Koehler, Simon Niedermayr, Rüdiger Westermann, and Nils Thuerey. Apebench: A benchmark for autoregressive neural emulators of pdes. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 120252–120310. Curran Associates, Inc., 2024. doi: 10.52202/079017-3822. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/d9875ebcf74bccdc5076acab0dbee62c-Paper-Datasets_and_Benchmarks_Track.pdf.

Armand Kassaï Koupaï, Jorge Mifsut Benet, Yuan Yin, Jean-Noël Vittaut, and Patrick Gallinari. Geps: Boosting generalization in parametric pde neural solvers through adaptive conditioning. *Advances in Neural Information Processing Systems*, 38, 2024.

Nikola B. Kovachki, Samuel Lanthaler, and Andrew M. Stuart. Chapter 9 - operator learning: Algorithms and analysis. In Siddhartha Mishra and Alex Townsend (eds.), *Numerical Analysis Meets Machine Learning*, volume 25 of *Handbook of Numerical Analysis*, pp. 419–467. Elsevier, 2024.

doi: https://doi.org/10.1016/bs.hna.2024.05.009. URL https://www.sciencedirect.com/science/article/pii/S1570865924000097.

Zijie Li, Saurabh Patil, Francis Ogoke, Dule Shu, Wilson Zhen, Michael Schneier, John R. Buchanan, and Amir Barati Farimani. Latent neural pde solver: A reduced-order modeling framework for partial differential equations. *Journal of Computational Physics*, 524:113705, 2025. ISSN 0021-9991. doi: https://doi.org/10.1016/j.jcp.2024.113705. URL https://www.sciencedirect.com/science/article/pii/S0021999124009537.

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.

Phillip Lippe, Bastiaan S. Veeling, Paris Perdikaris, Richard E. Turner, and Johannes Brandstetter. Pde-refiner: Achieving accurate long rollouts with neural pde solvers, 2023. URL https://arxiv.org/abs/2308.05732.

William Peebles and Saining Xie. Scalable diffusion models with transformers, 2023. URL https://arxiv.org/abs/2212.09748.

Ilan Price, Alvaro Sanchez-Gonzalez, Ferran Alet, Tom R. Andersson, Andrew El-Kadi, Dominic Masters, Timo Ewalds, Jacklynn Stott, Shakir Mohamed, Peter Battaglia, Remi Lam, and Matthew Willson. Probabilistic weather forecasting with machine learning. *Nature*, 637 (8044):84–90, December 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-08252-9. URL http://dx.doi.org/10.1038/s41586-024-08252-9.

Salva Rühling Cachay, Bo Zhao, Hailey Joren, and Rose Yu. Dyffusion: A dynamics-informed diffusion model for spatiotemporal forecasting. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 45259–45287. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/8df90a1440ce782d1f5607b7a38f2531-Paper-Conference.pdf.

Mehdi SM Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. *Advances in neural information processing systems*, 31, 2018.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv:2010.02502*, October 2020. URL https://arxiv.org/abs/2010.02502.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021. URL https://arxiv.org/abs/2011.13456.

Anthony Zhou, Alexander Wikner, Amaury Lancelin, Pedram Hassanzadeh, and Amir Barati Farimani. Reframing generative models for physical systems using stochastic interpolants, 2025. URL https://arxiv.org/abs/2509.26282.