Variational Grey-Box Dynamics Matching

Gurjeet Sangra Singh
University of Geneva
HES-SO Geneva

Frantzeska Lavda
HES-SO Geneva

Giangiacomo Mercatali
HES-SO Geneva

Alexandros Kalousis
HES-SO Geneva

Abstract

Deep generative models such as flow matching and diffusion models have shown great potential in learning complex distributions and dynamical systems, but often act as black-boxes, neglecting underlying physics. In contrast, physics-based simulation models described by ODEs/PDEs remain interpretable, but may have missing or unknown terms, unable to fully describe real-world observations. We bridge this gap with a novel grey-box method that integrates incomplete physics models directly into generative models. Our approach learns dynamics from observational trajectories alone, without ground-truth physics parameters, in a simulation-free manner that avoids scalability and stability issues of Neural ODEs. The core of our method lies in modelling a structured variational distribution within the flow matching framework, by using two latent encodings: one to model the missing stochasticity and multi-modal velocity, and a second to encode physics parameters as a latent variable with a physics-informed prior. Furthermore, we present an adaptation of the framework to handle second-order dynamics. Our experiments on representative ODE/PDE problems and real-world weather forecasting demonstrate that our method performs on par with or superior to fully data-driven approaches and previous grey-box baselines, while preserving the interpretability of the physics model. Our code is available at `https://github.com/DMML-Geneva/VGB-DM`.

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300. Copyright 2026 by the author(s).

1 Introduction

Modelling the dynamics of physical systems is a fundamental challenge in various scientific domains, e.g., in fluid dynamics, geophysics and others. Such systems are described by physics models, often formulated as Ordinary or Partial Differential Equations (ODEs/PDEs). Because they are derived from physical laws such as conservation of energy, momentum, or mass, these models provide interpretable parameters and equations whose terms have clear physical meaning. However, real-world systems often exceed the complexity captured by known physics, resulting in incomplete models with unknown parameters or missing terms.

Deep generative models, by contrast, have shown remarkable success in learning complex data distributions. Techniques such as Neural ODEs [Chen et al., 2018] or Flow Matching [Lipman et al., 2023, Albergo et al., 2023, Liu et al., 2023] can model high-dimensional distributions effectively. Applied to dynamical systems, however, these methods act as black boxes; they learn the dynamics purely from observations without leveraging existing physical knowledge. This can lead to physically implausible predictions, poor generalization, and limited interpretability.

To bridge this gap, deep grey-box modelling approaches aim to integrate incomplete physics knowledge with data-driven methods. Different simulation-based methods, which leverage Neural ODEs [Chen et al., 2018], such as Physics-Integrated Variational Autoencoders (Phys-VAE) [Takeishi and Kalousis, 2021] and related methods [Yin et al., 2021, Mehta et al., 2021, Qian et al., 2021, Wehenkel et al., 2023, Verma et al., 2024] combine physics models with neural networks. While effective, these methods rely heavily on numerical ODE solvers during training, which can introduce scalability issues, numerical instability, and significant computational overhead.

Furthermore, learning the dynamics of physical sys-

tems often involves dealing with inherent stochasticity or multi-modal velocity fields. When physics is incomplete or parameters are unknown, multiple valid future trajectories might exist from the same state. Standard deterministic approaches struggle to capture this complexity. Recent advances in Variational Rectified Flow Matching [Guo and Schwing, 2025] highlight the importance of introducing stochasticity via latent variables to learn more expressive and accurate vector fields, arguing that this is essential for capturing complex dynamics.

In this work, we introduce Variational Grey-Box Dynamics Matching (VGB-DM)[1], a novel approach that integrates incomplete physics models into a simulation-free generative modelling paradigm while explicitly accounting for stochasticity and parameter uncertainty. Our approach builds upon the simulation-free methods, adapting them to learn dynamics directly from trajectories [Zhang et al., 2024]. We employ a variational approach inspired by [Guo and Schwing, 2025], but introduce a structural latent space tailored for grey-box modelling, one part represents random variability in the system's dynamics, while another part represents physical parameters that must be inferred from data.

Our contributions are summarized as follows:

1. Simulation-Free Grey-Box modelling: We propose a novel integration of incomplete physics models in a generative model inspired by gradient matching as used in recent Flow Matching generative models. This approach learns dynamics and infers parameters in a simulation-free manner, avoiding the computational bottlenecks, memory scalability and stability issues associated with solver-based methods like Neural ODEs, enhancing scalability for high-dimensional grey-box modelling.

2. Structured Variational Inference: We introduce a structured variational distribution over the velocity field, utilising separate latent variables for modelling stochasticity/multi-modality and inferring physics parameters. This allows the model to capture complex dynamics while maintaining physics interpretability.

3. Empirical Validation: We demonstrate the effectiveness of VGB-DM on different ODE/PDE systems and weather forecasting task, showing superior performance and faster convergence compared to state-of-the-art baselines.

[1] We have open-sourced our VGB-DM model and experiments in our VGB-DM GitHub repository

2 Background

2.1 Problem setting

Consider an unknown data-generating process (DGP) underlying the real-world dynamical system. We observe data as a collection of trajectories $\mathcal{D} = \{\boldsymbol{x}^i\}_{i=1}^N$. Each sample $\boldsymbol{x}^i$ corresponds to a time series, $\boldsymbol{x}^i = [x_0^i, \ldots, x_T^i]$, where $x_t^i \in \mathcal{X} \subseteq \mathbb{R}^d$ is the state at any discretized time $t \in \{0, \ldots, T\}$. These trajectories are assumed to be independent and identically distributed (i.i.d.) samples from an unknown *data* distribution $\pi(\boldsymbol{x}) \in \mathcal{P}(\mathcal{X}^{T+1})$, where $\mathcal{P}(\mathcal{X}^{T+1})$ is a set of all probability measures over the trajectory space $\mathcal{X}^{T+1}$.

A central goal in learning dynamical systems with generative models is to construct a conditional model that, given a current state x_t and possibly an arbitrary history of past observations $\boldsymbol{x}_{t-h:t}$ with $h \in \mathbb{Z}^+$, can generate plausible future trajectories. Such a model should forecast consistently by learning the underlying system evolution, including extrapolation beyond observed horizons, resulting into approximating the true conditional distribution $\pi(\boldsymbol{x} \mid \boldsymbol{x}_{t-h:t})$.

2.2 Grey-Box Modelling: Beyond Forecasting

Within the problem description given earlier, grey-box modelling approaches have demonstrated capabilities in generating robust dynamics and physics parameter inference [Takeishi and Kalousis, 2021, Wehenkel et al., 2023]. In contrast to purely data-driven forecasting, the grey-box setting assumes the availability of an approximate but useful physics model. This introduces an additional goal: not only generating consistent system dynamics, but also inferring underlying physics parameters from observed trajectories. Making them well-suited for real-world scenarios where physics knowledge is incomplete but valuable. In this framework, the physics model typically comes in the form of ordinary or partial differential equations (ODEs/PDEs): $\frac{\partial x_t}{\partial t} = f_p(x_t, \theta)$, where $f_p : \mathcal{X} \times \Theta \mapsto \mathbb{R}^d$ is the known (but potentially incomplete) physics model and $\theta \in \Theta \subseteq \mathbb{R}^p$ represents some interpretable physics parameters, often inferred. Deep grey-box models learn the time evolution of the dynamic system by leveraging some physics model by augmenting it with some Deep Neural Network (DNN): $\frac{\partial x_t}{\partial t} = f_\phi(x_t) \circ f_p(x_t, \theta)$, where $f_\phi : \mathcal{X} \mapsto \mathbb{R}^d$ is a DNN learned from data, and the operand $\circ$ the function composition operator (even if the additive form, $f_\phi(x_t) + f_p(x_t, \theta)$, is prevalent in the literature, we adopt a more general notation here). For example, in an RLC circuit, the dynamics of the grey-box can

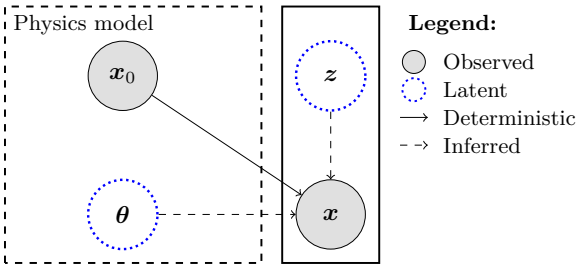

Figure 1: Probabilistic graphical model of variational grey-box models. Shaded circles are observed variables, empty dashed circles are latent variables, and dashed arrows indicate learned inference dependencies.

be described by:

$$\frac{\partial}{\partial t}\left[I_t \; U_t\right] = \underbrace{\left[\tfrac{I_t}{C} \;\; \tfrac{1}{L}(V_t - U_t)\right]}_{f_p} + \underbrace{\left[0 \;\; -\tfrac{R}{L} I_t\right]}_{f_\phi},$$

where f_p represents the known capacitor and inductor dynamics with $\theta = [L, C]$, and f_ϕ to be learned by the DNN, which captures the missing physics and the unknown resistance term.

However, a critical challenge arises from this formulation: in real-world systems, only state trajectories are observed, making the usage of the physics model more challenging, where physics' parameters are not paired. Estimating these parameters constitutes a challenging inverse problem due to potential ill-posedness and data limitations. To address this, generative grey-box models (e.g. [Takeishi and Kalousis, 2021, Wehenkel et al., 2023, Verma et al., 2024] solutions) embed inference mechanism, such as encoders that ground latent spaces in physical principles. This structure is illustrated in the probabilistic graphical model in Figure 1. Therefore, the goal in grey-box modelling extends beyond forecasting dynamics to include inference of the underlying physics parameters from observed data, while validating their meaningful usage within the model.

2.3 Flow Matching

Simulation-free generative models such as Flow Matching variant algorithms [Lipman et al., 2023, Albergo et al., 2023, Liu et al., 2023] have achieved great success in deep generative models. Given a known source distribution $\pi_0(x_0)$ (typically a standard Gaussian) and a target data distribution $\pi_1(x_1)$, flow matching algorithms [Lipman et al., 2023, Albergo et al., 2023, Liu et al., 2023] learn a time-dependent vector field $v_t^\phi : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$ that defines a continuous transformation of samples from $x_0 \sim \pi_0$ to $x_1 \sim \pi_1$. This transformation is governed by the continuity equation:

$$\frac{\partial p_t^\phi(x_t)}{\partial t} = -\operatorname{div}(p_t^\phi(x_t) v_t^\phi(x_t)),$$

where p_t is the marginal density at time $t \in [0,1]$. Instead of solving this equation through expensive simulations of forward dynamics, Conditional Flow Matching (CFM) learns directly v_t^ϕ by regressing it to a target velocity field by constructing a path of tractable conditional probabilities $p_t(x_t|x_1)$ defined along a linear interpolation from source samples to the data-target samples distribution, $x_t = (1-t)x_0 + tx_1$, with $(x_0, x_1) \sim \pi(x_0, x_1)$ and $\dot{x}_t = \frac{d}{dt}x_t = x_1 - x_0$. The training loss is then:

$$\mathcal{L}_{\text{CFM}}(\phi) = \mathbb{E}_{\substack{x_0 \sim \pi_0, x_1 \sim \pi_1 \\ t \sim \mathcal{U}(0,1)}} \left[\left\| v_t^\phi(x_t) - \dot{x}_t \right\|^2 \right], \qquad (1)$$

yielding a simulation-free objective. While linear interpolation is most common, Flow Matching algorithms also admit more general deterministic or stochastic paths, which influence the bias–variance tradeoffs of the estimator [Albergo et al., 2023, Liu et al., 2023].

3 Method

We now present multiple building block choices and modelling aspects of our algorithm, yielding to a generative model specifically designed for solving dynamical systems, which integrates physics priors in the form of physics differential equations fully trained in a simulation-free manner, and physics structured latent variables modelled by a variational distribution. The latter models physics latent variables and accounts for missing stochasticity. We denote this method as Variational Grey-Box Dynamic Matching (VGB-DM), as our goal is to learn dynamical systems by integrating physics-based components and grounding our generative model on the physics.

3.1 Variational Grey-Box Dynamics Matching

Integrating Incomplete Physics. Contrary to Conditional Flow Matching, which addresses generative tasks by mapping samples from noise to data samples, our setting focuses on dynamical systems where data naturally comes as time-series trajectories. We exploit the structure of the trajectory to optimize a target velocity vector field by building it on consecutive states of the trajectory, in the spirit of Trajectory Flow Matching (TFM) [Zhang et al., 2024], but extend it to incorporate incomplete physics f_p, which encode the underlying dynamics of the system.

More formally, let $\pi(\boldsymbol{x})$ denote the data distribution over trajectories $\boldsymbol{x}^i = (x_0^i, x_1^i, \ldots, x_T^i)$,

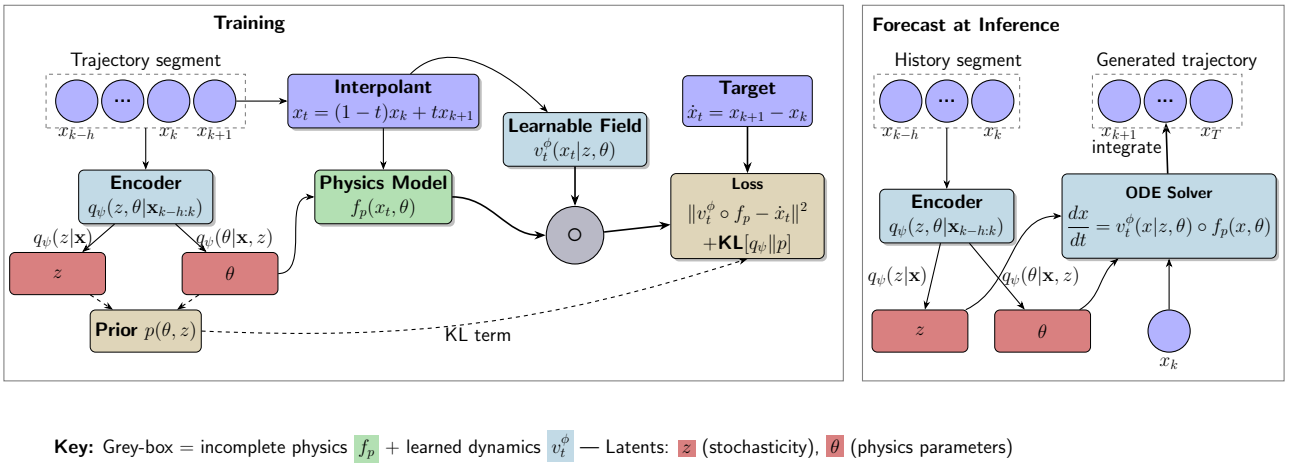

Figure 2: Overview of the Variational Grey-Box Dynamics Matching (VGB-DM) framework.

and by $\boldsymbol{x}_{k-h:k+1} \sim \pi(\cdot|\boldsymbol{x})$ the distribution over $(x_{k-h}, \ldots, x_k, x_{k+1})$ pairs of consecutive points induced by the trajectory with $k \in \mathcal{U}\{h, \ldots, T-1\}$, and $T > h$ ensures sufficient history. The objective optimizes v_t^ϕ by regressing a target velocity vector field $\dot{x}_t$ constructed on paired consecutive points, explicitly: where the target velocity field is constructed via linear interpolation in normalized time between current and next points: $x_t = (1-t)x_k + tx_{k+1}$, thus $\dot{x}_t = \frac{d}{dt}x_t = x_{k+1} - x_k$ for $t \in [0,1]$. Incorporating the physics model f_p into the model and objective yields the following GB-DM loss:

$$\mathcal{L}_{\text{GB-DM}}(\phi) = \mathbb{E}_{\substack{\boldsymbol{x}\sim\pi(\boldsymbol{x}) \\ x_{k-h:k+1}\sim\pi(\cdot|\boldsymbol{x}) \\ t\sim\mathcal{U}(0,1)}} \left\| (v_t^\phi(x_t \mid \boldsymbol{x}_{k-h:k}) \circ f_p(x_t, \cdot)) - \dot{x}_t \right\|^2, \tag{2}$$

where x_t describes the interpolant built between data points, and $\dot{x}_t$ is the associated time derivative [2]. The physics model f_p is combined with a learnable vector field v_t^ϕ, representing the missing dynamics f_ϕ, effectively learning to complete the physics dynamics by aligning the combined field with the target velocity constructed by the data.

Structured Variational Inference. A key challenge in incorporating the physics model $f_p(x_t, \theta)$ is that the physics parameters $\theta \in \Theta \subseteq \mathbb{R}^p$ are not provided in the data. Furthermore, purely deterministic physics models often fail to capture stochasticity or multimodal-behaviour that exists in many dynamic systems. This multi-modality manifests in a velocity vector field as multiple plausible flow directions at a single point in state-space. This is a common phenomenon in grey box modelling where unknown physics can lead to divergent trajectories from identical initial conditions.

To address these, we introduce a structured latent variable approach. We introduce a latent variable θ to represent the physics parameters and a second latent variable $z \in \mathcal{Z} \subseteq \mathbb{R}^h$ to capture the missing stochasticity and multi-modal behaviour in the velocity vector field.

We model the distribution of the vector field variationally by introducing a variational posterior conditioned on the trajectory segment $q_\psi(z, \theta \mid \boldsymbol{x}_{k-h:k})$. A probabilistic schema of this latent variable structure is provided in Figure 1.

$$\begin{aligned} \log p_t^\phi(\dot{x}_t \mid x_k) &= \log \int p_t^\phi(\dot{x}_t, \theta, z \mid x_k)\, d\theta\, dz \\ &= \log \int q_\psi(z, \theta \mid \boldsymbol{x}_{k-h:k}) \frac{p_t^\phi(\dot{x}_t, \theta, z \mid x_k)}{q_\psi(z, \theta \mid \boldsymbol{x}_{k-h:k})}\, d\theta\, dz \end{aligned} \tag{3}$$

We assume a structured factorization for the variational posterior, where the latent variable that corresponds to the physical parameters depends on the stochasticity latent variable: $q_\psi(z, \theta \mid \boldsymbol{x}_{k-h:k}) = q_\psi(\theta \mid \boldsymbol{x}_{k-h:k}, z) q_\psi(z \mid \boldsymbol{x}_{k-h:k})$. Applying Jensen's Inequality to equation 3, we derive a variational dynamics matching objective similarly to [Guo and Schwing, 2025]:

$$\begin{aligned} \log p_t^\phi(\dot{x}_t \mid x_k) \geq\ & \mathbb{E}_{q_\psi(z,\theta|\boldsymbol{x}_{k-h:k})}[\log p_t^\phi(\dot{x}_t \mid \theta, z, x_k)] \\ & - \mathbb{E}_{q_\psi(z|\boldsymbol{x}_{k-h:k})} \mathrm{KL}\left[q_\psi(\theta \mid \boldsymbol{x}_{k-h:k}, z) \| p(\theta)\right] \\ & - \mathrm{KL}\left[q_\psi(z \mid \boldsymbol{x}_{k-h:k}) \| p(z)\right], \end{aligned} \tag{4}$$

where $p(\theta)$ is the physics informed prior and $p(z) = \mathcal{N}(\mathbf{0}, \mathbf{I})$. Assuming a Gaussian likelihood for the velocity matching (corresponding to the squared error),

[2] We use the generic term of interpolant, which can be linear and non-linear. An example of the latter one is later introduced for solving II order ODE/PDEs

[Guo and Schwing, 2025], our objective becomes:

$$\mathcal{L}^{\text{VI}}_{\text{VGB-DM}}(\phi,\psi) = \mathbb{E}_{\pi(\boldsymbol{x}),\pi(\cdot|\boldsymbol{x})}\Big[- \mathbb{E}_{\substack{q_\psi(z,\theta|\boldsymbol{x}_{k-h:k})\\ t\sim\mathcal{U}(0,1)}}\Big\|(v_t^\phi(x_t \mid \theta, z)\circ f_p(x_t,\theta)) - \dot{x}_t\Big\|^2 - \text{KL}\left[q_\psi(\theta, z \mid \boldsymbol{x}_{k-h:k})\|p(\theta,z)\right]\Big] \quad (5)$$

In Appendix C, we provide the derivation of the ELBO and a proof that the learned distribution preserves the marginal data distribution. Practically, we model the approximate posterior by an encoder such that $q_\psi(z|\boldsymbol{x}_{k-h:k}) = \mathcal{N}\big(z;\mu_\psi(\boldsymbol{x}_{k-h:k}),\sigma_\psi(\boldsymbol{x}_{k-h:k})\big)$ and $q_\psi(\theta|\boldsymbol{x}_{k-h:k},z) = \mathcal{N}\big(\theta;\mu'_\psi(\boldsymbol{x}_{k-h:k},z),\sigma'_\psi(\boldsymbol{x}_{k-h:k},z)\big)$, enabling analytic computation of the KL-divergence in equation 5. We illustrate a schematic view of training and inference of our model in Figure 2.

Second-Order Dynamics. Physics systems can also include second-order dynamics, described by equations in the form of $\frac{\partial^2 x_t}{\partial t^2} = f_p(x_t,\theta)$, such as those governing pendulums, wave equations, and similar systems. To integrate these physics systems, we extend our objective equation 5 to this setting, by introducing a higher order, a target acceleration vector field. Specifically, we regress both velocity and acceleration vector field by a non-linear interpolation $\mathcal{I}(x_t; x_{k-1}, x_k, x_{k+1})$ using three consecutive trajectory points, retrieving its velocity and acceleration vector fields: $\frac{\partial\mathcal{I}}{dt} = \dot{x}_t$ and $\frac{\partial^2\mathcal{I}}{\partial t^2} = \ddot{x}_t$. The learned dynamics are then modelled as the paired system of velocity and acceleration fields

$$u_t^\phi(x_t,\dot{x}_t \mid \theta, z) := \begin{bmatrix} v_t^\phi(x_t \mid \theta, z) \\ a_t^\phi(x_t,\dot{x}_t \mid \theta, z)\circ f_p(x_t,\theta)\end{bmatrix}.$$

We modify the vector field matching term in equation 5 to obtain the second-order objective:

$$\mathbb{E}_{\substack{(z,\theta)\sim q_\psi(z,\theta|\boldsymbol{x}_{k-h:k})\\ t\sim\mathcal{U}(0,1)}}\Big[\|v_t^\phi(x_t \mid \theta, z) - \dot{x}_t\|^2 + \alpha\,\|a_t^\phi(x_t,\dot{x}_t \mid \theta, z)\circ f_p(x_t,\theta) - \ddot{x}_t\|^2\Big], \quad (6)$$

where α is a hyper-parameter that balances the optimization of the acceleration term to prevent excessive smoothing of the trajectory. In practice, we implement the non-linear interpolation using Lagrange interpolation.[3] Additionally, the velocity and acceleration vector fields u_t^ϕ share a backbone network, with the velocity and acceleration modelled as two distinct heads of the network. This architecture leverages shared parameters to ensure consistency between the learned velocity and acceleration fields. We implement this method when solving the Pendulum system as reported in our experiment, setting $\alpha = 0.5$.

[3]Other interpolation methods, such as Fourier interpolation or spline-based approaches, could be explored, but they are beyond the scope of this work.

4 Related works

We position our Variational Grey-Box Dynamics Matching (VGB-DM) framework within the landscape of learning dynamical systems, highlighting the distinctive features of our algorithm compared to existing data-driven and grey-box approaches.

Fully data-driven approaches such as Neural ODEs [Chen et al., 2018] and Flow Matching-based [Lipman et al., 2023, Albergo et al., 2023, Liu et al., 2023] including adaptation for forecasting time-series as in [Zhang et al., 2024] demonstrate modelling complex distributions using data, but operate as black boxes. However, these solutions can face limitations when training data are scarce, struggles when extrapolating beyond observed regimes, and offer limited interpretability. By contrast, VGB-DM incorporates incomplete physics models, enabling efficient learning even with limited data while maintaining interpretability through explicit physical parameters that can be inferred. Moreover, they enable further mechanistic inspection of the physics-informed and the learnable components, as studied by [Takeishi and Kalousis, 2023, Singh et al., 2025] analysing physics regularizers.

Recent grey-box modelling approaches build on Neural ODEs by combining physics-based models with data-driven neural networks, achieving strong results in parameter inference and dynamics forecasting [Yin et al., 2021, Takeishi and Kalousis, 2021, Mehta et al., 2021, Qian et al., 2021, Wehenkel et al., 2023, Verma et al., 2024]. However, these simulation-based methods require numerical ODE solvers during training along with physics model dynamics, presenting scaling difficulties for high-dimensional settings due to gradient computation through ODE solvers and parameter updates, particularly for long time horizons or complex dynamics leading to training difficulties and numerical stability [Chen et al., 2018, Takeishi and Kalousis, 2021, Wehenkel et al., 2023, Yin et al., 2021]. VGB-DM bypasses this paradigm by adopting a simulation-free method, avoiding these computational bottlenecks and stability issues, enabling more scalable and robust grey-box modelling for high-dimensional systems.

Notably, the work on Variational Rectified Flow Matching [Guo and Schwing, 2025] identifies a key limitation of deterministic flow matching methods: they struggle when modelling multi-modal velocity fields because the standard flow matching objective

Model	Pendulum ($\times 10^{-2}$)	RLC ($\times 10^{-1}$)	Reaction Diffusion ($\times 10^{-3}$)	Lorenz ($\times 10^{-1}$)
PhysVAE [Takeishi and Kalousis, 2021]	0.417 ± 0.001	0.29 ± 0.01	7.4 ± 1.3	13.12 ± 1.27
VBB-DM [Guo and Schwing, 2025]	1.058 ± 0.004	0.38 ± 0.03	8.0 ± 1.1	9.91 ± 1.22
BB-NODE [Chen et al., 2018]	0.452 ± 0.001	0.39 ± 0.04	7.5 ± 0.5	22.15 ± 7.84
TFM [Zhang et al., 2024]	0.348 ± 0.001	0.26 ± 0.01	8.5 ± 2.8	9.85 ± 1.32
Ours	$\mathbf{0.283} \pm 0.001$	$\mathbf{0.23} \pm 0.01$	$\mathbf{7.1} \pm 0.9$	$\mathbf{4.68} \pm 0.16$

Table 1: **Forecasting performance comparison (MSE).** Results on four dynamical systems benchmarks. Our method achieves the lowest error across all tasks.

leads to averaging over multiple target directions of the velocity vector field. To overcome this issue, they introduce a variational posterior $q_\psi(z|x_t, t)$ over latent variables, providing the necessary flexibility to model multimodal velocity distributions. However, VGB-DM introduces a structured variational posterior tailored for grey-box modelling and explicitly separates latent variables into two components: one capturing stochasticity and multi-modality in the dynamics, and another inferring interpretable physical parameters. This structured approach enhances both the expressiveness of the learned dynamics and the interpretability of the inferred parameters.

Finally, to integrate second-order dynamics systems, we extend our simulation-free method using non-linear interpolants (e.g. Lagrange Interpolation). This enables direct regression of both velocity and acceleration fields, addressing a broader class of physical systems than existing flow matching approaches.

5 Experiments

We conduct experiments on both synthetic datasets derived from differential equations (PDE/ODEs) (Section 5.1) and a real-world weather modelling task (Section 5.2). All experiments are repeated with multiple random seeds to ensure consistency and evaluate the stability of the compared methods[4].

5.1 Differential-Equation Datasets

Experimental setup. We evaluate our method on four canonical dynamical systems governed by differential equations: an RLC circuit, a reaction–diffusion (RD) system, a damped pendulum, and a chaotic Lorenz attractor system. In all experiments, grey-box models have access only to incomplete physics equations, while the dataset consists of state trajectories generated from the complete governing equations, following the experimental protocol of [Takeishi and Kalousis, 2021, Wehenkel et al., 2023]. Details of dataset construction and model architectures (encoder and vector field architectures) are provided in Appendix A.1. For each experiment, we follow the same network architecture design proposed in previously established benchmarks of grey-box models, for PhysVAE, this includes the additional regularization and hyperparameters suggested in [Takeishi and Kalousis, 2021, Wehenkel et al., 2023, Yin et al., 2021].

Baselines. For comparison, we benchmark against four baselines: PhysVAE [Takeishi and Kalousis, 2021], a state-of-the-art grey-box method, and two widely adopted black-box approaches where physics is not included in the model nor in the objective: Neural ODE for time series [Chen et al., 2018] (BB-NODE), the Variational Rectified Flow Matching [Guo and Schwing, 2025] adapted for time series (VBB-DM) and Trajectory Flow Matching (TFM) [Zhang et al., 2024]. All models are compared with equivalent network architecture complexity to ensure fair evaluation among them.

Forecasting performance. Table 1 presents mean squared error (MSE) results for time-series forecasting, where each model was trained on the largest training set size for each task. To assess extrapolation capabilities and generalization beyond the training regime, forecasts were tested on horizons three or four times longer than those encountered during training. Across all four benchmarks, our method consistently achieves lower errors than all baselines, demonstrating superior predictive accuracy and highlighting the benefit of the incomplete physics model as an inductive bias for learning the dynamics. In addition to the trajectory forecasts, Figure 9 of Appendix A.2 also reports the physics parameter estimation errors of the grey-box models, as part of our interpretability analysis.

Convergence behaviour. In Figure 3, we show the training convergence of each method on the test set across three training seeds. Our approach converges rapidly and attains lower final error than competing methods. By leveraging a simulation-free training ob-

[4] All our models were trained on a single RTX 3060 (12 GB), while ClimODE was trained on an RTX A5000 and required at least 25 GB.

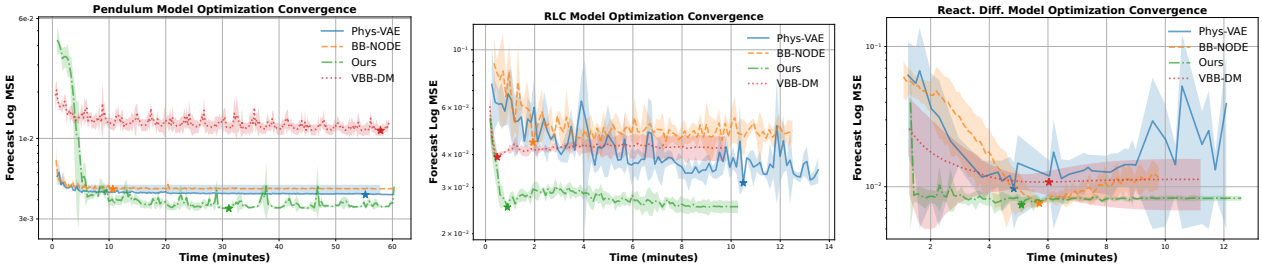

Figure 3: **Optimization convergence analysis.** Forecast logMSE vs. training time (minutes) for the Pendulum (left), RLC (center), and Reaction-Diffusion (right) tasks using the largest training dataset size. Our method demonstrates significantly faster convergence and achieves lower final error compared to simulation-based methods (PhysVAE, BB-NODE), while also exhibiting better stability than the black-box dynamics matching baseline (VBB-DM). This highlights the benefit of avoiding backpropagation through the ODE solver.

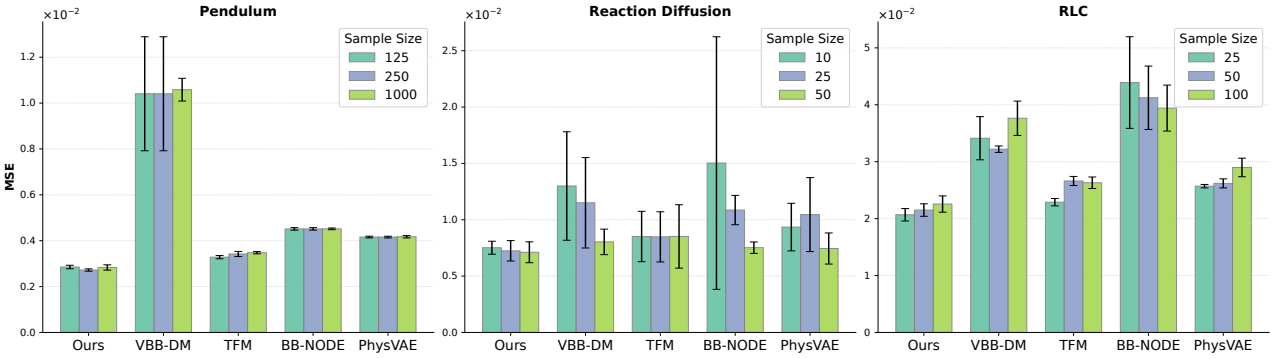

Figure 4: **Sample efficiency analysis.** Forecast MSE across varying training sample sizes for Pendulum, RLC, and Reaction-Diffusion tasks. Grey-box methods (Ours and PhysVAE) consistently outperform their black-box counterparts (VBB-DM and BB-NODE) across all regimes, with our method achieving the lowest error and variance, particularly under limited data.

jective, it avoids backpropagation through the ODE solver, which is a known source of gradient instability in simulation-based approaches such as PhysVAE and BB-NODE. This results in smoother and more consistent optimization trajectories. Furthermore, since each training step does not require running a numerical integrator, our method is significantly cheaper per iteration, translating into faster wall-clock convergence.

Sample efficiency. Grey-box approaches are often credited with providing robust performance under limited data [Wehenkel et al., 2023, Yin et al., 2021, Takeishi and Kalousis, 2021, Takeishi and Kalousis, 2023]. We evaluate each method across three training set sizes, ranging from 10 to 1000 samples, in order to assess robustness to data scarcity. In Figure 4, both grey-box models (ours and PhysVAE) consistently outperform their black-box counterparts: VGB-DM (ours) vs. VBB-DM and Phys-VAE vs. BB-NODE. Our method achieved lower error with smaller variance across all regimes, including when compared to TFM. Moreover, it further improves upon PhysVAE, exhibiting more stable predictions and reduced sensitivity to sample size. In contrast, VBB-DM and BB-NODE display substantially higher variance, especially in low-data regimes, suggesting that the physical prior in the grey-box models plays a critical regularization role when training data is scarce. These results confirm that embedding structural knowledge into the model is particularly advantageous when data collection is costly or limited.

5.2 Weather modelling

Physics-based formulations can provide valuable inductive biases for real-world datasets. In particular, large-scale weather dynamics can be modelled using the Advection equation, as demonstrated by [Verma et al., 2024]. To assess the effectiveness of our method, we conduct experiments on the ERA5 reanalysis dataset for medium-range weather forecasting. The dataset includes five key meteorological variables: *(i)* geopotential, *(ii)* ground temperature, *(iii)* atmospheric temperature, *(iv)* and *(v)* the two ground-level wind components (u10 and v10).

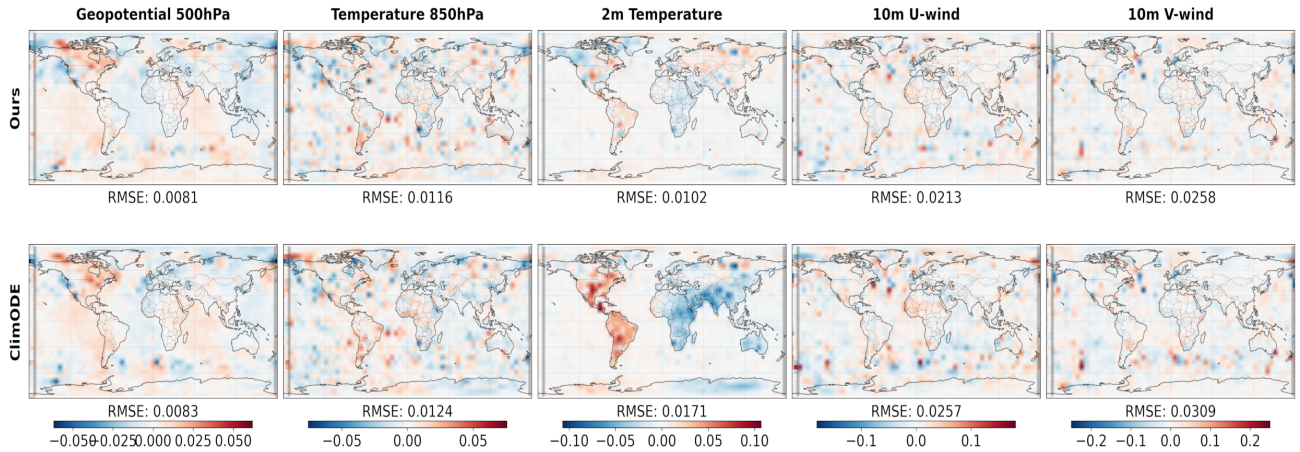

Figure 5: **Residual maps for weather forecasting** Visualization of absolute error maps between ground truth and predicted fields for five meteorological variables (Ours vs ClimODE)

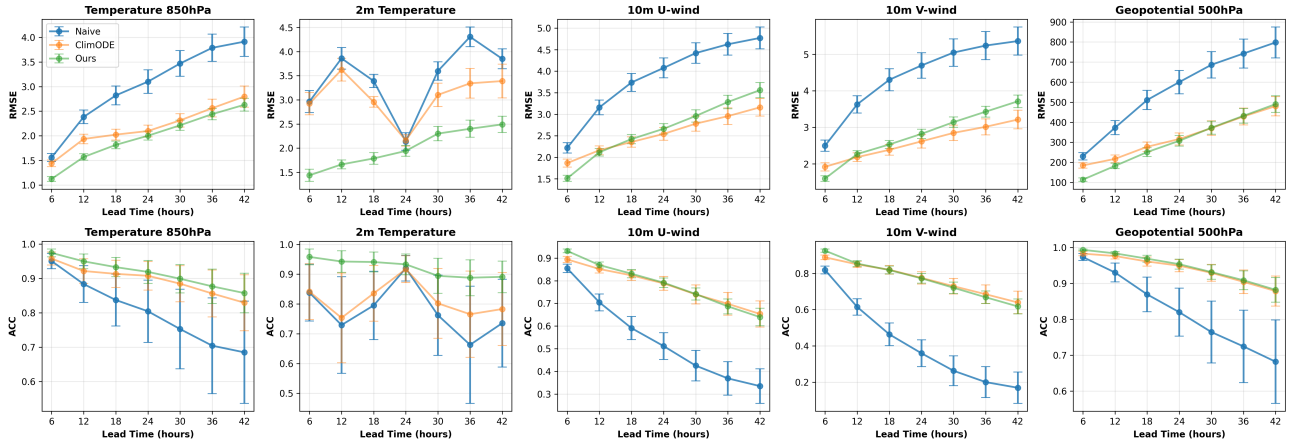

Figure 6: **Forecasting performance over** 42 **hours.** Latitude-weighted RMSE (top, lower is better) and ACC (bottom, higher is better) for ERA5 dataset. Our method performs better or on par than ClimODE.

Experimental Setup. We adopt the preprocessing pipeline and train/test splits introduced in ClimODE [Verma et al., 2024], and evaluate performance at two temporal resolutions: hourly and monthly. To ensure a fair comparison, we train our model and ClimODE under identical conditions, using the same maximum number of optimization steps (2000) and a fixed time budget of six hours. Evaluation is based on two standard metrics: latitude-weighted Root Mean Squared Error (RMSE) and the Anomaly Correlation Coefficient (ACC). Our goal is to provide a simulation-free alternative to physics-informed neural ODEs; hence, we restrict our comparison to ClimODE. We also include a naive persistence baseline (last observation) as a reference point to contextualize the results. Additional implementation and training details are provided in Appendix B.

Residual maps for hourly resolution. Figure 5 visualizes the absolute error maps and per-feature RMSE for a representative test sample. Across all five variables, our method yields lighter residual structures compared to ClimODE, particularly over oceanic regions. This indicates systematically reduced errors, which is further corroborated by consistently lower RMSE values in quantitative evaluations.

Results for hourly resolution. Figure 6 presents RMSE and ACC values over a 42-hour forecast horizon. Our approach performs on par with or better than ClimODE, achieving higher ACC and lower RMSE across the time steps.

Results for monthly resolution. Figure 7 extends the evaluation to a 5-month forecast horizon. The advantage becomes more pronounced as the forecast horizon extends, indicating that our method retains stronger temporal stability compared to Neural ODE–based formulations. This aligns with our ear-

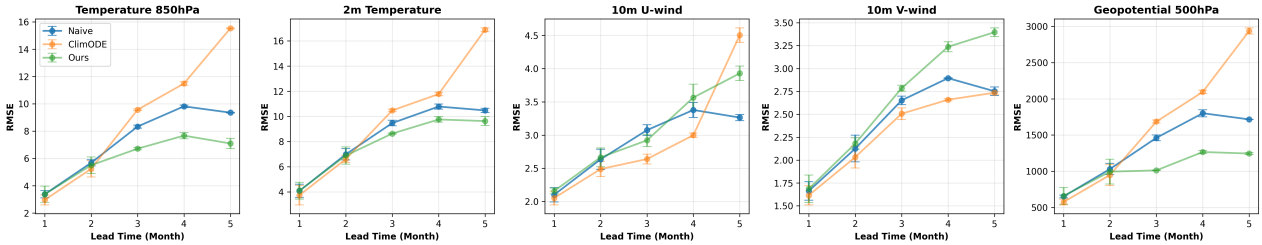

Figure 7: **Long-term forecasting performance over 5 months.** Latitude-weighted RMSE for ERA5 dataset over an extended period. Our method outperforms ClimODE by mitigating compounding errors over long horizons, with the performance gap widening as the forecast horizon extends.

lier experiments on synthetic ODE datasets, where our method also demonstrated greater robustness over extended prediction horizons, underscoring its ability to mitigate compounding errors — a challenge that traditional Neural ODE models struggle to overcome.

6 Conclusion

We introduced Variational Grey-Box Dynamics Matching (VGB-DM), a novel framework that combines the interpretability of physics-based models with the efficiency of simulation-free generative learning. By adopting a simulation-free training paradigm, VGB-DM bypasses computational and memory bottlenecks and stability issues inherent in traditional solver-based grey-box methods. Our structured variational approach effectively disentangles physical parameters from unknown stochasticity, enabling robust dynamics learning and interpretable parameter inference. Empirical evaluations on various ODE/PDE systems and a real-world weather forecasting task demonstrate that VGB-DM achieves state-of-the-art forecasting performance and faster convergence than existing grey-box and black-box baselines, crucially, while preserving the interpretability of the underlying physics model.

Limitations and Future Work. While VGB-DM offers significant advantages, it relies on several assumptions. First, the framework assumes that the underlying dynamics and the physics model are differentiable and relatively smooth. The performance on highly stiff or discontinuous physical systems remains an area for future investigation. However, the simulation-free nature of VGB-DM avoids gradient instability issues often encountered when backpropagating through solvers applied to stiff systems. Beyond these considerations, an important direction for future work is transfer across related physical systems. In similar domain (e.g., from a single to a double pendulum), the encoder can be reused as a pre-trained feature extractor, with only the output layers adapted if physical parameters are different. In this setting, a pre-trained VGB-DM model provides a strong warm start and can be fine-tuned to capture additional latent components arising from richer dynamics, enabling efficient knowledge transfer across families of systems.

Acknowledgements

We acknowledge the financial support of the Swiss National Science Foundation (SNF). G. Mercatali has been supported by SNF project IZLJZ2_214000, Interpretable Condition Monitoring, F. Lavda has been supported by SNF 200021_207428, Learning generative models for molecules, G. Singh SNF project CRSII5_209434, Migrate, A Multidisciplinary and IteGRated Approach for geoThermal Exploration. The computations were performed at the University of Geneva on "Baobab" and "Yggdrasil" HPC clusters.

References

[Albergo et al., 2023] Albergo, M. S., Boffi, N. M., and Vanden-Eijnden, E. (2023). Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*.

[Chen et al., 2018] Chen, R. T. Q., Rubanova, Y., Bettencourt, J., and Duvenaud, D. (2018). Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*.

[Guo and Schwing, 2025] Guo, P. and Schwing, A. G. (2025). Variational rectified flow matching. *arXiv preprint arXiv:2502.09616*.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Lipman et al., 2023] Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M., and Le, M. (2023). Flow matching for generative modeling. In *International Conference on Learning Representations*.

[Liu et al., 2023] Liu, X., Gong, C., and Liu, Q. (2023). Flow straight and fast: Learning to gen-

erate and transfer data with rectified flow. In *International Conference on Learning Representations*.

[Loshchilov and Hutter, 2017] Loshchilov, I. and Hutter, F. (2017). Decoupled weight decay regularization. In *International Conference on Learning Representations*.

[Mehta et al., 2021] Mehta, V., Char, I., Neiswanger, W., Chung, Y., Nelson, A. O., Boyer, M. D., Kolemen, E., and Schneider, J. G. (2021). Neural dynamical systems: Balancing structure and flexibility in physical prediction. In *IEEE Conference on Decision and Control*, pages 3735–3742.

[Qian et al., 2021] Qian, Z., Zame, W. R., Fleuren, L. M., Elbers, P., and van der Schaar, M. (2021). Integrating expert ODEs into neural ODEs: Pharmacology and disease progression. In *Advances in Neural Information Processing Systems*.

[Rasp et al., 2020] Rasp, S., Dueben, P. D., Scher, S., Weyn, J. A., Mouatadid, S., and Thuerey, N. (2020). WeatherBench: A benchmark data set for data-driven weather forecasting. *Journal of Advances in Modeling Earth Systems*, 12(11):e2020MS002203.

[Singh et al., 2025] Singh, G. S., Falkiewicz, M., and Kalousis, A. (2025). Hybrid generative modeling for incomplete physics: Deep grey-box meets optimal transport. In *ICLR Workshop on Explainable AI for Science*.

[Takeishi and Kalousis, 2021] Takeishi, N. and Kalousis, A. (2021). Physics-integrated variational autoencoders for robust and interpretable generative modeling. In *Advances in Neural Information Processing Systems*, volume 34, pages 14809–14821.

[Takeishi and Kalousis, 2023] Takeishi, N. and Kalousis, A. (2023). Deep grey-box modeling with adaptive data-driven models toward trustworthy estimation of theory-driven models. In *International Conference on Artificial Intelligence and Statistics*, volume 206, pages 4089–4100.

[Verma et al., 2024] Verma, Y., Heinonen, M., and Garg, V. (2024). ClimODE: Climate and weather forecasting with physics-informed neural ODEs. In *International Conference on Learning Representations*.

[Wehenkel et al., 2023] Wehenkel, A., Behrmann, J., Hsu, H., Sapiro, G., Louppe, G., and Jacobsen, J.-H. (2023). Robust hybrid learning with expert augmentation. *Transactions on Machine Learning Research*.

[Yin et al., 2021] Yin, Y., Le Guen, V., Dona, J., de Bézenac, E., Ayed, I., Thome, N., and Gallinari, P. (2021). Augmenting physical models with deep networks for complex dynamics forecasting. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124012.

[Zhang et al., 2024] Zhang, X., Pu, Y., Kawamura, Y., Loza, A., Bengio, Y., Shung, D. L., and Tong, A. (2024). Trajectory flow matching with applications to clinical time series modelling. In *Advances in Neural Information Processing Systems*, volume 37, pages 107198–107224.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes] See Section 2.1 and 3.
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes] See Section 3
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes] Will release upon acceptance.

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes] See Sections 2 and 3
   (b) Complete proofs of all theoretical results. [Yes] Appendix C.
   (c) Clear explanations of any assumptions. [Yes] See Section 2.1

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes] We report in Appendix A for ode-datasets and in Appendix B.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes] Appendix 5.1 for ode-datasets and in Appendix B.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes] In Section 5

(d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes] In Section 5.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:

   (a) Citations of the creator If your work uses existing assets. [Yes]

   (b) The license information of the assets, if applicable. [Yes]

   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]

   (d) Information about consent from data providers/curators. [Not Applicable] Not used.

   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable] Not used.

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

   (a) The full text of instructions given to participants and screenshots. [Not Applicable] Not used.

   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable] Not used.

   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable] Not used.

Supplementary Materials

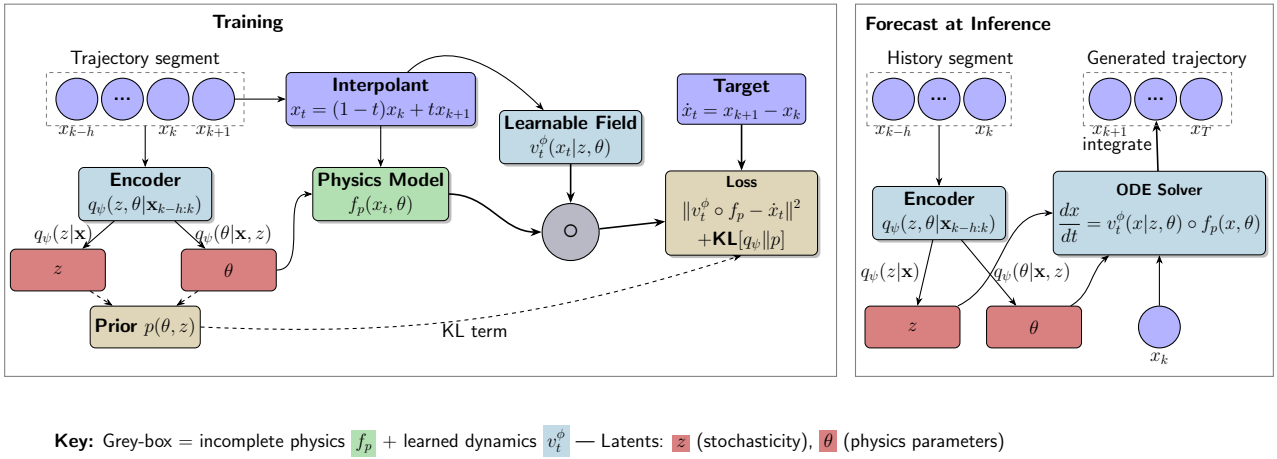

Figure 8: Overview of the Variational Grey-Box Dynamics Matching (VGB-DM) framework.

Method Overview. Figure 8 illustrates the proposed **Variational Grey-Box Dynamics Matching (VGB-DM)** framework. During training, the model is *simulation-free*: it does not require integrating the dynamics to compute gradients. Instead, the physics-based component f_p is directly combined with the learnable vector field v_t^ϕ. The training objective is optimized without numerical simulations, avoiding memory and computation bottlenecks of backpropagation through the solver and optimization instability. The model includes an encoder that maps a segment of the observed trajectory into two structured latent variables: (i) the stochastic latent variable z, and (ii) the physics-related variable θ. These latents condition both the physics model and the learnable field, allowing the system to infer underlying physical parameters and represent multimodal velocity vector fields.

At inference (forecasting) time, the model predicts future trajectories by integrating the combined dynamics of the physics model and the learnt component. Multiple forecast realisations can be generated by sampling different latents from the learnt posterior.

A Differential-Equation Benchmarks

The synthetic dataset is based on Differential Equation, where initial conditions and parameters are stochastic, following the work of [Takeishi and Kalousis, 2021, Wehenkel et al., 2023]. In the next subsections we report each experiment describing data generation and model architecture and hyperparameters used. For each experiment, we followed the same networks architecture design proposed by [Takeishi and Kalousis, 2021, Wehenkel et al., 2023]. We compare our method with its black-box versions using the simulation-based methods (BB-NODE) and the simulation-free methods (VBB-DM), where in the latter physics is not included in the model. In addition to our grey-box model we also compared its performance with the Phys-VAE [Takeishi and Kalousis, 2021] used also by [Wehenkel et al., 2023]. We follow for each experiment all the hyperparameters used by the models and regularisers as listed in their work.

A.1 Experiments and implementations

RLC. The RLC series circuit, consisting of a resistor (R), capacitor (C), and inductor (L), models a broad class of transfer functions. The system state at time t is defined as $x_t = [I_t, U_t]^\top$, where U_t denotes the capacitor

voltage and I_t the circuit current. The state evolution follows:

$$\frac{\partial}{\partial t}\left[I_t \; U_t\right] = \underbrace{\left[\tfrac{I_t}{C} \;\; \tfrac{1}{L}(V_t - U_t)\right]}_{f_p} + \underbrace{\left[0 \;\; -\tfrac{R}{L} I_t\right]}_{\text{missing physics}} . \tag{7}$$

The dataset is generated by sampling L and C uniformly from $[1, 3] \times [0.5, 1.5]$, and $R \sim \mathcal{U}(1, 3)$. The input voltage V_t is a fixed, known signal. Initial conditions are given by $U_0 \sim \mathcal{N}(0, 1)$ and $I_0 = 0$. The incomplete physics model f_p in the grey-box setup includes only the first term of the full equation.

Each trajectory is simulated from $t_0 = 0$ to $t_{\max} = 20$ seconds with a time resolution of $\Delta t = 0.1$ s, yielding 200 time points per trajectory. The training, validation, and test sets contain 1000, 100, and 100 samples, respectively. During training, trajectories are divided into segments (history windows) of size $h = 25$. We also evaluate model performance under different training sample sizes, as reported in Figure 4.

For this experiment, we adopt the same network architecture as in [Wehenkel et al., 2023]. Optimization is performed using AdamW [Loshchilov and Hutter, 2017, Kingma and Ba, 2014] with a learning rate of 0.001, a cosine annealing scheduler, and a weight decay of 5×10^{-7}.

Reaction diffusion In this experiment, we address a high-dimensional problem, where $\boldsymbol{x} \in \mathbb{R}^{2\times32\times32\times\tau}$, governed by a two-dimensional reaction–diffusion PDE of the FitzHugh–Nagumo type:

$$\frac{du}{dt} = \underbrace{\Delta u + u - u^3 - v}_{f_p} - k, \quad \frac{dv}{dt} = \underbrace{b\Delta v + u - v}_{f_p}. \tag{8}$$

Here, Δ denotes the Laplace operator, while b, and k are system parameters, and $u(0), v(0)$ define the initial conditions. Trajectory samples are generated by drawing both the initial conditions and parameters from uniform distributions, following the setup of [Wehenkel et al., 2023]. The incomplete physics model f_p omits the parameter k. Each simulation produces a state space representation $[u(t), v(t)] \in \mathbb{R}^{2\times32\times32}$.

We trained the models on three datasets containing $50, 25$ and 10 trajectories, respectively. Each trajectory spans a time interval $[t_0, t_{\max}] = [0.0, 1.0]$ and is divided into segments (history windows) of size $h = 5$. The test set consists of 250 trajectories with a longer time horizon of length 50 ($t_0 = 0.0, t_{\max} = 5.0, \Delta t = 0.1$), used to evaluate extrapolation and generalization performance.

Our model adopts the same network architecture as in [Wehenkel et al., 2023]. Optimization is performed using AdamW with a learning rate of 0.001, a cosine annealing scheduler, and a weight decay of 5×10^{-7}.

Damped Pendulum We generated the dataset by solving a non-linear damped pendulum:

$$\underbrace{\frac{\mathrm{d}^2 x(t)}{\mathrm{d}t^2} + \omega^2 \sin x(t)}_{f_p} + \underbrace{\xi \frac{dx(t)}{dt}}_{\text{missing physics}} = 0, \tag{9}$$

where the initial pendulum angle is $x(0) \sim \mathcal{U}(-1.57, 1.57)$, the angular frequency $\omega \sim \mathcal{U}(0.785, 3.14)$, and the damping coefficient $\xi \sim \mathcal{U}(0.6, 1.5)$. The incomplete physics model f_p corresponds to a frictionless pendulum. We train the model on 1000 training, 250 validation, and 100 test instances, each with trajectory length 200 ($t_0 = 0.0, t_{\max} = 20, \Delta t = 0.1$), split into segment (history windows) of size $h = 25$.

In this experiment, VGB-DM regresses 2D vector fields — velocity and acceleration — constructed via Lagrange interpolation using three consecutive points from each sampled trajectory segment, as described in equation 6. The encoder architecture is identical to that of Phys-VAE. The learnt 2D vector field, $[v_t^\phi, a_t^\phi]$, is modelled using a shared MLP with two hidden layers of size [64, 64], followed by two distinct head networks (each with layers [64, 64]) corresponding to the velocity and acceleration components, respectively. We set the acceleration hyperparameter in equation 6 to $\alpha = 0.5$, which provides a balance between smoothness and curvature in the learnt dynamics. The model is optimized using AdamW with a learning rate of 0.001, a cosine annealing learning rate scheduler, and a weight decay of 5×10^{-7}.

Chaotic Lorenz Attractor System We analysed the Lorenz System for varying initial conditions and parameters. The system dynamics are governed by:

$$\frac{\mathrm{d}u}{\mathrm{d}t} = \sigma(v - u), \tag{10}$$

$$\frac{\mathrm{d}v}{\mathrm{d}t} = \underbrace{u(\rho - w)}_{\text{missing physics}} - v, \tag{11}$$

$$\frac{\mathrm{d}w}{\mathrm{d}t} = uv - \beta w. \tag{12}$$

We generate the trajectory with $u(0), v(0), w(0)$ sampled from a Gaussian distribution and the parameters are also randomly sampled with $\sigma \sim \mathcal{U}(9.5, 10.5), \rho \sim \mathcal{U}(27.0, 29.0), \beta \sim \mathcal{U}(2.6, 2.8)$. Each trajectory spans a time interval $t_0 = 0.0$ to $t_{\max} = 2.0$ and time resolution $\Delta t = 0.0339$. The training dataset is made of 1000 trajectories divided into segments of size $h = 30$. Instead, the validation and test set consists of 250 trajectories of length 60.

For our grey-box model, the known physics $f_p(\cdot; \theta)$ incorporates the parameters $\theta = [\sigma, \beta]$. The term $u(\rho - w)$, representing buoyancy and non-linear advection, is treated as the unmodeled, stochastic part of the system. We remark that all latent parameters are inferred in a fully unsupervised manner in our method.

We designed a multi-layer gated recurrent unit (GRU) RNN-based encoder with 64 hidden states and two linear layers to infer both the physics parameters θ and the missing stochasticity $z = \rho$ directly from the observed trajectory. The velocity vector field v^ϕ is modelled using a simple conditional MLP with 4 hidden layers of 128 neurons. The model is optimized using AdamW with a learning rate of 0.001, a cosine annealing learning rate scheduler, and a weight decay of 5×10^{-7}.

A.2 Additional Results and Analyses

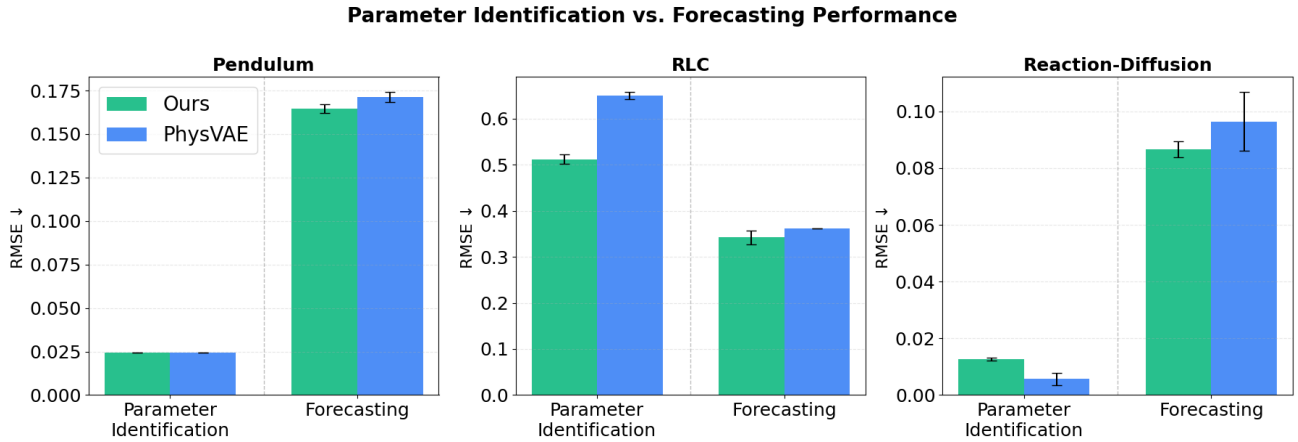

Figure 9: Parameter identification (inference) vs. forecasting performance across the three dynamical systems (lower RMSE is better).

Figure 9 compares parameter identification (inference) and forecasting performance across the three dynamical systems. Our method achieves consistently better forecasting accuracy and comparable or slightly improved parameter inference. In particular, forecasting errors are lower across all systems, while parameter estimation remains on par with PhysVAE. Unlike PhysVAE, which requires careful regularization tuning for stable parameter recovery, our approach achieves balanced performance without such tuning, maintaining strong forecasting capability and robust generalization.

Analysis of Loss Decomposition The complete variational objective, $\mathcal{L}^{\text{VI}}_{\text{VGB-DM}}(\phi, \psi)$ is composed of three distinct terms: the Flow Matching (FM) loss which focuses on forecasting the dynamics , and the Physics Parameter KL-divergence (Ph-KL), and the Latent Variable KL-divergence (Z-KL), as defined in Equation 5.

To assess the relative contribution of each term to the total loss and to identify potential optimization bottlenecks, we report their mean values and standard deviations evaluated on the test set in Table 2. This decomposition allows for a clear comparison of which component dominates the overall objective for each experimental system.

$$\mathcal{L}^{\text{VI}}_{\text{VGB-DM}}(\phi,\psi) = \mathbb{E}_{\pi(\boldsymbol{x}),\pi(\cdot|\boldsymbol{x})}\Big[\mathbb{E}_{\substack{q_\psi(z,\theta|\boldsymbol{x}_{k-h:k})\\ t\sim\mathcal{U}(0,1)}} \underbrace{\Big\|(v_t^\phi(x_t \mid \theta, z)\circ f_p(x_t,\theta)) - \dot{x}_t\Big\|^2}_{\text{FM Loss}}$$

$$- \underbrace{\text{KL}\Big[q_\psi(\theta \mid z, \boldsymbol{x}_{k-h:k}) \parallel p(\theta)\Big]}_{\text{Ph-KL}} \tag{13}$$

$$- \underbrace{\text{KL}\Big[q_\psi(z \mid \boldsymbol{x}_{k-h:k}) \parallel p(z)\Big]}_{\text{Z-KL}}\Big] \tag{14}$$

Table 2: Evaluation of the N-ELBO terms

Experiment	**FM**	**Ph-KL**	**Z-KL**
Pendulum	$3.9\times10^{-5}\pm1.8\times10^{-5}$	$8.8\times10^{-4}\pm1.1\times10^{-4}$	2.5 ± 0.22
RLC	0.52 ± 0.03	3.4 ± 1.4	0.70 ± 0.001
RD	$3\times10^{-3}\pm1\times10^{-3}$	2.7 ± 1.1	$2\times10^{-3}\pm2\times10^{-4}$
Lorenz	3.8 ± 1.5	2.3 ± 0.8	0.003 ± 0.001

A.2.1 Parameter Consistency Across Trajectory

To validate the consistency of parameter estimates—given that our model infers them from input segment assumed to be representative—we evaluated their variation across each full trajectory.

We computed the Coefficient of Variation (CV) for the estimated parameter values across the sliding windows of each trajectory. The CV, which normalizes the standard deviation by the mean, confirms the stability regardless of the parameter's scale. The median CV across all trajectories is 0.05 (or 5%) or less, thus Median $\left(\frac{\sigma}{|\mu|}\right) \leq 0.05$. This demonstrates that the estimated physical parameters do not drift or fluctuate significantly during the episode, confirming the robustness of the inference method. We report their values for each experiment in Table 3.

Exp	**Phys. Params. CV**
RLC	0.0461, 0.0391
Pendulum	0.0496
Reaction–Diffusion	0.0072, 0.0031
Lorenz	0.0012, 0.0026

Table 3: Median consistency of physics parameters across trajectory segments (test set).

B Experimental details for Weather experiments

B.1 Dataset details

We utilize the preprocessed ERA5 dataset provided by WeatherBench [Rasp et al., 2020], a widely used benchmark framework for evaluating data-driven weather forecasting models. The original ERA5 data at 0.25° resolution is regridded by WeatherBench to coarser resolutions of 5.625°, 2.8125°, and 1.40625°; in this work, we adopt the 5.625° dataset at 6-hour intervals. Our study focuses on $K = 5$ key variables: 2-metre temperature (t2m), atmospheric temperature (t), geopotential (z), and the 10-metre wind vector components (u10, v10). All variables are normalized to the range $[0, 1]$ using min–max scaling. Among these, z and t are standard verification

variables in medium-range Numerical Weather Prediction (NWP) models, while t2m and (u10, v10) are directly relevant to surface-level conditions that impact human activities.

Following [Verma et al., 2024], the dataset spans ten years of training data (2006–2015), one year for validation (2016), and two years for testing (2017–2018). The full set of ERA5 variables employed in our experiments is summarized in Table 4.

Table 4: ECMWF data variables from ERA5 used in our dataset. *Static* variables are time-independent, *Single* represents surface-level variables, and *Atmospheric* represents time-varying atmospheric properties at chosen altitudes.

Type	Variable name	Abbrev.	ECMWF ID	Levels
Static	Land-sea mask	lsm	172	
Single	2 metre temperature	t2m	167	
Single	10 metre U wind component	u10	165	
Single	10 metre V wind component	v10	166	
Atmospheric	Geopotential	z	129	500
Atmospheric	Temperature	t	130	850

B.2 Details of Climate VGB-DM

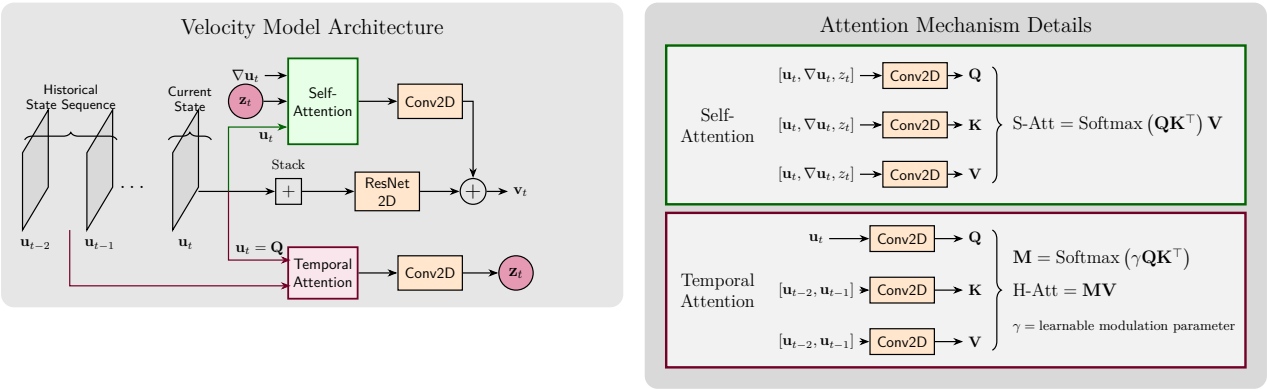

Figure 10: Diagram of the Velocity model architecture (left) and its lightweight attentions mechanism (right)

For the climate modelling experiments in Section 5.2 of the paper, our neural architecture is based on the design presented in [Verma et al., 2024], adapted to our framework and excluding ClimODE's external emission network g. Beyond the original ClimODE model, we additionally incorporate a lightweight temporal attention encoder applied to the trajectory states, with a window size $h = 2$, and model the source terms. A full description of our climate modelling framework is given in equation 15, and the architecture of the velocity model is illustrated in the schematic of Figure 10. We optimize the model using AdamW with a learning rate of 5×10^{-4}.

The lightweight temporal attention encoder uses multi-head temporal attention in which the current state attends to a short sequence of past states independently at each spatial location. Queries are computed from the current state, while keys and values are obtained from the history via learned convolutional projections, preserving spatial resolution. Attention is applied exclusively over the temporal dimension, and the resulting history-aware features are projected back to the original channel space and aligned with the current state. During training, the history consists of the previous two time steps, enabling the encoder to exploit true temporal context. At inference time, when only the current state is available, this state is replicated to match the required history length and used as the attention context. This encoder processes the sampled state field $\boldsymbol{u}_{t-h:t}$ and produces a latent z_t, which is concatenated with the input to the grey-box model as described in the next paragraph[5]. Although our main formulation allows for a variational posterior over the encoder latent variables, in practice we

[5]The implementation of the Climate VGB-DM model is provided in our VGB-DM GitHub repository

employ a deterministic temporal encoder, i.e. no KL regularization term is optimized. This choice is motivated by the fact that the underlying physics model does not involve latent physical parameters, but rather accounts for unmodelled dynamics such as velocity and source terms. Since all model evaluations are performed on forecasting performance, a deterministic encoder provides a simpler and effective alternative to a variational formulation without affecting the predictive objective.

Weather Grey-Box Dynamics. For consistency with the notation introduced in [Verma et al., 2024], we remind the reader that spatio-temporal states are indicated by u_t, while x is used for spatial coordinates. We outline the grey-box model of the advection type using a partial differential equation to represent the temporal evolution of the spatial field $u(x,t)$:

$$\frac{\partial u}{\partial t} = \underbrace{-v_t^{\phi}(u_t, \nabla u_t, z_t, \psi) \cdot \nabla u_t - u_t \left(\nabla \cdot v_t^{\phi}(u_t, \nabla u_t, z_t, \psi)\right) + s_t^{\phi}(u_t, \nabla u_t, \psi)}_{V_t^{\phi}(u_t|z_t) = v_t^{\phi}(u_t|z_t) \circ f_p(u_t, v_t^{\phi}) \,\circ\, s_t^{\phi}(u_t, \nabla u_t, z_t, \psi)}, \tag{15}$$

where v_t^{ϕ} is the learnable flow's velocity of the grey-box model and s_t^{ϕ} models the sources, z_t describes the latent variable from the encoder, finally the spatio-temporal embeddings is denoted by $\psi \in \mathbb{R}^{C \times H \times W}$, as in [Verma et al., 2024] to keep the same notation. We also recall that $\nabla = \nabla_x$ denotes spatial gradients, and $\nabla \cdot v_t^{\phi} = \mathrm{tr}(\nabla v_t^{\phi})$ the divergence as denoted in ClimODE [Verma et al., 2024].

B.3 Time resolutions

We have two resolutions, hourly over 42 hours, and monthly, over 5 months. The hourly resolution results are in Figure 6 and Figure 5. The monthly resolution results are in Figure 7.

B.4 Evaluation metrics

Following [Verma et al., 2024], we assess benchmarks using latitude-weighted RMSE and Anomaly Correlation Coefficient (ACC) following the de-normalization of predictions.

$$\mathrm{RMSE} = \frac{1}{N} \sum_t \sqrt{\frac{1}{HW} \sum_h^H \sum_w^W \alpha(h)(y_{thw} - u_{thw})^2}, \quad \mathrm{ACC} = \frac{\sum_{t,h,w} \alpha(h) \tilde{y}_{thw} \tilde{u}_{thw}}{\sqrt{\sum_{t,h,w} \alpha(h) \tilde{y}_{thw}^2} \sqrt{\sum_{t,h,w} \alpha(h) u_{thw}^2}}$$

where $\alpha(h) = \cos(h) / \frac{1}{H} \sum_h^H \cos(h')$ is the latitude weight and $\tilde{y} = y - C$ and $\tilde{u} = u - C$ are averaged against empirical mean $C = \frac{1}{N} \sum_t y_{thw}$.

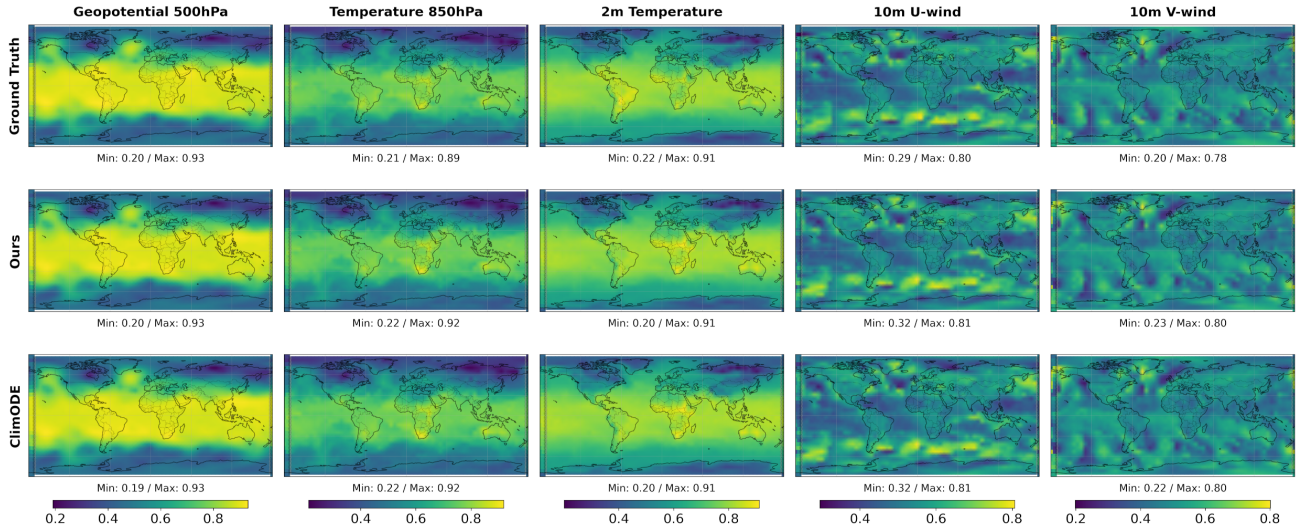

Figure 11: Example of Climate forecasting using VGB-DM over 42 hours.

B.5 Predictions visualization

In Figure 11 we show the forecast of our model on the ERA5 dataset over 42 hours horizon. In the main text we reported the error between true and predicted values to facilitate showing the difference between our model and ClimODE.

C Proofs and Derivations

In this section we prove that the marginal of the our velocity vector field are preserved, we demonstrate it by using the existing **Proposition 3.2 (Coupling Preservation)** of [Zhang et al., 2024]. In addition in the sub-section C.3, we report the full ELBO derivation of our objective 5.

C.1 Proof: Coupling Preservation with learnt Latents by Marginal Preservation

Setup Denote $\boldsymbol{x} := (x_{t_1}, x_{t_2}, \ldots, x_{t_T})$ as a trajectory sampled from the data distribution $\pi(\boldsymbol{x})$ and:

$$p_t(x_t|x) := \mathcal{N}((t_{k+1} - t)x_{t_k} + (t - t_k)x_{t_{k+1}}, \sigma^2(t_{k+1} - t)(t - t_k)\mathbf{I}), \tag{16}$$

$$u_t(x_t|x) := \frac{x_{t_{k+1}} - x_t}{t_{k+1} - t}. \tag{17}$$

where $p_t(x_t|x)$ is the Gaussian interpolant between consecutive trajectory points x_{t_k} and $x_{t_{k+1}}$ and $u_t(x_t|\boldsymbol{x})$ is the target velocity vector field.

Consider our simulation-free method, where the conditioning variable is given by latent representations produced by an encoder network. More formally, let $\mathbf{c} = (\theta, z) \in \mathbb{R}^{d_c}$ denote the concatenated latent variables obtained from an encoder $E_\psi : \mathcal{X}^T \to \mathbb{R}^{d_c}$ that maps trajectories to latent codes. The grey-box velocity field is defined as

$$V_t^\phi(x_t|\mathbf{c}) = v_t^\phi(x_t|\mathbf{c}) \circ f_p(x_t, \theta),$$

where V_t^ϕ is the total velocity vector field incorporating the physics model, $u_t(x_t|\boldsymbol{x})$ is the true velocity field conditioned on the full trajectory $\boldsymbol{x}$, $q_\psi(\mathbf{c}|\boldsymbol{x})$ is the encoder posterior parameterised by ψ, and $p(\mathbf{c})$ is the prior distribution over the latents.

Our training objective is:

$$\mathcal{L}_{\text{VGB-DM}}(\phi, \psi) = \mathbb{E}_{t,\pi(\boldsymbol{x})}\left[\mathbb{E}_{q_\psi(\mathbf{c}|\boldsymbol{x})}\left[\|V_t^\phi(x_t|\mathbf{c}) - u_t(x_t|\boldsymbol{x})\|^2\right] + \text{KL}(q_\psi(\mathbf{c}|\boldsymbol{x})\|p(\mathbf{c}))\right], \tag{18}$$

where V_t^ϕ is the total velocity vector field incorporating the physics model, $u_t(x_t|\boldsymbol{x})$ is the true velocity field conditioned on the full trajectory x, $q_\psi(\mathbf{c}|\boldsymbol{x})$ is the encoder posterior parameterised by ψ, and $p(\mathbf{c})$ is the prior distribution over the latents.

Main Result We prove that training with the learnt latent conditioning and the grey-box velocity field preserves the optimal coupling between trajectory distributions, ensuring the marginal vector field $u_t(x_t|\mathbf{c})$ matches the true trajectory-conditioned $u_t(x_t|\boldsymbol{x})$ vector field in expectation.

Our proof schema follows as:

1. We introduce **Assumption (A4)** characterizing the optimal encoder.
2. We prove that **(A4)** ensures the marginal vector field and satisfies the conditions of **Proposition 3.2** (Coupling Preservation) [Zhang et al., 2024].
3. We recall **Proposition 3.2** to conclude that $\Pi(u)^\star = \Pi^\star(\boldsymbol{x}_{1:T})$.

Throughout, we rely on the regularity conditions established in **Lemma A.1** of [Zhang et al., 2024] regarding Lipschitz continuity and integrability.

Assumption (A4): Optimal Posterior Encoder The encoder learns the optimal posterior distribution, that is, there exists ψ^* such that $q_{\psi^*}(\mathbf{c}|\boldsymbol{x}) = q^*(\mathbf{c}|\boldsymbol{x})$ where

$$q^*(\mathbf{c}|\boldsymbol{x}) = \arg\min_{q(\mathbf{c}|\boldsymbol{x})} \mathbb{E}_{t,x\sim\pi(\boldsymbol{x}),\mathbf{c}\sim q(\mathbf{c}|\boldsymbol{x}),x_t\sim p_t(x_t|\boldsymbol{x})} \left[\|V_t^\phi(x_t|\mathbf{c}) - u_t(x_t|\boldsymbol{x})\|_2^2 + \mathrm{KL}(q(\mathbf{c}|\boldsymbol{x})\|p(\mathbf{c}))\right] = 0 \tag{19}$$

This assumption says that the encoder captures sufficient trajectory-specific information in the latent code $\mathbf{c} = (\theta, z)$ such that conditioning on $\mathbf{c}$ allows perfect recovery of the trajectory-dependent vector field $u_t(x_t|\boldsymbol{x})$ via the grey-box model V_t^ϕ (i.e., the minimal MSE is zero), while the KL divergence encourages the posterior to match the prior.

Main Proposition

Proposition 1 (**Coupling Preservation with learnt Latents by Marginal Preservation**). *Under the regularity conditions of Lemma A.1 [Zhang et al., 2024], if* ***Assumption (A4)*** *holds and* $x, \pi(\boldsymbol{x}), p_t(x_t|\boldsymbol{x})$, *and* $u_t(x_t|\boldsymbol{x})$, *then training with* $\mathcal{L}_{VGB\text{-}DM}(\phi, \psi^*)$ *preserves the optimal coupling:*

$$\Pi(u)^\star = \Pi^\star(\boldsymbol{x}_{1:T}).$$

C.2 Proof of Proposition 1

Step 1: Recall Proposition 3.2. [Zhang et al., 2024] establishes that under mild regularity conditions, if

$$\mathbb{E}_{t,x\sim\pi(\boldsymbol{x}),\mathbf{c}\sim q(\mathbf{c}|\boldsymbol{x}),x_t\sim p_t(x_t|\boldsymbol{x})}\|u_t(x_t|\boldsymbol{x},\mathbf{c}) - u_t(x_t|\mathbf{c})\|_2^2 = 0, \tag{20}$$

then the push-forward flows satisfy $\int_0^T u_t(x_t|\boldsymbol{x},\mathbf{c})dt = \phi(x_0,\mathbf{c},\boldsymbol{x}) = \phi(x_0,\mathbf{c}) = \int_0^T u_t(x_t|\mathbf{c})dt$ and consequently $\Pi(u)^\star = \Pi^\star(\boldsymbol{x}_{1:T})$.

We aim to confirm this condition by applying **Assumption (A4)** (see **Steps 2-3**), observing that the grey-box speed V_t^ϕ serves as an approximation for u_t under optimal conditions.

Step 2: Marginal Vector Field. The marginal vector field conditioned on $\mathbf{c}$ is critical because it determines the push-forward flow $\phi(x_0,\mathbf{c}) = \int_0^T u_t(x_t|\mathbf{c})dt$ that defines the coupling $\Pi(u)$. Therefore, by marginalising it over trajectories, we have:

$$u_t(x_t|\mathbf{c}) = \mathbb{E}_{\boldsymbol{x}\sim\pi(\boldsymbol{x}|\mathbf{c})}[u_t(x_t|\boldsymbol{x})].$$

By Bayes' rule, the posterior over trajectories given the latent code is:

$$\pi(\boldsymbol{x}|\mathbf{c}) = \frac{q(\mathbf{c}|\boldsymbol{x})\pi(\boldsymbol{x})}{\int q(\mathbf{c}|\boldsymbol{x}')\pi(\boldsymbol{x}')d\boldsymbol{x}'}.$$

When the encoder satisfies **Assumption (A4)**, we have $q(\mathbf{c}|\boldsymbol{x}) = q^*(\mathbf{c}|\boldsymbol{x})$, and therefore:

$$u_t(x_t|\mathbf{c}) = \mathbb{E}_{\boldsymbol{x}\sim\pi^*(\boldsymbol{x}|\mathbf{c})}[u_t(x_t|\boldsymbol{x})]. \tag{21}$$

At optimality, $V_t^\phi(x_t|\mathbf{c})$ matches this marginal.

Step 3: Verification of Proposition 3.2 Condition. We now show that Assumption (A4) implies the condition required by Proposition 3.2. Consider:

$$\begin{aligned}&\mathbb{E}_{t,x\sim\pi(\boldsymbol{x}),\mathbf{c}\sim q^*(\mathbf{c}|\boldsymbol{x}),x_t\sim p_t(x_t|\boldsymbol{x})}\|u_t(x_t|\boldsymbol{x},\mathbf{c}) - u_t(x_t|\mathbf{c})\|_2^2\\ &= \mathbb{E}_{t,x\sim\pi(\boldsymbol{x}),\mathbf{c}\sim q^*(\mathbf{c}|\boldsymbol{x}),x_t\sim p_t(x_t|\boldsymbol{x})}\|u_t(x_t|\boldsymbol{x}) - u_t(x_t|\mathbf{c})\|_2^2,\end{aligned}$$

where the equality follows since $u_t(x_t|x,\mathbf{c}) = u_t(x_t|\boldsymbol{x})$ (the true vector field depends only on the trajectory x).

By the Tower property of conditional expectation:

$$\begin{aligned}&\mathbb{E}_{t,x\sim\pi(\boldsymbol{x}),\mathbf{c}\sim q^*(\mathbf{c}|\boldsymbol{x}),x_t\sim p_t(x_t|\boldsymbol{x})}\|u_t(x_t|\boldsymbol{x}) - u_t(x_t|\mathbf{c})\|_2^2\\ &= \mathbb{E}_{t,\mathbf{c}}\left[\mathbb{E}_{x\sim q^*(x|\mathbf{c}),x_t\sim p_t(x_t|\boldsymbol{x})}\left[\|u_t(x_t|\boldsymbol{x}) - u_t(x_t|\mathbf{c})\|_2^2 \,\middle|\, \mathbf{c}\right]\right].\end{aligned}$$

For the optimal encoder $q^*(\mathbf{c}|\boldsymbol{x})$, by its defining property in Assumption (A4), we have:

$$u_t(x_t|\mathbf{c}) = \mathbb{E}_{x\sim q^*(x|\mathbf{c})}[u_t(x_t|\boldsymbol{x})], \tag{22}$$

which is the best predictor of $u_t(x_t|\boldsymbol{x})$ given $\mathbf{c}$ in terms of L^2 norm. Moreover, since Assumption (A4) specifies that the minimal value of the MSE objective is zero (with the KL term providing regularization but not affecting the zero-MSE at optimality), the conditional variance is zero:

$$\mathbb{E}_{x\sim q^*(x|\mathbf{c}),x_t\sim p_t(x_t|\boldsymbol{x})}\left[\|u_t(x_t|\boldsymbol{x}) - u_t(x_t|\mathbf{c})\|_2^2 \,\middle|\, \mathbf{c}\right] = 0$$

for each fixed $\mathbf{c}$ and t. Therefore:

$$\mathbb{E}_{t,x\sim\pi(\boldsymbol{x}),\mathbf{c}\sim q^*(\mathbf{c}|\boldsymbol{x}),x_t\sim p_t(x_t|\boldsymbol{x})}\|u_t(x_t|\boldsymbol{x}) - u_t(x_t|\mathbf{c})\|_2^2 = 0. \tag{23}$$

Step 4: Application of Proposition 3.2. Since the condition of Proposition 3.2 is satisfied under Assumption (A4), and all regularity conditions from Lemma A.1 hold (Lipschitz continuity of δ_{data} and appropriate integrability conditions for exchange of integrals), we conclude that the push-forward flows satisfy:

$$\phi(x_0, \mathbf{c}, \boldsymbol{x}) = \phi(x_0, \mathbf{c}) \tag{24}$$

for all x_0. Therefore, the optimal coupling is preserved:

$$\Pi(u)^\star = \Pi^\star(\boldsymbol{x}_{1:T}). \tag{25}$$

□

Some Practical Considerations : **Assumption (A4)** can be approximately satisfied through the following training strategy:

- **Joint optimization:** Train both the encoder E_ψ and the velocity field components (v_ϕ and the physics-informed f_p) to jointly minimize $\mathcal{L}_{\text{VGB-DM}}(\phi, \psi)$.
- **Adequate capacity:** Make sure the latent dimension $d_c = \dim((\theta, z))$ is large enough to encapsulate information specific to the trajectory (it acts as a sufficient statistic for trajectory), encompassing the parameters of the physics model.

C.3 ELBO Derivation

$$\begin{aligned}
\log p_t^\phi(\dot{x}_t \mid x_k) &= \log \int p_t^\phi(\dot{x}_t, \theta, z \mid x_k)\, d\theta\, dz && (26)\\
&= \log \int q_\psi(z, \theta \mid \boldsymbol{x}) \frac{p_t^\phi(\dot{x}_t, \theta, z \mid x_k)}{q_\psi(z, \theta \mid \boldsymbol{x})}\, d\theta\, dz && (27)\\
&\geq \int q_\psi(z, \theta \mid \boldsymbol{x}) \log \frac{p_t^\phi(\dot{x}_t, \theta, z \mid x_k)}{q_\psi(z, \theta \mid \boldsymbol{x})}\, d\theta\, dz \quad \text{(Jensen's inequality)} && (28)\\
&= \int q_\psi(z, \theta \mid \boldsymbol{x}) \log \frac{p_t^\phi(\dot{x}_t \mid \theta, z, x_k) p_t^\phi(\theta, z \mid x_k)}{q_\psi(z, \theta \mid \boldsymbol{x})}\, d\theta\, dz && (29)\\
&= \mathbb{E}_{q_\psi(z,\theta|\boldsymbol{x})}\left[\log p_t^\phi(\dot{x}_t, \theta, z \mid x_k) - \log \frac{q_\psi(z, \theta \mid \boldsymbol{x})}{p_t^\phi(\theta, z \mid x_k)}\right] && (30)\\
&= \mathbb{E}_{q_\psi(z,\theta|\boldsymbol{x})}[\log p_t^\phi(\dot{x}_t \mid \theta, z, x_k)] - \mathrm{KL}\left[q_\psi(z, \theta \mid \boldsymbol{x}) \| p_t^\phi(\theta, z \mid x_k)\right] && (31)
\end{aligned}$$

Assuming independent and data and time independent priors:

$$p_t^\phi(\theta, z \mid x_k) = p(\theta)p(z) \tag{32}$$

Assuming joint conditional **dependency** of:

$$q_\psi(z, \theta \mid \boldsymbol{x}) = q_\psi(\theta \mid z, \boldsymbol{x}) q_\psi(z \mid \boldsymbol{x}) \tag{33}$$

The objective becomes:

$$\log p_t^\phi(\dot{x}_t \mid x_k) \geq \tag{34}$$
$$\mathbb{E}_{q_\psi(z,\theta|\boldsymbol{x})}[\log p_t^\phi(\dot{x}_t \mid \theta, z, x_k)] - \mathbb{E}_{q_\psi(z|\boldsymbol{x})}\mathrm{KL}\left[q_\psi(\theta \mid z, \boldsymbol{x}) \| p(\theta)\right] - \mathrm{KL}\left[q_\psi(z \mid \boldsymbol{x}) \| p(z)\right]$$

C.4 Reduced Complexity and Optimization Stability

We provide an additional intuition for the empirical sample efficiency and optimization stability observed in our method. These improvements arise from two complementary mechanisms: a reduction in functional complexity induced by the incorporation of an incomplete physics model and a more favourable optimisation landscape enabled by a simulation-free training objective.

Reduced Functional Complexity. Compared to fully black-box models such as BB-NODE or VBB-DM, our approach benefits from a strong inductive bias introduced through the incomplete physics model $f_p(x, \theta)$. In black-box formulations, the learning algorithm must explore a large function space to recover the full system dynamics from data alone. In contrast, by explicitly integrating f_p, we impose a structural constraint that encodes known physical principles. As a result, the learning problem is simplified: rather than discovering physical laws from scratch, the model focuses on inferring a low-dimensional set of unknown parameters θ and latent variables z, while learning a residual or corrective dynamics term v_ϕ. This reduction in hypothesis space directly contributes to improved sample efficiency.

Optimization Landscape and Simulation-Free Training. While physics-informed latent variable models such as PhysVAE also benefit from reduced functional complexity, our method exhibits faster convergence and improved empirical sample efficiency. This difference can be attributed to the choice of training objective. PhysVAE relies on a simulation-based loss that requires integrating the system dynamics through a numerical ODE solver over a time horizon T, of the form

$$\mathcal{L}_{\text{sim}} \approx \left\| X - \text{ODESolve}(v^\phi \circ f_p, x_0, T) \right\|^2.$$

The gradient $\nabla \mathcal{L}_{\text{sim}}$ must be computed by backpropagating through the numerical solver, resulting in a computational cost of $\mathcal{O}(L)$, where L denotes the number of function evaluations performed by the solver.

In contrast, our method optimizes a simulation-free dynamics matching objective, which directly aligns the learned vector field with a target velocity $\dot{x}_t$ obtained from data interpolation:

$$\mathcal{L}_{\text{DM}} \approx \mathbb{E}_t \left[\left\| (v_t^\phi \circ f_p) - \dot{x}_t \right\|^2 \right].$$

The gradient $\nabla \mathcal{L}_{\text{DM}}$ is computed via standard backpropagation and requires only a single forward pass through the network, yielding $\mathcal{O}(1)$ computational complexity per update.

Lemma 1 - Gradient Variance and Convergence. Beyond computational cost, the simulation-free formulation leads to more stable optimization dynamics. Estimating $\nabla \mathcal{L}_{\text{sim}}$ is known to be challenging due to the need to differentiate through a numerical ODE solver. This process can introduce high gradient variance, susceptibility to exploding or vanishing gradients—particularly for long integration horizons or stiff systems—and the accumulation of numerical errors. In contrast, $\nabla \mathcal{L}_{\text{DM}}$ corresponds to a supervised regression gradient, which avoids integration-induced instabilities.

Formally, this suggests that the variance of the gradient estimator under the simulation-free objective is lower than that of the simulation-based objective,

$$\text{Var}(\nabla \mathcal{L}_{\text{DM}}) < \text{Var}(\nabla \mathcal{L}_{\text{sim}}),$$

which in turn implies faster convergence rates under stochastic optimization. This reduction in gradient variance, combined with the reduced functional complexity discussed above, explains the observed improvements in both optimization stability and sample efficiency.