Journal of Computational Physics 491 (2023) 112342

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Adaptive weighting of Bayesian physics informed neural networks for multitask and multiscale forward and inverse problems

Sarah Perez [a], Suryanarayana Maddu [b,c,d,e,f], Ivo F. Sbalzarini [b,c,d,e], Philippe Poncet [a,*]

[a] Universite de Pau et des Pays de l'Adour, E2S UPPA, CNRS, LMAP UMR CNRS-UPPA 5142, Pau, France
[b] Technische Universität Dresden, Faculty of Computer Science, Nöthnitzer Str. 46, 01062 Dresden, Germany
[c] Max Planck Institute of Molecular Cell Biology and Genetics, Pfotenhauerstr. 108, 01307 Dresden, Germany
[d] Center for Systems Biology Dresden, Pfotenhauerstr. 108, 01307 Dresden, Germany
[e] Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI), Dresden/Leipzig, Germany
[f] Center for Computational Biology, Flatiron Institute, 162 5th Avenue, New York NY 10010, USA

ARTICLE INFO

Article history:
Received 27 February 2023
Received in revised form 23 June 2023
Accepted 29 June 2023
Available online 7 July 2023

Keywords:
Hamiltonian Monte Carlo
Uncertainty Quantification
Multi-objective training
Adaptive weight learning
Artificial Intelligence
Bayesian physics-informed neural networks

ABSTRACT

In this paper, we present a novel methodology for automatic adaptive weighting of Bayesian Physics-Informed Neural Networks (BPINNs), and we demonstrate that this makes it possible to robustly address multi-objective and multiscale problems. BPINNs are a popular framework for data assimilation, combining the constraints of Uncertainty Quantification (UQ) and Partial Differential Equation (PDE). The relative weights of the BPINN target distribution terms are directly related to the inherent uncertainty in the respective learning tasks. Yet, they are usually manually set a-priori, that can lead to pathological behavior, stability concerns, and to conflicts between tasks which are obstacles that have deterred the use of BPINNs for inverse problems with multiscale dynamics.

The present weighting strategy automatically tunes the weights by considering the multitask nature of target posterior distribution. We show that this remedies the failure modes of BPINNs and provides efficient exploration of the optimal Pareto front. This leads to better convergence and stability of BPINN training while reducing sampling bias. The determined weights moreover carry information about task uncertainties, reflecting noise levels in the data and adequacy of the PDE model.

We demonstrate this in numerical experiments in Sobolev training, and compare them to analytically ε-optimal baseline, and in a multiscale Lotka-Volterra inverse problem. We eventually apply this framework to an inpainting task and an inverse problem, involving latent field recovery for incompressible flow in complex geometries.

1. Introduction

Direct numerical simulation relies on appropriate mathematical models, derived from physical principles, to conceptualize real-world behavior and provide an understanding of complex phenomena. Experimental data are mainly used for

* Corresponding author.
E-mail address: philippe.poncet@univ-pau.fr (P. Poncet).

https://doi.org/10.1016/j.jcp.2023.112342

parameter identification and a-posteriori model validation. However, a wide range of real-world applications are characterized by the absence of predictive physical models, notably in the life sciences. Data-driven inference of physical models has therefore emerged as a complementary approach in those applications [32]. The same is true for applications that rely on data assimilation and inverse modeling, for example in the geosciences. This has established data-driven models as complementary means to theory-driven models in scientific applications.

Depending on the amount of data available, several data-driven modeling strategies can be chosen. An overview of the state of the art in data-driven modeling, as well as of the remaining challenges, has recently been published [13] with applications focusing on porous media research. It covers methods ranging from model inference using sparse regression [8,14,52,31], where the symbolic structure of a Partial Differential Equation (PDE) model is inferred from the data, to equation-free forecasting models based on extrapolation of observed dynamics [40,61,33]. Therefore, model inference methods are available for both physics-based and equation-free scenarios.

A popular framework combining both scenarios are Physics Informed Neural Networks (PINNs) [46]. They integrate potentially sparse and noisy data with physical principles, such as conservation laws, expressed as mathematical equations. These equations regularize the neural network while the network weights $\theta \in \mathbb{R}^d$ and unknown equation coefficients $\Sigma \in \mathbb{R}^p$ are inferred from data. This has enabled the use of PINNs as surrogate models, for example in fluid mechanics [39,56,43]. Overall, PINNs provide an effective alternative to purely data-driven methods, since a lack of high-fidelity data can be compensated by physical regularization [27,37].

Despite their effectiveness and versatility, PINNs can be difficult to use correctly, as they are prone to a range of training instabilities. This is because their training amounts to a weighted multi-objective optimization problem for the joint set of parameters $\Theta = \{\theta, \Sigma\}$,

$$\widehat{\Theta} = \arg\min_{\Theta} \sum_{k=0}^{K} \lambda_k \mathcal{L}_k(\Theta), \tag{1}$$

where each term $\mathcal{L}_k(\Theta)$ of the loss function corresponds to a distinct inference task. For typical PINNs, these tasks include: data fitting, PDE residual minimization, boundary and initial condition matching, and additional physical constraints such as divergence-freeness of the learned field. Such a combination of measurement and physical constraints, usually encountered in the PINN formulation, readily results in multi-objective problems with tens of different tasks. Proper training of this multitask learning problem hinges on correctly setting the loss term weights λ_k [34]. An unsuitable choice of weights can lead to biased optimization [45], vanishing task-specific gradients [53,10], or catastrophic forgetting [34]. Automatically optimizing the loss weights is crucial and not straightforward, especially if the multi-objective problem is highly nonlinear and suffers from multiscale issues.

The problem of how to tune the loss weights of a PINN is widely known and several potential solutions have been developed to balance the objectives [9,63,64,34]. This offers criteria to impartially optimize the different tasks and provide a good exploration of the optimal Pareto front [49]. While it improves reliability by reducing optimization bias, several open questions remain regarding the confidence in the predictions, noise estimates, and model adequacy [17,37,70]. These questions motivate a need for uncertainty quantification (UQ) to ensure trustworthy inference, extending PINNs to Bayesian inference in the form of Bayesian Physics-Informed Neural Networks (BPINNs) [69,26]. How to adapt successful PINN weighting strategies to BPINNs and integrate them with UQ, however, is still an open problem.

BPINNs benefit from the combined advantages of PINN structures in building parameterized surrogate models and Bayesian inference standards in estimating target distribution. They enable integration of UQ by providing posterior distribution estimates of the predictions — also known as Bayesian Model Averages [65] — based on Markov Chain Monte Carlo (MCMC) sampling. One of the most popular MCMC schemes for BPINNs is Hamiltonian Monte Carlo (HMC), which provides a particularly efficient sampler for high-dimensional inference problems with smooth (physical) dynamics [4]. Although HMC has been shown to be more efficient for BPINNs, its formulation implies potential energy that is closely related to the cost function of the PINN. The multi-objective loss of a PINN directly translates to multi-potential energy and thus to a weighted multitask posterior distribution for BPINN sampling. Sampling from such a target posterior distribution resulting from dozens of distinct tasks, which may be conflicting or involve different scales, is not straightforward. Unsuitable weights can even prevent the sampler from identifying the mode neighborhood in the parameter space that maximizes the overall posterior density. Therefore, it suffers from the same difficulties as a PINN to avoid bias in the sampling and provide an efficient exploration of the Pareto front, characterizing the region of the highest posterior probability.

This often causes HMC to not correctly explore the Pareto front neighborhood during BPINN training. Efficient exploration of a high-dimensional Pareto front remains challenging for multitask and multiscale learning problems incorporating UQ and has not yet been addressed in the Bayesian case. The challenge arises because each term of the multi-potential energy is weighted within the Bayesian framework by parameters that relate to scaling, noise magnitude, and ultimately the inherent uncertainties in the different learning tasks [10]. While these weights are recognized as critical parameters in the sampling procedure, they are mostly hand-tuned [38,26,37], introducing hidden priors when the true uncertainties are not known. Not relying on such a-priori manual calibration of the posterior distribution becomes critical when considering multitask inference problems with potential nonlinear and multiscale effects. In such cases, appropriately setting these parameters is neither easy nor computationally efficient and can lead to either a biased estimation of the uncertainties or a considerable

waste of energy in ineffective tuning. Properly optimizing these weights is therefore essential to ensure that HMC samples from the posterior distribution around the Pareto front. This is not only required for robust BPINN training but also for enhanced reliability of the UQ estimates, not subject to biased priors.

In order to robustly handle multitask UQ inference in BPINNs, the open questions addressed in this article are: How can we automatically adjust the weights in BPINNs to efficiently explore the Pareto front and avoid bias in the UQ inference? How can we manage sensitivity to the noise distributions (homo- or hetero-scedastic) and their amplitude, without imposing hidden priors, to ensure reliable uncertainties estimates?

We intend to provide a reliable framework to address multitask Bayesian inference problems, with potential multiscale effects, stiffness, or competing tasks, such that we do not rely on a-priori hand-tuning or biased calibration of the posterior distribution. We aim to automatically and adaptively build the weighted posterior distribution that concentrates on the region of highest posterior probability, localized in the Pareto front neighborhood. To the best of our knowledge, this manuscript's main novelty is to address the problem of appropriate weighting in multitask inference problems from the posterior distribution perspective, which becomes crucial when dealing with physics-based inference. Related state-of-the-art adaptive methods regard the problem from the adaptation of some hyperparameters (e.g. the leapfrog parameters in the No-U-Turn sampler [18]) or the adaptation of the momentum based on local geometrical information (e.g. Riemann Hamiltonian Monte Carlo [16]). Other strategies aim to speed up the Bayesian inversion and reduce the computational cost of evaluating the forward model itself, with some adaptations based on multi-fidelity surrogate models either computed through polynomial chaos expansion [67] or neural network proxies [68]. These adaptations are usually performed through online and local refinement of the surrogate models using only a few simulations from the high-fidelity model. However, all these adaptive methods do not consider the case where we face multi-objective problems with potentially conflicting tasks or multiscale issues.

We start by characterizing potential BPINN failure modes, which are particularly prevalent for multiscale or multitask inverse problems. We then propose a modified HMC sampler, called Adaptively Weighted Hamiltonian Monte Carlo (AW-HMC), which avoids the problem by balancing gradient variances during sampling. We show that this leads to a weighted posterior distribution that is well suited to exploring the Pareto front neighborhood, since it concentrates on the region of highest posterior probability by leveraging gradient information of the different tasks during the adaptation procedure. Our benchmarks show that this strategy reduces sampling bias and enhances the BPINN robustness. In particular, our method improves the stability along the leapfrog steps during training, since it ensures optimal integration and frees the sampling from excessive time step decrease. Moreover, it is able to automatically adjust the potential energy weights and with them, the uncertainties according to the sensitivity to noise of each term and their different scaling. This considerably improves the reliability of the UQ by reducing the need for hyperparameter tuning, including the prior distributions, and reducing the need for prior knowledge of noise levels or appropriate task scaling. We show that this improves BPINNs with respect to both the convergence rate and the computational cost. Moreover, we introduce a new metric for the quality of the prediction, quantifying the convergence rate during the marginalization step. We finally demonstrate that our proposed approach enables the use of BPINNs in multiscale and multitask Bayesian inference over complex real-world problems with sparse and noisy data.

The remainder of this manuscript is organized as follows: In Sect. 2, we review the general principles of BPINNs and the HMC sampler and characterize their failure mode in Sobolev training. Sect. 3 describes the proposed adaptive weighting strategy for UQ using BPINNs. We validate this strategy in a benchmark with a known analytical solution in Sect. 3.2 and Appendix B. We then demonstrate the effectiveness of the proposed AW-HMC algorithm on a Lotka-Volterra inverse problem in Sect. 3.3, focusing on a multiscale inference of dynamical parameters. We then illustrate the use of AW-HMC in a real-world problem from fluid dynamics in Sect. 4. This particularly demonstrates successful inpainting of incompressible stenotic blood flow from sparse and noisy data, highlighting UQ estimates consistent with the noise level and noise sensitivity. Finally, Sect. 4.2 considers an inverse flow problem in a complex geometry, where we infer both the flow regime (the inverse Reynolds number) and the latent pressure field from partial velocity measurements. We conclude and discuss our observations in Sect. 5.

2. From Uncertainty Quantification to Bayesian Physics-Informed Neural Networks: concepts and limitations

Real-world applications of data-driven or black-box surrogate models remain a challenging task. Predictions often need to combine prior physical knowledge, whose reliability can be questioned, with sparse and noisy data exhibiting measurement uncertainties. These real-world problems also suffer from non-linearity [26], scaling [11], and stiffness [22] issues that can considerably impact the efficiency of the usual methodologies. This needs the development of data-driven modeling strategies that robustly address these issues.

At the same time, the need to build upon Bayesian inference raises the question in the research community of ensuring trustable intervals in the estimations. This is important for quantifying uncertainties on both the underlying physical model and the measurement data, although it may be challenging in the context of stiff, multiscale, or multi-fidelity problems. Therefore, embedding UQ in the previous data-driven methodologies is essential to effectively manage real-world applications.

2.1. HMC-BPINN concepts and principles

The growing popularity of Bayesian Physics-Informed Neural Networks [69,26,25,38,29] offers the opportunity to incorporate uncertainty quantification into PINNs standards, and benefit from their predictive power. It features an interesting Bayesian framework that claims to handle real-world sparse and noisy data and, as well, it bestows reliability on the models together with the predictions.

The basic idea behind a BPINN is to consider each unknown, namely the neural network and inverse parameters, Θ, as random variables with specific distributions instead of single parameters as for a PINN. The different sampling strategies all aim to explore the posterior distribution of Θ

$$P(\Theta|\mathcal{D},\mathcal{M}) \propto P(\mathcal{D}|\Theta)P(\mathcal{M}|\Theta)P(\Theta) \tag{2}$$

through a marginalization process, given some measurement data $\mathcal{D}$ and a presumed model $\mathcal{M}$, rather than looking for the best approximation satisfying the optimization problem (1). The posterior distribution expression (2) is obtained from Bayes theorem and basically involves a data-fitting likelihood term $P(\mathcal{D}|\Theta)$, a PDE-likelihood term $P(\mathcal{M}|\Theta)$ and a joint prior distribution $P(\Theta)$. These specific terms are detailed, case-by-case, in the applications, along with the different sections. The Bayesian marginalization then transfers the distribution of the parameters Θ into a posterior distribution of the predictions, also known as a Bayesian Model Average (BMA):

$$\underbrace{P(y|x,\mathcal{D},\mathcal{M})}_{\text{predictive BMA distribution}} = \int \underbrace{P(y|x,\Theta)}_{\text{prediction for }\Theta} \underbrace{P(\Theta|\mathcal{D},\mathcal{M})}_{\text{posterior}} \mathrm{d}\Theta \tag{3}$$

where x and y respectively refer to the input (e.g. spatial and temporal points) and output (e.g. field prediction) of the neural network. In this equation, the different predictions arising from all the Θ parameters sampling (2) are weighted by their posterior probability and averaged to provide an intrinsic UQ of the BPINN output. Overall, BPINNs introduce a Bayesian marginalization of the parameters Θ which forms a predictive distribution (3) of the quantities of interest (QoI), namely the learned fields and inverse parameters.

Different approaches were developed for Bayesian inference in deep neural networks including Variational Inference [66, 28] and Markov Chain Monte Carlo methods. A particular MCMC sampler based on Hamiltonian dynamics — the Hamiltonian Monte Carlo (HMC) — has drawn increasing attention due to its ability to handle high-dimensional problems by taking into account the geometric properties of the posterior distribution. Betancourt explained the efficiency of HMC through a conceptual comprehension of the method [4] and theoretically demonstrated the ergodicity and convergence of the chain [6,30]. From a numerical perspective, Yang et al. [69] highlighted the out-performance of BPINNs-HMC formulation on forward and inverse problems compared to its Variational Inference declination. This has established HMC as a highly effective MCMC scheme for the BPINNs, both theoretically and numerically.

In the following, we briefly review the basic principles of the classical BPINNs-HMC and point out their limitations, especially in the case of multi-objective and multiscale problems.

The idea of HMC is to assume a fictive particle of successive positions and momenta (Θ, r) which follows the Hamiltonian dynamics on the frictionless negative log posterior (NLP) geometry. It requires the auxiliary variable r to immerse the sampling of (2) into the exploration of a joint probability distribution $\pi(\Theta, r)$ in the phase space

$$\pi(\Theta, r) \sim \mathrm{e}^{-H(\Theta,r)}. \tag{4}$$

The latter relies on a particular decomposition of the Hamiltonian $H(\Theta, r) = U(\Theta) + K(r)$ where the potential and kinetic energies, $U(\Theta)$ and $K(r)$ respectively, are chosen such that

$$\pi(\Theta, r) \propto P(\Theta|\mathcal{D},\mathcal{M})\mathcal{N}(r|0,\mathbf{M}) \tag{5}$$

and the momentum follows a centered multivariate Gaussian distribution, with a covariance — or mass — matrix $\mathbf{M}$ often scaled identity. The Hamiltonian of the system is thus given by

$$H(\Theta, r) = U(\Theta) + \frac{1}{2} r^T \mathbf{M}^{-1} r \tag{6}$$

where the potential energy directly relates to the target posterior distribution. This energy term is usually expressed as the negative log posterior $U(\Theta) = -\ln P(\Theta|\mathcal{D},\mathcal{M})$, which results in a multi-potential as detailed in Sect. 2.2. This ensures that the marginal distribution of Θ provides immediate samples of the target posterior distribution

$$P(\Theta|\mathcal{D},\mathcal{M}) \sim \mathrm{e}^{-U(\theta)} \tag{7}$$

since an efficient exploration of the joint distribution $\pi(\Theta, r)$ directly projects to an efficient exploration of the target distribution, as described by Betancourt [4]. The HMC sampling process alternates between deterministic steps, where we solve for the path of a frictionless particle given the Hamiltonian dynamical system

$$\begin{cases} \mathrm{d}\Theta = \mathbf{M}^{-1} r \, \mathrm{d}t \\ \mathrm{d}r = -\nabla U(\Theta) \, \mathrm{d}t, \end{cases} \tag{8}$$

and stochastic steps, where the momentum is sampled according to the previously introduced Gaussian distribution. As Hamilton's equation (8) theoretically preserves the total energy of the system, each deterministic step is then constrained to a specific energy level while the stochastic steps enable us to diffuse across the energy level set for efficient exploration in the phase space. This theoretical conservation of the energy level set during the deterministic steps requires numerical schemes that ensure energy conservation.

A symplectic integrator is thus commonly used to numerically solve for the Hamiltonian dynamics (8): the Störmer-Verlet also known as the leapfrog method. However, these integrators are not completely free of discretization errors that may disrupt, in practice, the Hamiltonian conservation through the deterministic iterations. Hence, a correction step is finally added in the process to reduce the bias induced by these discretization errors in the numerical integration: this results in a Metropolis-Hasting criterion based on the Hamiltonian transition. This acceptance criterion tends to preserve energy by rejecting samples that lead to divergent probability transition. The exploration of the deterministic trajectories though remains sensitive to two specific hyperparameters managing the integration time: the step size δt and the number of iterations L used in the leapfrog method. Tuning these parameters can be challenging, especially if the posterior distribution presents pathological or high curvature regions [4], yielding instability, under-performance, and poor validity of the MCMC estimators. Despite the use of numerical schemes that preserve the Hamiltonian properties, a conventional HMC-BPINN can be confronted with pathological discrepancies.

To counteract these divergence effects, efforts have been put into developing strategies to either adaptively set the trajectory length L [19] while preserving detailed balance condition or use standard adaptive-MCMC approaches to adjust the step size δt on the fly [5]. In this regard, one of the most popular adaptive strategies is the No-U-Turn sampler (NUTS) from Hoffmann and Gelman [18]. Nonetheless, these divergent trajectories indicate significant bias in the MCMC estimation even if such adaptive methods may offer an alternative to overcome them. This raises the question of the validity of this adaptation when facing multi-potential energy terms that lead to significantly different geometrical behaviors or different scaling in the posterior distribution. In fact, the adaptive strategy mostly tunes the leapfrog parameters so that the most sensitive term respects the energy conservation, which may result in poorly-chosen hyper-parameters for the other potential energy terms, and then the whole posterior distribution. This reflects the limitations of such adaptive strategies that rely on adjusting the leapfrog hyperparameters.

When these divergent pathologies become prevalent, another approach suggested by Betancourt [4] is to regularize the target distribution, which can become strenuous in real-world applications and lead to additional tuning. Nevertheless, it offers a great opportunity to investigate the impact of each learning task on the overall behavior of the target distribution and paves the way for novel adaptive weighting strategies.

In the next sections, we focus particularly on the challenges arising from real-world multitasks and multiscale paradigms. We show that present BPINN methods result in major failures in these cases and we identify the main pathologies using powerful diagnostics based on these divergent probability transitions.

2.2. The multi-objective problem paradigm

As for the issue of the multi-objective optimization problem in a PINN, sampling of the target posterior distribution (2) arising from a direct or inverse problem requires the use of a multi-potential energy term $U(\Theta)$. Furthermore, in real-world applications, we have to deal with sparse and noisy measurements whose fidelity can also cover different scales: this is the case of multi-fidelity problems with multi-source data [27,37].

For sake of generality, we introduce a spatio-temporal domain $\Omega = \widetilde{\Omega} \times \mathcal{T}$ with $\widetilde{\Omega} \subset \mathbb{R}^n$, $n = 1, 2, 3$ and we assume a PDE system in the following form:

$$\begin{cases} \mathcal{F}(u(t,x),\ \Sigma) &= 0, \qquad (t,x) \in \Omega \\ \mathcal{H}(u(t,x),\ \Sigma) &= 0, \qquad (t,x) \in \Omega \\ \mathcal{B}(u(t,x),\ \Sigma) &= 0, \qquad (t,x) \in \Omega^{\partial} := \partial\widetilde{\Omega} \times \mathcal{T} \\ \mathcal{I}(u(t,x),\ \Sigma) &= 0, \qquad (t,x) \in \Omega^{I} := \widetilde{\Omega} \times \mathcal{T}_0 \end{cases} \tag{9}$$

where u is the principal unknown, $\mathcal{F}$ the main differential equation (e.g. the Navier-Stokes equation), $\mathcal{H}$ an additional constraint (e.g. incompressibility condition), $\mathcal{B}$ and $\mathcal{I}$ the boundary and initial conditions respectively, and Σ the PDE model parameters, either known or inferred. Some partial measurements of the solution field u may also be available in a subset $\Omega^u \subset \Omega$. Such a continuous description of the spatio-temporal domain is then discretized to enable the selection of the training dataset, which is used in BPINNs sampling.

We first define the dataset $\mathcal{D}$ of training data which is decomposed into $\mathcal{D} = \mathcal{D}^{\Omega} \cup \mathcal{D}^{\partial} \cup \mathcal{D}^{I} \cup \mathcal{D}^{u}$ and includes scattered and noisy measurements sampled in their respective sets Ω, Ω^{∂}, Ω^{I}, and Ω^{u}. Regarding data corruption, we consider independent Gaussian noise for the sparse observations on u, such that $\mathcal{D}^{u}$ is defined as

$$\mathcal{D}^{u} = \left\{ (t_i, x_i, u_i), \quad \text{s.t.} \quad (t_i, x_i) \in \Omega^{u} \quad \text{and} \quad u_i := u(t_i, x_i) + \xi_u(t_i, x_i),\ i = 1 \ldots N^{u} \right\} \tag{10}$$

where the noise $\xi_u \sim \mathcal{N}(0, \sigma_u^2 I)$ and the standard deviation σ_u might be estimated from the sensor fidelity, if accessible. The neural network component of the BPINN then provides a surrogate model of u denoted u_Θ for each sample of the parameters $\Theta = \{\theta, \Sigma\}$, whose prior distribution is referred to as $P(\Theta)$. The latter takes into account both the priors on the neural network parameters θ, which are assumed to be centered and independent Gaussian distributions, and the priors on the model parameters Σ, so that $P(\Theta) = P(\theta)P(\Sigma)$ under the independence condition. In the case of a forward problem, where the PDE model parameters are prescribed, the prior distribution reduces to $P(\theta)$. When some measurements of the unknown are available, meaning $\mathcal{D}^u$ is not an empty set, which is the case in inverse or inpainting problems, then the surrogate model u_Θ should satisfy a data-fitting likelihood term in the Bayesian framework. This consists in quantifying, over the set $\mathcal{D}^u$, the fit between the neural network prediction and the training data defined by:

$$P(\mathcal{D}^u|\Theta) \propto \prod_{i=1}^{N^u} \exp\left(\frac{-(u_\Theta(t_i, x_i) - u_i)^2}{2\sigma_u^2}\right). \tag{11}$$

Similarly, the boundary conditions of the model output are imposed on the set $\mathcal{D}^\partial$

$$\mathcal{D}^\partial = \left\{(t_i, x_i, \mathcal{B}(u_i)), \quad \text{s.t.} \quad (t_i, x_i) \in \Omega^\partial \quad \text{and} \quad \mathcal{B}(u_i) := \mathcal{B}(u(t_i, x_i)) + \xi_b(t_i, x_i),\ i = 1...N^\partial\right\} \tag{12}$$

by satisfying the following boundary-likelihood term

$$P(\mathcal{D}^\partial|\Theta) \propto \prod_{i=1}^{N^\partial} \exp\left(\frac{-(\mathcal{B}(u_\Theta(t_i, x_i)) - \mathcal{B}(u_i))^2}{2\sigma_b^2}\right). \tag{13}$$

The noise sensitivity on the boundary condition term is also characterized by independent Gaussian distributions in the sense that $\xi_b \sim \mathcal{N}(0, \sigma_b^2 I)$ where the standard deviation σ_b needs to be estimated. Such a distinction between ξ_u and ξ_b is prescribed since there is no guarantee that the data corruption is uniform: in fact, the measurement distribution variances can differ locally when facing heteroscedastic noise. This is the case in geosciences, where data-driven modeling based on X-Ray microtomography images require special attention on this boundary noise estimation ξ_b. This is mainly due to the artifact limitations (e.g. partial volume effect, edge-enhancement) that tend to enhance the blurring effects at the material interface and therefore impact the quantification of the medium effective properties, such as the permeability and micro-porosity [41,2]. The same holds for the initial condition with potentially a different sensitivity ξ_i. In a BPINN, the previous data-fitting terms are complemented with physical principles that regularize the neural network predictions, given the PDE system (9).

Concerning the PDE-likelihood term, the $\mathcal{D}^\Omega$ dataset is defined as the training points on which we force the PDE and the additional physical constraint to be satisfied by the surrogate modeling:

$$\mathcal{D}^\Omega = \left\{(t_i, x_i) \in \Omega, \quad \mathcal{F}(u_\Theta(t_i, x_i)) = \xi_f(t_i, x_i) \quad \text{and} \quad \mathcal{H}(u_\Theta(t_i, x_i)) = \xi_h(t_i, x_i),\ i = 1...N^\Omega\right\} \tag{14}$$

with ξ_f and ξ_h standing for the model uncertainty in both equations, which are usually unknown and can easily lead to physical model misspecification. According to these notations, a forward problem consists in $\mathcal{D}^u = \varnothing$ and Σ is known to perform a direct prediction of the field u_Θ on Ω based only on the PDE physical assumptions. On the contrary, an inverse problem aims to infer Σ using together the PDE model with the partial and noisy information $\mathcal{D}^u$ of the predictive field u. Finally, an inpainting problem relies on these partial measurements to complement and recover some missing information on the predictive field, in addition to the PDE-based priors.

Finally, the target posterior distribution of Θ (2) is decomposed according to the Bayes rule, into a sequence of multitask likelihood terms — involving data-fitting and PDE likelihood — and the priors:

$$P(\Theta|\mathcal{D}, \mathcal{M}) \propto P(\mathcal{D}^u|\Theta)P(\mathcal{D}^\partial|\Theta)P(\mathcal{D}^I|\Theta)P(\mathcal{D}^\Omega, \mathcal{F}|\Theta)P(\mathcal{D}^\Omega, \mathcal{H}|\Theta)P(\theta)P(\Sigma) \tag{15}$$

which results, for the HMC sampler, in the multi-potential energy

$$U(\Theta) = \frac{\|u_\Theta - u\|^2_{\mathcal{D}^u}}{2\sigma_u^2} + \frac{\|\mathcal{B}(u_\Theta) - \mathcal{B}(u)\|^2_{\mathcal{D}^\partial}}{2\sigma_b^2} + \frac{\|\mathcal{I}(u_\Theta) - \mathcal{I}(u)\|^2_{\mathcal{D}^I}}{2\sigma_i^2} + \frac{\|\mathcal{F}(u_\Theta)\|^2_{\mathcal{D}^\Omega}}{2\sigma_f^2} + \frac{\|\mathcal{H}(u_\Theta)\|^2_{\mathcal{D}^\Omega}}{2\sigma_h^2} + \frac{\|\theta\|^2_{\mathbb{R}^d}}{2\sigma_\theta^2} + \frac{\|\Sigma - \mu_\Sigma\|^2_{\mathbb{R}^p}}{2\sigma_\Sigma^2} \tag{16}$$

according to equation (7). The notation $\|\cdot\|$ refers to either the RMS (root mean square) norm — inherited from the functional $\mathbb{L}^2$-norm on the open set Ω — for the log-likelihood terms or to the usual Euclidean norm for the log-prior terms. In addition, the multi-potential (16) is written here, in a general framework, based on the prior assumptions $P(\theta) \sim \mathcal{N}(0, \sigma_\theta^2 I_d)$ and $P(\Sigma) \sim \mathcal{N}(\mu_\Sigma, \sigma_\Sigma^2 I_p)$. We note that the log-prior term can be regarded as a $\mathbb{L}^2$-regularization in the equivalent constrained optimization problem. Nonetheless, suitable selection of these prior distributions — hence appropriate tuning of

the parameters σ_θ, μ_Σ, and σ_Σ — is usually not straightforward and is time-consuming. Overall, equation (16) highlights that, even in a simple problem setup, a BPINN may face a potential energy term that closely resembles a weighted multi-objective loss appearing in a PINN. This multi-potential energy results, according to equation (7), in a multitask weighted posterior distribution for which an appropriate adaptive weighting strategy is critical.

Therefore, the main challenge is to sample near the Pareto-optimal solution such that the BPINNs provide efficient and reliable prediction and UQ. Otherwise, the risk is that the samples obtained gravitate around a local minimum, corresponding to one of the multi-potential terms at the cost of the others. The present work focuses on how to estimate a well-fitted weighted posterior distribution that concentrates on the region of the highest posterior probability. The latter, localized in the Pareto front neighborhood, should ensure balanced conditions for all the different terms in equation (16) and therefore provide robust Bayesian inference for multitask problems with multiscale or multi-objective issues. Effectively dealing with multitask Bayesian inference problems is a distinct concern from capturing multi-modal distributions. In this paper, we intend to address the first issue, while the second one is out of consideration. This comes from the observation that the HMC formulation, which serves as a basis of the present article, is prone to struggle with multi-mode scenarios while having excellent performances in sampling single-mode distributions. The main reason is that HMC relies on local gradient information that may ignore other isolated modes [35]. Efficiently sampling from multi-modal distributions requires other samplers. These alternatives include, for instance, Variational Hybrid Monte Carlo [58], wormhole Hamiltonian Monte Carlo [23], and augmented Markov Chain Monte Carlo with normalizing flow methods for adapting the proposal distribution (transition kernels) during sampling [15], but this is beyond the scope of this work. A way to handle likely multi-modal distributions with the HMC sampler is to consider different initializations of the neural network parameters or inverse parameters in order to incorporate sampling variability. This article focuses on providing a robust sampling strategy for multitask weighted posterior distributions in single-mode scenarios, with potentially multiscale inference in the inverse parameters.

Secondly, while the standard deviations $\sigma_\bullet$ in (16) are critical parameters to select and are related to the uncertainties on the inherent tasks, most of the authors either assign them a given value or train them as additional hyperparameters [38,57,26]. This can lead to highly biased predictions, especially when setting the PDE-residual standard deviations σ_f and σ_h which introduce strong priors on the model adequacy. Recently Psaros et al. [44] discussed, *inter alia*, alternatives generalizing the adjustment of some of these parameters — mainly the data-fitting standard deviations — in the context of unknown and heteroscedastic noise distributions. They either rely on *offline learning* at the cost of a pre-trained Generative Adversarial Network (GAN) or *online learning* of the weights based on additional parameter training. In particular, the number of these additional parameters may increase drastically when considering location-dependent variances, as suggested in [44], for realistic applications and consequently suffer from computational costs. The open question remains on how to deal with such unknown (homo- or hetero-scedastic) noise distributions without adding computational complexity by learning additional hyperparameters. Finally, although the question of physical model misspecification was pointed out in the total uncertainty quantification, the latter has not been addressed in [44] when misleading model uncertainty is assumed on the physical constraints $\mathcal{F}$ and $\mathcal{H}$. As a result, the issue of not introducing strong priors on the model adequacy by hand-tuning the hyperparameters σ_f and σ_h, usually unknown or prescribed, is still a challenging task. Therefore, we consider the question of how to adaptively set the weights in multitask Bayesian inference problems such that we do not rely on a-priori manual calibration of the noise magnitude, task scaling, or model adequacy.

In view of this, we wanted to test the robustness of the usual BPINNs-HMC approach, as introduced in Sect. 2.1, on a test case demonstrating the issues arising from the multi-objective and multiscale nature of the sampling using Sobolev training of neural networks.

2.3. Sobolev training for BPINNs failure mode

Sobolev training is a special case of multi-objective BPINN sampling that likely leads to stiff learning due to the disparate scales involved [34]. Nevertheless, it is commonly used in the machine learning community to improve the prediction efficiency and generalization of neural networks, by adding information about the target function derivatives to the loss or its equivalent potential energy [12,55,62,72].

This special training provides a baseline for testing the robustness of the present BPINNs-HMC method against the failure mode of vanishing task-specific gradients [34]. It also offers the opportunity to benchmark against the analytically ε-optimal weights that are known for Sobolev multi-objective optimization [34].

The BPINNs-HMC sampling is tested here on a Sobolev regression task, which means the dataset is restricted to $\mathcal{D} = \mathcal{D}^u$ involving measurements of a function and its derivatives D_x^k, $k \geq 1$ up to order K, such that the target posterior distribution is

$$P(\Theta|\mathcal{D}) \propto \prod_{k=0}^{K} P(\mathcal{D}^u, D_x^k u|\Theta) P(\Theta) \tag{17}$$

and the potential energy hence has the general form

$$U(\Theta) = \sum_{k=0}^{K} \left[\frac{\lambda_k}{2\sigma_k^2} \| D_x^k u_\Theta - D_x^k u \|^2 \right] + \frac{\lambda_{K+1}}{2\sigma_{K+1}^2} \|\Theta\|^2 := \sum_{k=0}^{K+1} \lambda_k \mathcal{L}_k(\Theta) \tag{18}$$

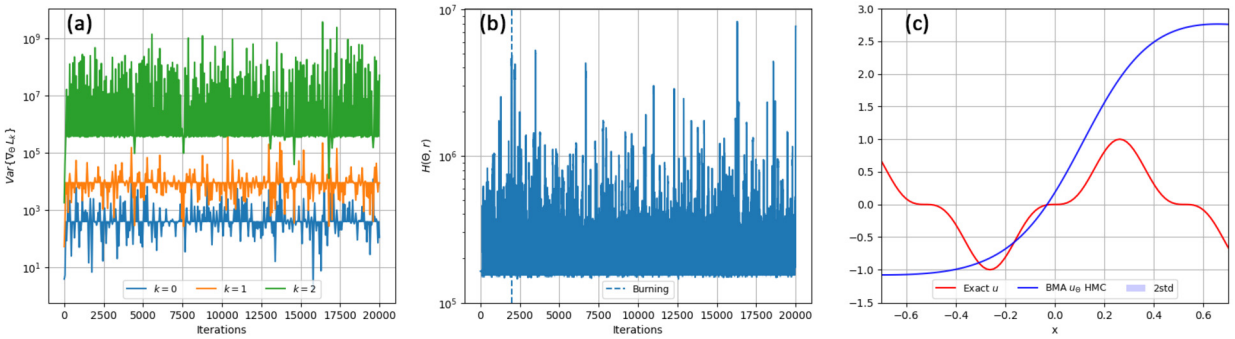

Fig. 1. HMC uniform-weighting failure mode for Sobolev training up to second-order derivatives: (a) Variances of the effective gradient $\nabla_\Theta L_k$ distributions ($k = 0, 1, 2$), plotted with respect to the ($N_s \times L$) HMC iterations, showing strong imbalances between the tasks. (b) Non-conservative Hamiltonian that illustrates the resulting instabilities issues. (c) Extremely poor approximation of the function and non-existent uncertainty estimates due to the massive rejection. (For interpretation of the colors in the figures, the reader is referred to the web version of this article.)

where $L_k = \lambda_k \mathcal{L}_k$ refers to the weighted k^{th} objective term, with λ_k some positive weighting parameters to define (see Sect. 3.1). In this section, we use only a uniform weighting strategy, with $\lambda_k = 1, \forall k$, which corresponds to the classical BPINNs-HMC formulation. For sake of readability, equation (18) gathers the log-prior terms of the neural network and inverse parameters, assuming they all have the same prior distribution.

We first introduce a 1D Sobolev training up to second-order derivatives, with a test function $u(x) = \sin^3(\omega x)$ defined on $\Omega = [-0.7, 0.7]$ for $\omega = 6$. We use 100 training points, set the leapfrog parameters $L = 100$ and $\delta t = 1\text{e}{-3}$ for the number of iterations and time step respectively, and perform $N_s = 200$ sampling iterations. We also restrict the test to a function approximation problem so that subsequently Θ refers only to the neural network parameters. In the following and unless otherwise indicated, all the σ_k are equal to one since in practice we do not have access to the values of these parameters for the derivatives or residual PDE terms, but rather to the observation noise on the data field u only, if available.

Similarly to PINNs, this test case with uniform weights λ_k leads to imbalanced gradient variances between the different objective terms. In particular, the higher-order derivatives present dominant gradient variances that contribute to the vanishing of the other tasks and lead to biased exploration of the posterior distribution. In Fig. 1 (left) we see that the term $\text{Var}\{\nabla_\Theta L_2\}$ corresponding to the higher-order derivative quickly develops two orders of magnitude greater than the other effective gradient variances. In addition to inefficient exploration of the Pareto front, we also face instability issues, generated by the highest order derivative terms, that result in a lack of conservation of the Hamiltonian along the leapfrog trajectories (see Fig. 1 middle). The HMC sampler with uniform weighting strategy fails on this test case, resulting in poor prediction of the surrogate model u_θ (see Fig. 1 right). Indeed, none of the proposed samples are accepted according to the Metropolis-Hasting criterion. Hence, there are neither uncertainty estimates nor adjustments of the predictive BMA distribution. Both are solely based on the surrogate model corresponding to the prior distributions of the neural network parameters (Fig. 1 right). As specified in Sect. 2.1, such divergence pathologies on the classical HMC with uniform weighting are powerful diagnostics of bias in the resulting estimators and raise suspicions about the validity of the latter.

An alternative to counteract these effects consists in reducing the time step δt to balance the order of magnitude of the derivative terms and improve the Metropolis-Hasting acceptance rate of the BPINNs-HMC. However, a small time step within the leapfrog iterations is more likely to generate pathological random walk behaviors or biased sampling [18,4]. To this aim, we attempt an adaptive strategy by using the No-U-Turn sampler (NUTS) with step-size adaptation, as detailed in Algorithm 5 from Hoffmann and Gelman [18] and implemented in the Python Open Source package hamiltorch [11]. We consider the same exact set of leapfrog parameters as previously — in order to comply with the same assumptions — and we impose $N = 20$ adaptive steps that lead to a final adapted time step of $\delta t = 1.29\text{e}{-4}$. In this case, we again reached a configuration where we were not efficiently exploring the Pareto front, as evidenced by the variances of the effective gradients in Fig. 2. This resulted in a better approximation of the second derivative compared to the signal itself and demonstrated biased sampling in the sense that the signal u is determined up to a linear function due to the prevalence of the higher derivative term. This linear deviation is also shown in Fig. 2 — bottom right. This confirms that the NUTS time-step adaptation focuses rather on the prevailing conservation of the higher-order derivative which induced the stiffness.

In short, even a simple 1D Sobolev training with trivial uniform weights induces major failure of the classical BPINNs-HMC approaches because of the sensitivity of the posterior distribution to the higher-order derivatives that generate instabilities. Consequently, such divergence in the Hamiltonian conservation renders the sampling approach inoperative. Moreover, the alternatives ensuring the Hamiltonian conservation are ineffective because they face either inefficient exploration of the energy levels or a strong imbalance in the multitask and multiscale sampling. This suggests that the Hamiltonian Markov chain cannot adequately explore the Pareto front of the target distribution resulting from this potential energy, and that strong imbalanced conditions cannot be overcome with the usual methodologies.

The purpose is therefore to develop a strategy to provide balanced conditions between the different tasks, independently of their scales, by looking for an appropriate weighting formulation. This approach is essential regardless of the usual HMC

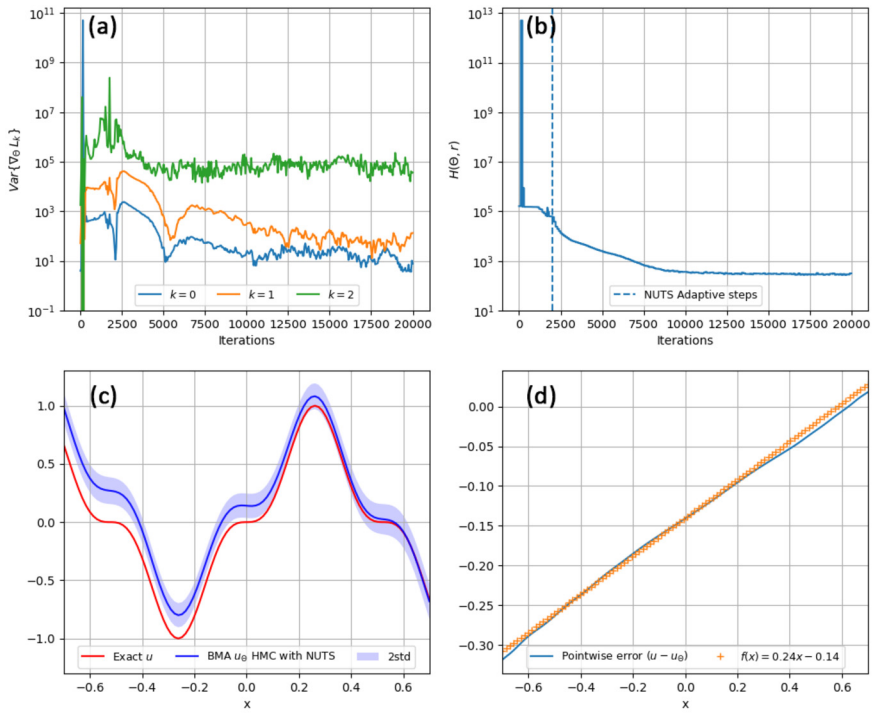

Fig. 2. Failure mode of NUTS step-size adaptation in Sobolev training up to second order derivatives: (a) Variances of the effective gradient $\nabla_\Theta L_k$ distributions, showing task imbalances. (b) Hamiltonian evolution along the sampling, resulting in a weak exploration of the energy levels. The vertical dotted line delimits the number of adaptive steps in the NUTS sampler. (c) BMA signal reconstruction with its uncertainty. (d) Pointwise error between the surrogate BMA reconstruction u_Θ and the function u, highlighting the linear deviation of u_Θ and biased inference of the distinct tasks.

concerns about the adaptive settings of the leapfrog parameters, and presents the advantage of reducing the instabilities without needlessly decreasing the time step.

3. An adaptive weighting strategy for unbiased Uncertainty Quantification in BPINNs

Conventional BPINN formulations exhibit limitations regarding multi-objective inferences, such as stability and stiffness issues, pathological conflicts between tasks, and biased exploration of the Pareto-optimal front. These problems cannot be tackled merely by adaptively setting the leapfrog parameters, as in the NUTS sampler, nor by hand-tuning the standard deviations σ, which introduces additional computational costs or energy waste. We therefore investigate another adaptive approach that focuses instead on the direct regularization of the target distribution: it aims to balance task weighting by automatically adapting the critical σ parameters.

3.1. An Inverse Dirichlet Adaptive Weighting algorithm: AW-HMC

The development of a new alternative considering the limits of the HMC-BPINN approach (previously discussed in Sect. 2) becomes crucial, especially in the case of complex multi-objective problems arising from real-world data. This strategy must address the main pathologies identified by: 1) ensuring the exploration of the Pareto front of the target posterior distribution, 2) managing the scaling sensitivity of the different terms, and 3) controlling the Hamiltonian instabilities.

Independently of these pathological considerations, there remains the issue of setting the critical σ parameters, particularly when the level of noise on the data and the confidence in the PDE model are not prior knowledge. While manual tuning of these parameters is still commonplace, we could rely on the λ weight adaptations to implicitly determine the noise and inherent task uncertainties rather than introduce strong priors on the model adequacy that may lead to misleading predictions.

In order to fulfill all these requirements, we consider an Inverse-Dirichlet based approach that has demonstrated its effectiveness in the PINNs framework when dealing with balanced training and multiscale modeling [34]. It relies on adjusting the weights based on the variances of the loss term gradients, which can be interpreted as a training uncertainty with respect to the main descent direction in a high-dimensional multi-objective optimization problem. This strategy also offers considerable improvement in convergence over conventional training and avoids the vanishing of specific tasks.

The idea of developing an Inverse Dirichlet adaptively weighted algorithm for BPINNs is to incorporate such training uncertainties on the different tasks within the Bayesian framework so that it can simultaneously take into account the noise,

the model adequacy and the sensitivity of the tasks, all while ensuring Pareto front exploration. Therefore, we are trying to determine the positive weighting parameters λ_k, $k = 0, ..., K$ in such a way that the weighted gradient $\nabla_\Theta L_k = \lambda_k \nabla_\Theta \mathcal{L}_k$ distributions of the potential energy terms have balanced variances. We propose to ensure gradient distributions with the same variance

$$\gamma^2 := \mathrm{Var}\{\lambda_k \nabla_\Theta \mathcal{L}_k\} \simeq \min_{t=0..K} (\mathrm{Var}\{\nabla_\Theta \mathcal{L}_t\}), \quad \forall k = 0, ..., K \tag{19}$$

by setting the weights on an Inverse-Dirichlet based approach:

$$\lambda_k = \left(\frac{\min_{t=0..K} \left(\mathrm{Var}\{\nabla_\Theta \mathcal{L}_t\}\right)}{\mathrm{Var}\{\nabla_\Theta \mathcal{L}_k\}} \right)^{1/2} = \left(\frac{\gamma^2}{\mathrm{Var}\{\nabla_\Theta \mathcal{L}_k\}} \right)^{1/2} \tag{20}$$

such that

$$\lambda_k \mathcal{N}(\mu_k, \mathrm{Var}\{\nabla_\Theta \mathcal{L}_k\}) = \left(\frac{\gamma^2}{\mathrm{Var}\{\nabla_\Theta \mathcal{L}_k\}} \right)^{1/2} \mathcal{N}(\mu_k, \mathrm{Var}\{\nabla_\Theta \mathcal{L}_k\}) = \mathcal{N}(\mu_k, \gamma^2). \tag{21}$$

Note that we do not discuss here the case of λ_{K+1} corresponding to the prior $P(\Theta)$, since the log-prior term acts rather as a $\mathbb{L}^2$-regularization in the equivalent constrained optimization problem, such that the weight balancing approach focuses essentially on the log-likelihood terms of the potential energy. In fact, the sampling should enable us to efficiently explore the Pareto front corresponding to balanced conditions between the data-fitting and the different PDE-based likelihood terms. On the contrary, we do not want to rely on a non-informative prior to achieve task balancing, so we impose the following upper bound

$$\mathrm{Var}\{\lambda_{K+1} \nabla_\Theta \mathcal{L}_{K+1}\} \leq \gamma^2, \tag{22}$$

which can be achieved with setting $\lambda_{K+1} \leq \sigma_{K+1}$, related to the assumption on the prior $P(\Theta) \sim \mathcal{N}(0, \sigma_{K+1}^2 I)$. This comes from the observation that

$$\lambda_{K+1} \nabla_\Theta \mathcal{L}_{K+1}(\Theta^{t_\tau}) = \frac{\lambda_{K+1}}{\sigma_{K+1}^2} \Theta^{t_\tau} \quad \text{s.t.} \quad \mathrm{Var}\{\lambda_{K+1} \nabla_\Theta \mathcal{L}_{K+1}(\Theta^{t_\tau})\} = \frac{\lambda_{K+1}^2}{\sigma_{K+1}^4} \mathrm{Var}\{\Theta^{t_\tau}\} \leqslant \frac{1}{\sigma_{K+1}^2} \mathrm{Var}\{\Theta^{t_\tau}\} \tag{23}$$

with Θ^{t_τ} the set of parameters sampled at iteration τ. The latter upper bound also provides a dispersion indicator between the posterior variance of Θ and its prior distribution, that can be used to set the value of σ_{K+1} given γ^2.

We investigate on-the-way methods to deal with the BPINNs-HMC failure mode, so that the weight adaptation strategy (20) depends on the sampling iterations τ. This results in a modified Hamiltonian Monte Carlo, denoted Adaptively Weighted Hamiltonian Monte Carlo (AW-HMC) and detailed in Algorithm 1. The weighting strategy is carried on until a number of adaptive iterations N, potentially different from the usual burn-in steps N_{burn}. It assumes that $N \leq N_{\text{burn}}$, and enables us to reach a weighted posterior distribution well-suited to the exploration of the Pareto front. Indeed, we aim to adaptively estimate a well-fitted posterior distribution that concentrates on the high probability-density region. This is achieved during the adaptive steps by leveraging gradient information of the different tasks and their respective sensitivities through variance-based weighting. Once the adaptive procedure converges, the weights stabilize, and we effectively sample from a well-fitted target distribution that mainly explores the region of highest posterior probability, characterized by the Pareto front neighborhood. Finite adaptation preserves ergodicity and asymptotic convergence of the chain while keeping $N \leq N_{\text{burn}}$ ensures the posterior distribution is drawn from the same weighted potential energy. In practice, the a-priori burn-in phase is closely linked to the number of adaptive steps by taking $N_{\text{burn}} = N$. We also introduce the notation $H_{\lambda_\tau}(\Theta, r)$ for the weighted Hamiltonian

$$H_{\lambda_\tau}(\Theta, r) = \sum_{k=0}^{K+1} \lambda_k(\tau) \mathcal{L}_k(\Theta) + \frac{1}{2} r^T \mathbf{M}^{-1} r \tag{24}$$

which defines the new transition probability for the Metropolis-Hasting acceptance criterion.

The question remains how to set the number of adaptive steps N. This is closely related to characterizing the convergence of the weighted posterior distribution toward the neighborhood where the posterior density is maximized, that is the Pareto front. To decide when to stop weights adaptation, we define a systematic stopping criterion based on the evolution of the Hamiltonian energy in equation (24). Our method for stopping weights adaptation is to reach either a maximum number of iterations $N_{\max}$ or a threshold $S_{\min}$ for the local variation of the Hamiltonian function. Indeed, if there is no significant variation, there is no more evolution of weights and task values, and therefore adaptation can be stopped. In our study, this variation is denoted $\overline{S}_\tau$ and is simply the slope of the Hamiltonian function, which is computed as the averaged linear regression of its early history, due to the strongly stochastic nature of the Hamiltonian evolution. Details on this stopping

Algorithm 1: Adaptively Weighted Hamiltonian Monte Carlo (AW-HMC).

Input: Initial Θ^{t_0}, N_s number of samples, L number of leapfrog steps, δt leapfrog step size, $N_{\max}$ maximum number of adaptive iterations, N_{burn} burn-in steps and $\mathbf{M}$ the mass matrix.

Sampling procedure:
for $\tau = 1...N_s$ **do**
  Sample $r^{t_{\tau-1}} \sim \mathcal{N}(0, \mathbf{M})$;
  Set $(\Theta_0, r_0) \leftarrow (\Theta^{t_{\tau-1}}, r^{t_{\tau-1}})$;
  Weights adaptation:
  if $(\tau \leq N_{\max})$ **and** $(\overline{S}_\tau \geq S_{\min})$ **then**
    Compute $\lambda_k(\tau) = \left(\frac{\min_{j=0..K} (\text{Var}\{\nabla_\Theta \mathcal{L}_j(\Theta_0)\})}{\text{Var}\{\nabla_\Theta \mathcal{L}_k(\Theta_0)\}} \right)^{1/2} \quad \forall k = 0..K \text{ and } \lambda_{K+1}(\tau) = 1;$
  else
    $\lambda_k(\tau) = \lambda_k(\tau - 1) \quad \forall k = 0..K \text{ and } \lambda_{K+1}(\tau) = 1$
  end
  Leapfrog:
  for $i = 0...L-1$ **do**
    $r_i \leftarrow r_i - \frac{\delta t}{2} \sum_{k=0}^{K+1} \lambda_k(\tau) \nabla_\Theta \mathcal{L}_k(\Theta_i)$;
    $\Theta_{i+1} \leftarrow \Theta_i + \delta t \mathbf{M}^{-1} r_i$;
    $r_{i+1} \leftarrow r_i - \frac{\delta t}{2} \sum_{k=0}^{K+1} \lambda_k(\tau) \nabla_\Theta \mathcal{L}_k(\Theta_{i+1})$;
  end
  Metropolis-Hastings:
  Sample $p \sim \mathcal{U}(0, 1)$;
  Compute $\alpha = \min(1, \exp(H_{\lambda_\tau}(\Theta_0, r_0) - H_{\lambda_\tau}(\Theta_L, r_L))$ using (24);
  if $p \leqslant \alpha$ **then**
    $\Theta^{t_\tau} = \Theta_L$;
  else
    $\Theta^{t_\tau} = \Theta_0$;
  end
  Collect the samples after burn-in : $\{\Theta^{t_i}\}_{i=N_{\text{burn}}}^{N_s}$
end

criterion are provided in Appendix C. The number of adaptive steps N is the minimum value between $N_{\max}$ and the number of samplings τ required to reach the threshold $S_{\min}$. In addition to this criterion, the user must check that there are effective lower bounds on the weights (e.g. Fig. 6 in Sobolev training) to verify that there is no singular behavior. Indeed, in a singular case, the objective and the Hamiltonian could decrease only by reducing of the weights to 0.

The present balancing of the target distribution, based on the minimum variance of the gradients (20) can be interpreted as adjusting the weights with respect to the most likely or the least sensitive term of the multi-potential energy. It therefore offers the advantage of improving the convergence of the BPINNs toward the Pareto-optimal solution and also enhances the reliability of the uncertainty quantification of the output, whose samples are drawn from the Pareto front. Indeed, this weighting strategy induces an automatic increase in the uncertainty of the least likely task by adaptively adjusting the λ parameters. Such observations arise from the development of upper bounds for each of the gradient variances, as detailed in Appendix A, which involves prediction errors and PDE residuals, as well as sensitivity terms characterizing the variability of the mean gradient descent directions for each task. In light of this, we were able to provide an upper bound on the joint variance γ^2 which is developed in equation (A.8) in a basic and general perspective.

Last but not least, the Inverse-Dirichlet based adaptive weighting relieves us from an unreasonable decrease in the time step, which no longer has to meet all the stiff scaling requirements to ensure Hamiltonian conservation. This approach then renders the sampling free of excessive tuning adaptation of the leapfrog hyperparameters δt and L. In addition, this prevents pathological random-walk or divergence behaviors in the sampling since it enables the use of optimal integration time, both in terms of convergence rate and adequacy of the time step to the distinct learning tasks.

The current AW-HMC algorithm is first validated on a Sobolev training benchmark with different complexities, which provides a basis for comparison with ε-optimality results. This also allows us to establish a new indicator for convergence diagnostics of the BPINNs. The robustness and efficiency of the present method are then experimented on more complex multitask and multiscale problems, along the different sections.

3.2. Sobolev training benchmark and convergence diagnostics

We investigate the performances of the proposed auto-weighted BPINN methodology on several applications, starting in this section with a Sobolev training benchmark. We first apply this new Adaptively Weighted strategy to the 1D Sobolev

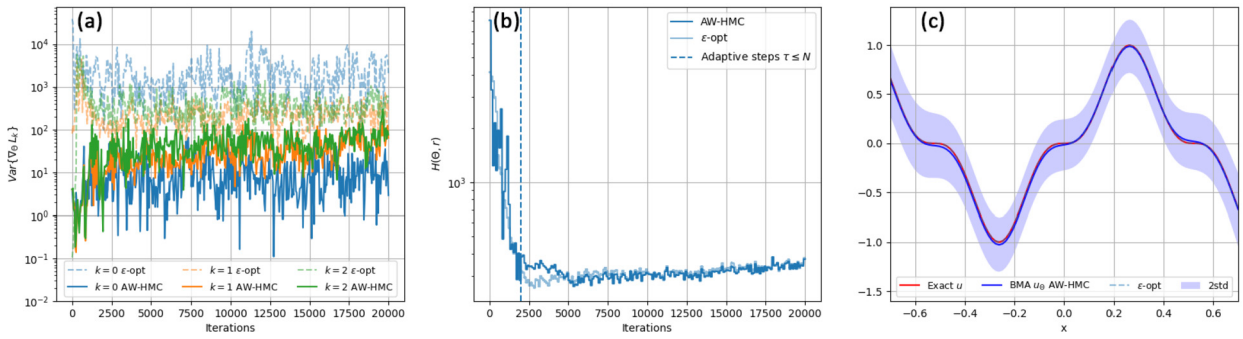

Fig. 3. Adaptively Weighted Hamiltonian Monte Carlo (AW-HMC) on Sobolev training up to second-order derivatives compared to ε-optimal weighting. (a) Effective gradient distributions variances $\mathrm{Var}\{\nabla_\Theta L_k\}$ with balanced conditions between the tasks. (b) Hamiltonian evolution throughout the sampling satisfying energy conservation. (c) Resulting field predictions with a comparison between ε-optimal results and AW-HMC strategy.

training introduced in Sect. 2.3 with the same exact set of hyperparameters. The number of adaptive steps is set to $N = 20$, as for the NUTS declination, to ensure an impartial comparison of the distinct methodologies.

In addition, we compare the predictions with a reference case where the weights λ_k are set accordingly to ε-optimal analytical solution [60], which can only be provided for linear problems. Such analytical ε-optimal weights are only available for the optimization problem of minimizing the multi-potential energy in equation (18). In the particular case of Sobolev training, these weights can be determined based on the following equation (see [34], for instance):

$$\lambda_k^{\varepsilon} = \frac{\prod_{j\neq k} \mathcal{I}_j}{\sum_{k=1}^{K} \prod_{j\neq k} \mathcal{I}_j} \quad \text{s.t. } \mathcal{I}_k = \int_\Omega |D_x^k(u)|^2 \, \mathrm{d}x. \tag{25}$$

These theoretical weights (plotted in Fig. 6 on the right) alleviate stiffness issues that lead to vanishing gradient phenomena [34]. By doing so, these weights ensure that the objectives — or tasks — are minimized in an unbiased way, hence reaching Pareto optimal solution. Although we consider a marginalization process in the Bayesian context, these analytical weights provide an interesting baseline for comparison on stiff Sobolev problems. Appropriate weighting of the posterior distribution based on the ε-optimal rule (25) ensures effective sampling that we compare with our AW-HMC strategy. In this sense, ε-optimal weighting allows us to verify that our adaptive weighing strategy reaches the Pareto front neighborhood during the adaptive steps and hence efficiently samples the region of the highest posterior probability. We tested our methodology against this ε-optimal solution assuming observation noise $\xi_u \sim \mathcal{N}(0, \sigma_u^2 I)$ such that $\sigma_k = \sigma_u, \forall k$ with $\sigma_u = 0.1$. It provides good agreement between both approaches with a similar convergence of the Hamiltonian toward the same energy level, in Fig. 3 (middle): in fact, the $\mathbb{L}^2$-relative error on the Hamiltonian values between the ε-optimal and AW-HMC methods scales around 1e−4 after the adaptive steps. The AW-HMC method also provides $\mathbb{L}^2$-relative errors, compared to this optimal solution, ranging around 1e−3 for both the signal and its derivatives. Finally, we also point out in Fig. 3 (left) balanced gradient variances in the same way as observed with ε-optimal analytical weights. The AW-HMC methodology, therefore, provides similar results to ε-optimal solutions in terms of balance between the gradient variances, exploration of the Hamiltonian energy levels, and overall BMA predictions.

Our new approach shows exceptionally balanced conditions between the different tasks: the effective gradient distribution variances $\mathrm{Var}\{\nabla_\Theta L_k\}$ present the same orders of magnitude throughout the training, even with a finite number of adaptive steps. This means that the posterior distribution reached after the auto-adjustment of the weights is well-suited to converging toward the Pareto front exploration, thus making the sampling more efficient and comparable to sampling a weighted posterior based on the ε-optimal rule (25). Preventing strong imbalance behavior on the gradient variances, and therefore task-specific bias has considerably improved the marginalization of such multi-objective potential energy, in comparison with the conventional approaches that presented major failures in Sect. 2.3.

To further demonstrate the robustness of the method, we consider the third-order derivative extension of this test case, where even a NUTS adaptive strategy on the time step (reaching $\delta t = 1.36\text{e}{-7}$) generates, here, pathological random-walk behavior making the sampling completely defective (Fig. 4 top row and Fig. 5). Such a significant decrease in the time step is clearly explained by the enhanced stiffness induced by the third-order derivative term in this multitask learning. Indeed, the Hamiltonian trajectories are more likely to diverge during the deterministic steps due to this stiffness and require a small δt to compensate for the divergence. To avoid the resulting pathological random walks, the overall integration time must be increased but this inevitably leads to excessive computational costs — under such a constraint on the leapfrog time step. This highlights the main limitation of NUTS when facing stiff multitask sampling that involves separate scales.

In contrast, our approach overcomes these major failures (see Fig. 5) without additional constraints on δt and provides balanced gradient variances between the different tasks as illustrated in Fig. 4 (bottom row). We also compare the results of

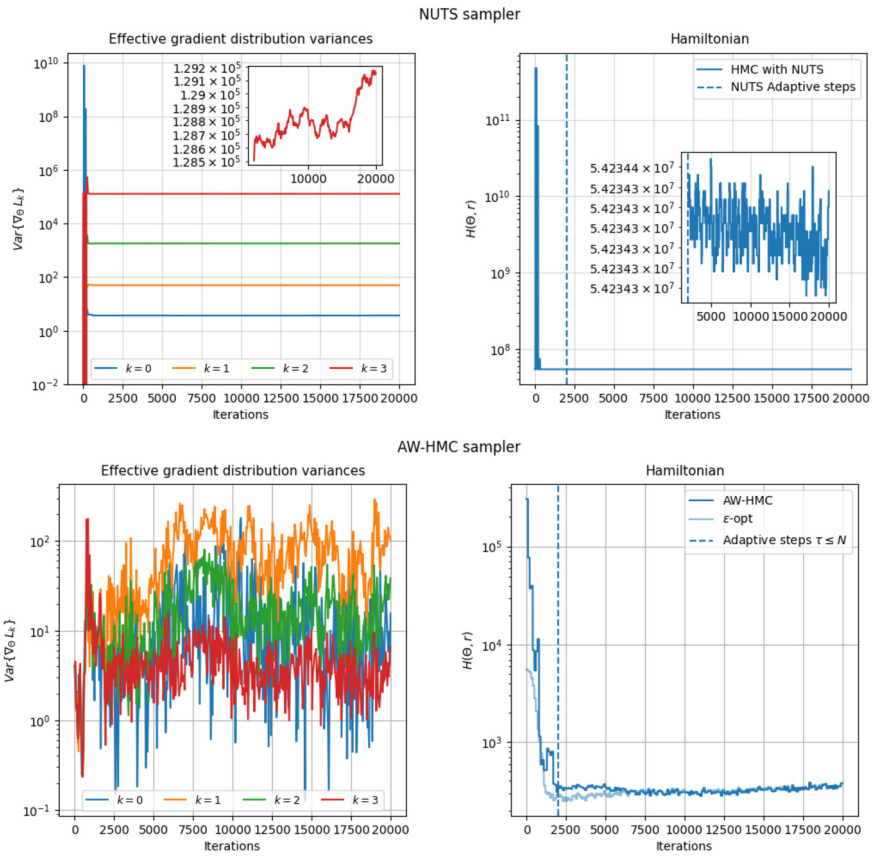

Fig. 4. Comparison between AW-HMC and NUTS samplers on Sobolev training up to third-order derivative: Effective gradient distribution variances $\mathrm{Var}\{\nabla_\Theta L_k\}$ and Hamiltonian evolution. The NUTS formulation (top row) highlights strong imbalances between the learned tasks, on the left, and random-walk behaviors in exploring the energy level sets, on the right. AW-HMC strategy (bottom row) compared with ε-optimality.

the AW-HMC methodology with analytical weights from ε-optimality and show great agreement between the approaches. In addition, in order to deal with the stochastic-induced process of the BPINNs induced by sampling variability, we perform various repetitions of the sampling with different initialization of the neural network parameters and momentum. This leads to averaged weight evolution along the adaptive steps presented in Fig. 6 that show similar order of magnitude as the analytical ε-optimal weights. The slight difference between the computed and analytical weights in Fig. 6 can be explained by the difference of nature of both problems, respectively a marginalization process and an optimization one. Indeed, the weights determined by analytical ε-optimal analysis lead to the tightest upper bound in the corresponding optimization problem — that minimizes the multi-objective potential energy (18) — but do not incorporate any uncertainty quantification, unlike the computed weights based on Inverse-Dirichlet weighting in the AW-HMC sampler. Nonetheless, the comparison is interesting and illustrates that the overall scaling of the distinct tasks is well captured during the adaptive steps of AW-HMC (see Fig. 6). The comparison of our adaptive weighting strategy with a posterior distribution weighted by the ε-optimal rule (25) shows similar convergence properties, driving the sampler close to a well-conditioned posterior distribution (see Fig. 4 bottom-right). Therefore, the weighted posterior distribution determined by AW-HMC enables exploring of the optimal Pareto front neighborhood, avoids stiffness issues, and thus provides an efficient sampling of the highest posterior probability region.

Apart from these qualitative comparisons between the different methodologies and the analytical solution, we subsequently introduce a new metric that quantifies the quality of the predictions. This complements the usual metrics with a convergence quantification of the sampling along the marginalization process. The samples collected after the burn-in steps in the AW-HMC process — i.e. all the instances of $\{\Theta^{t_i}\}_{i=N_{\text{burn}}}^{N_s}$ — are first used to determine a Bayesian Model Average estimation as defined in equation (3). Each sample provides a prediction $P(y|x,\Theta^{t_i})$, for the neural network characterized by Θ^{t_i}, and is theoretically drawn from the posterior distribution $P(\Theta|\mathcal{D},\mathcal{M})$ such that the BMA is usually approximated by [65]:

$$P(y|x,\mathcal{D},\mathcal{M}) \simeq \frac{1}{N_s - N_{\text{burn}}} \sum_{i=N_{\text{burn}}}^{N_s} P(y|x,\Theta^{t_i}) \quad \text{with} \quad \Theta^{t_i} \sim P(\Theta|\mathcal{D},\mathcal{M}). \tag{26}$$

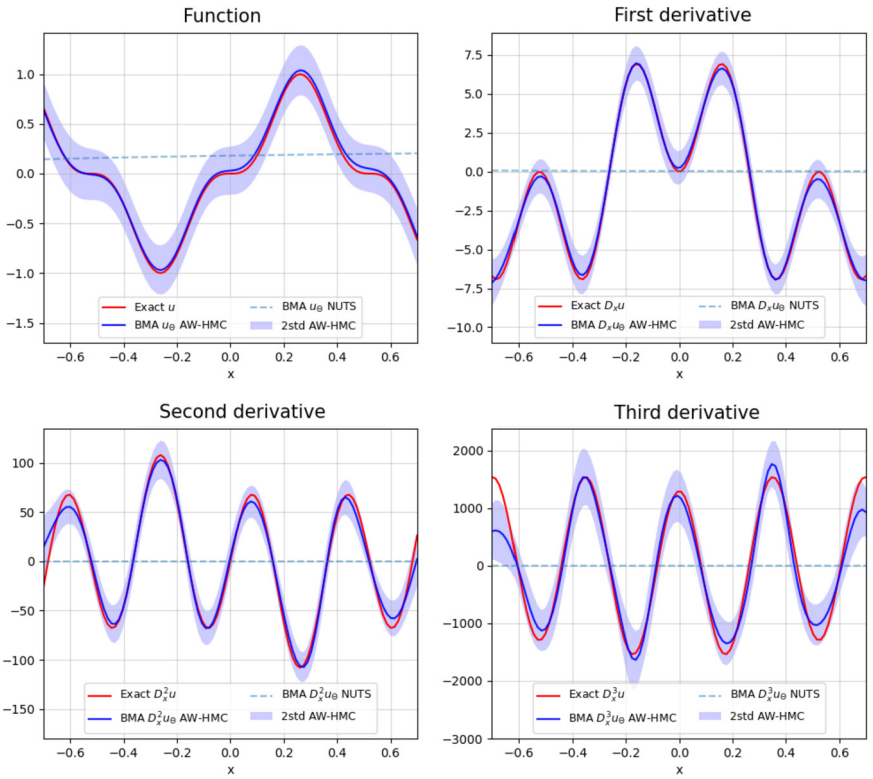

Fig. 5. Comparison of the BMA predictions (on the function and its derivatives) between the AW-HMC and NUTS formulations for 1D Sobolev training up to third-order derivative. The imbalance between tasks and random walk behavior of NUTS (see Fig. 4) results in ineffective BMA predictions. The AW-HMC methodology overcomes these effects and significantly improves the sampling of the target distribution.

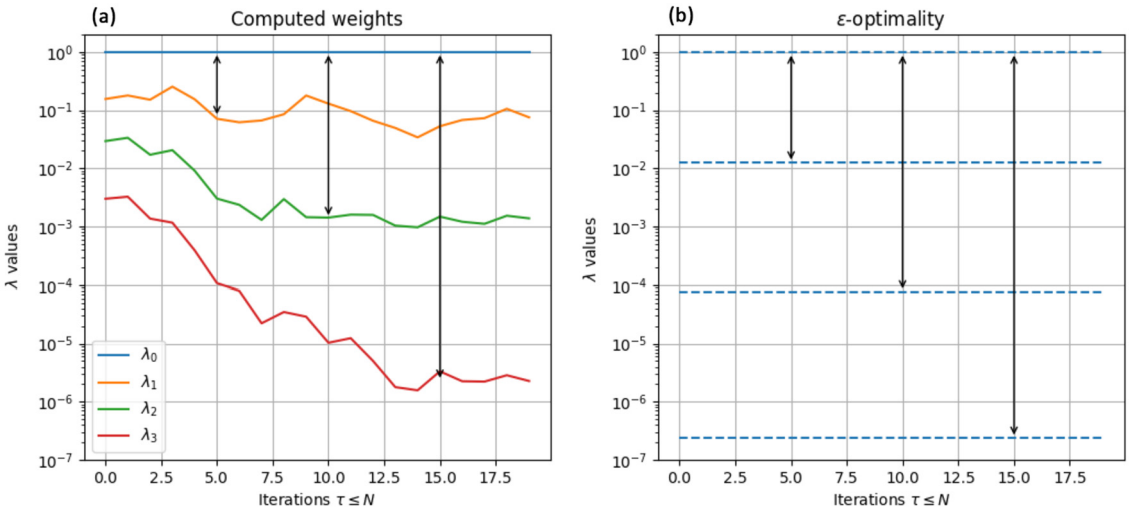

Fig. 6. λ trajectories with AW-HMC and ε-optimal weighting strategies for Sobolev training up to third-order derivative: (a) Evolution of the λ_k weights along the adaptive steps ($\tau \leq N$), averaged over several repetitions of the AW-HMC sampling. This is induced by different initializations of the neural network parameters and momentum to take into account sample variability. (b) Comparison with analytical ε-optimal weights. The order of magnitude of the relative weights λ_0/λ_i, $i = 1...3$, in both methods, are represented by the double-headed arrows.

In Sobolev training, we consider as the neural network outputs, the prediction of the function itself and all its derivatives $y = \left\{D_x^k u_\Theta,\ k = 0...K\right\}$, such that we can compute, according to equation (26), relative BMA errors with respect to each output defined by:

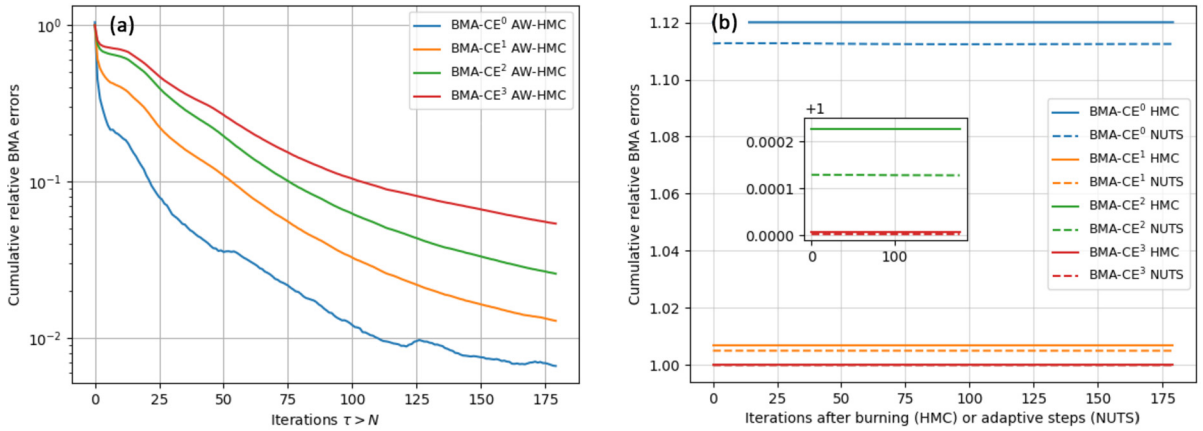

Fig. 7. Convergence diagnostics with the comparison between the HMC, NUTS, and AW-HMC samplers for Sobolev training up to third-order derivative: (a) Cumulative relative BMA errors, computed according to equation (28), plotted throughout the sampling iterations $\tau > N$ with the AW-HMC strategy. (b) Cumulative relative BMA errors for the classical HMC and NUTS formulations. These quantities remain nearly constant in pathological cases, due either to massive rejection or pathological random walk, highlighting the lack of convergence in the usual BPINNs-HMC formulations.

$$\text{BMA-E}^k = \frac{\left\| P\left(D_x^k u_\Theta \,|\, x, \mathcal{D}, \mathcal{M}\right) - D_x^k u \right\|^2}{\|D_x^k u\|^2}, \qquad \forall k = 0 \ldots K \tag{27}$$

where the notation $\|\cdot\|$ used here refers to the functional $\mathbb{L}^2$-norm. Based on the previous definition and in order to incorporate convergence on the BMA along the marginalization process, we introduce a new diagnostic called cumulative (relative) BMA error, defined as follows:

$$\text{BMA-CE}^k(\tau) = \frac{\left\| \frac{1}{\tau - N} \sum_{i=N}^{\tau} P\left(D_x^k u_\Theta \,|\, x, \Theta^{t_i}\right) - D_x^k u \right\|^2}{\left\| D_x^k u \right\|^2}, \qquad \forall k = 0 \ldots K \tag{28}$$

depending on the sampling iterations after the adaptive steps, for $\tau > N$ in Algorithm 1. These formulae can be directly extended to all the neural network outputs, in a more general framework and quantify the sampling efficiency in terms of convergence rate. The cumulative BMA errors are represented in Fig. 7 for the third-order extension of Sobolev training highlighting the convergence of the AW-HMC sampler for each of the functional tasks (on the left). Instead, these quantities remain nearly constant for the pathological HMC and NUTS formulations, due to massive rejections and random-walk behavior, respectively (see Fig. 7 on the right).

We finally extended this Sobolev training test case to several benchmarks on 2D, where we studied the impact of the functional complexity and the number of training points on the Bayesian Model Average errors. The details of these benchmark problems and the training setup are provided in Appendix B and have shown enhanced robustness and efficiency of the AW-HMC algorithm for the BPINNs.

3.3. A multiscale Lotka-Volterra inverse problem

We demonstrate the use of AW-HMC on a multiscale dynamical inverse problem to quantify the impact of the scaling. As Linka et al. [26] pointed out, sensitivity to scaling may hinder the performance of classical BPINNs, especially when considering nonlinear dynamical systems. The multiscale nature and stiffness resulting from real-world problems, where vanishing task-specific gradients are commonplace, is therefore an interesting benchmark to quantify the robustness of the present method.

In this context, we consider a Lotka-Volterra dynamical system with parameters of highly varying orders of magnitude defined by the following ordinary differential equation (ODE) system:

$$\begin{cases} \dfrac{\mathrm{d}u}{\mathrm{d}t} = \alpha u - \beta u v, \quad t \in \Omega \\ \dfrac{\mathrm{d}v}{\mathrm{d}t} = \delta u v - \gamma v, \quad t \in \Omega \\ u(0) = u_0, \; v(0) = v_0 \end{cases} \tag{29}$$

which characterizes the temporal evolution of predator-prey species. The notations $u(t)$ and $v(t)$ respectively refer to the prey and predator population size at a given time t, whereas the parameters $\alpha, \beta, \delta, \gamma \geq 0$ control the population dynamics,

as growing and shrinkage rates. Thereafter, we set the initial populations to $u_0 = 100$ and $v_0 = 20$ with the following parameters $\alpha = 1$, $\beta = 0.1$, $\delta = 0.01$ and $\gamma = 0.5$ intentionally selected with different orders of magnitude. This sets up an inverse problem benchmark based on real-world dynamics with separate scales involved.

The observation data are first numerically generated by solving the ODE system (29) on a uniform temporal grid $\Omega = [0, 50]$ with a thin resolution of 400 points. The data are randomly sampled so as to consider only half in the training phase of the different samplers. The dataset $\mathcal{D}$ then involves these partial measurements of u and v at 200 different times, potentially with some added noise, and the same collocation points are kept to satisfy the ODE constraints. In this section, we focus on an inverse problem by inferring the unknown model parameters $\Sigma = \{\alpha, \beta, \delta, \gamma\}$ from these measurement data while recovering the whole species evolution on the original finer resolution. Although the overall Bayesian inference problem is multiscale by construction, each inverse parameter is assumed to follow single-mode posterior distributions.

Regarding the noticeable scaling difference between the two populations, we consider a predator-prey split of the tasks such that each field u and v satisfies a data-fitting likelihood term and an ODE-residual likelihood term. We also assume log-normal prior distributions on Σ to ensure positivity of the inverse parameters, as Yang et al. [69] have shown that such priors improve the inference, and we set independent normal distributions on the neural network parameters θ. In practice though, we use a change of variable by introducing $\Sigma = e^{\widetilde{\Sigma}} := \left\{e^{\widetilde{\alpha}}, e^{\widetilde{\beta}}, e^{\widetilde{\delta}}, e^{\widetilde{\gamma}}\right\}$ for each of the inverse parameters to infer $\widetilde{\Sigma}$ assuming normal prior distributions as well. For this test case, we impose weakly informed priors, especially on $\widetilde{\Sigma}$, since we expect our methodology to handle the multiscale inference due to the unbiased auto-weighting of the tasks. We therefore assume that both the neural network and inverse parameters all gather the same prior distribution, given by $\Theta \sim \mathcal{N}(0, \sigma_\Theta^2 I_{p+d})$ where $\Theta = \{\theta, \widetilde{\Sigma}\}$.

Under these assumptions, we can define the multi-potential energy of the corresponding Hamiltonian system:

$$\begin{aligned} U(\Theta) = &\frac{\lambda_0}{2\sigma_0^2}\left\|u_\Theta - u\right\|_{\mathcal{D}}^2 + \frac{\lambda_1}{2\sigma_1^2}\left\|v_\Theta - v\right\|_{\mathcal{D}}^2 + \frac{\lambda_2}{2\sigma_2^2}\left\|\frac{\mathrm{d}u_\Theta}{\mathrm{d}t} - \alpha_\Theta u_\Theta + \beta_\Theta u_\Theta v_\Theta\right\|_{\mathcal{D}}^2 \\ &+ \frac{\lambda_3}{2\sigma_3^2}\left\|\frac{\mathrm{d}v_\Theta}{\mathrm{d}t} - \delta_\Theta u_\Theta v_\Theta + \gamma_\Theta v_\Theta\right\|_{\mathcal{D}}^2 + \frac{1}{2\sigma_\Theta^2}\left\|\Theta\right\|_{\mathbb{R}^{p+d}}^2 \end{aligned} \tag{30}$$

where the inferred inverse parameters are defined by $\Sigma_\Theta = e^{\widetilde{\Sigma_\Theta}}$ and we also set all the $\sigma_\bullet$ equal to one, as we do not wish to impose strong priors on the tasks and model uncertainty. As mentioned previously in Sect. 2.2, the norms are respectively the RMS and the Euclidean norm for the last term. The prior on the parameters is assumed to follow a Gaussian distribution with a larger standard deviation $\sigma_\Theta = 10$, in the sense that a slightly diffuse distribution induces weakly informed priors on the Θ parameters. This also ensures that constraint (22) for a non-informative prior is satisfied.

For such inverse modeling, the sampling is decomposed using sequential training. This means that 1) the neural network parameters are sampled with an AW-HMC strategy to mainly target the data-fitting likelihood terms (setting $\lambda_2 = \lambda_3 = 0$). 2) We then introduce the ODE-residual tasks in (30) to provide estimations of the missing inverse parameters, using the AW-HMC algorithm with initial neural network parameters θ^{t_0} resulting from 1). The BMA predictions and uncertainty quantification finally rely on this entire sampling procedure. In the two-step sequential training, the number of adaptive and sampling iterations are first given by $N = 19$ and $N_s = 100$, and then $N = 48$ and $N_s = 200$ (see Appendix C for details) while the leapfrog parameters are given by $L = 100$, $\delta t = 5\mathrm{e}{-4}$ and $2\mathrm{e}{-4}$ respectively, for the time steps in 1) and 2). The neural network itself is composed of 4 layers with 32 neurons per layer and we use the sin activation function considering the periodic nature of the solution for the Lotka-Volterra system.

On such an inverse problem the classical BPINNs-HMC algorithm faces massive rejection because the Hamiltonian trajectories are not conserved, which results in inoperative sampling (Fig. D.19). Even the adaptive strategies on the time step struggle to deal with the multiscale dynamics and require an extreme decrease in the δt value to obtain some stability, as detailed in Appendix D. The natural implication of such constraints on the leapfrog time step is lack of convergence toward the Pareto front and poor inference of the inverse parameters, subject to weakly informed priors (see Fig. D.20 and D.21 from Appendix D). In fact, Linka et al. [26] addressed the same issue on learning COVID-19 dynamics and imposed (in Sect. 4.3 of [26]) log-normal prior distributions on the inverse parameters that already rely on appropriate scaling. The need for such appropriate scaling strongly impacts the inference in the sense that it requires prior knowledge which biases the sampling.

On the contrary, we assume independent priors with respect to the scaling and show that our approach is able to properly recover all the Σ parameters as well as predict the species evolution with minimal tuning and decrease on δt. The recovery of separate scales no longer requires prior knowledge of the inverse parameter scaling to converge to their respective single modes. The results shown in Fig. 8 represent both the marginal posterior distributions of each inferred inverse parameter Σ_Θ and their trajectories when exploring the phase space distribution $\pi(\Theta, r)$. For the latter, we plotted the entire sampling trajectories that converge toward their respective mode during the adaptive steps, to finally sample around them as illustrated by the final trajectories for $\tau > N$. This confirms the ability of AW-HMC to quickly identify the separate modes of this inverse problem and manage such multiscale dynamics.

We ensure that the number of overall samples is sufficient to provide meaningful estimates of the posterior distribution (in Fig. 8) through a convergence diagnostic based on the autocorrelation of the samples. Therefore, we compute the lag-k

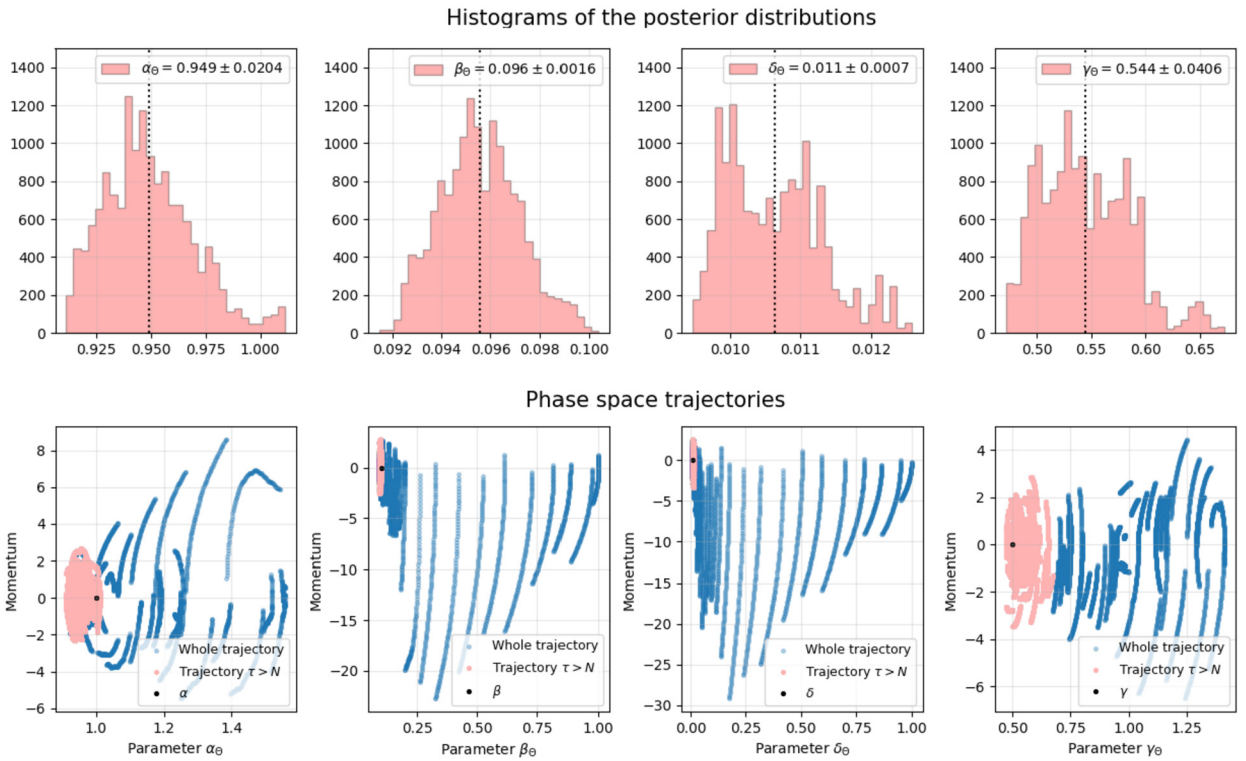

Fig. 8. Posterior distributions of the inverse parameters on the Lotka-Volterra multiscale inference. Top pictures show histograms of the marginal posterior distributions for the inverse parameters α_Θ, β_Θ, δ_Θ and γ_Θ. Bottom pictures show phase diagrams of the parameter trajectories throughout the sampling that characterized convergence toward their respective modes during the adaptive steps (in blue) and efficient exploration of the mode neighborhood after the adaptive steps (in red). The ground truth parameters are respectively $\alpha = 1$, $\beta = 0.1$, $\delta = 0.01$ and $\gamma = 0.5$ and establish an inverse problem with separate scales.

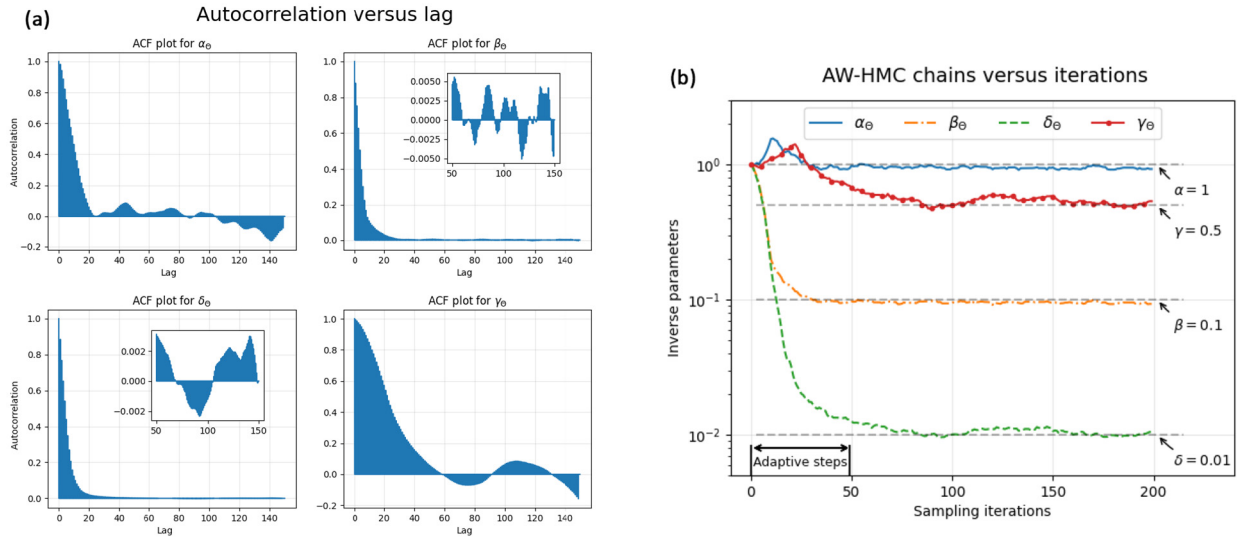

Fig. 9. Convergence diagnostics on the Lotka-Volterra multiscale inference problem: (a) Lag-k autocorrelation functions (ACF) of the inverse parameters α_Θ, β_Θ, δ_Θ and γ_Θ, plotted versus lag. (b) Inverse parameters chain traces along the sampling iterations of the AW-HMC sampler, showing convergence toward their respective single modes during the adaptive steps. The groundtruth values of each parameter are respectively represented by the gray horizontal dotted lines.

autocorrelation function (ACF), identified as a standard metric for chain convergence characterization [20,50,24] and defined as the correlation between samples k steps apart. We apply it to the AW-HMC chains for the four inverse parameters in the Lotka-Volterra inference problem. Fig. 9(a) shows the results of lag-k autocorrelation for their respective AW-HMC

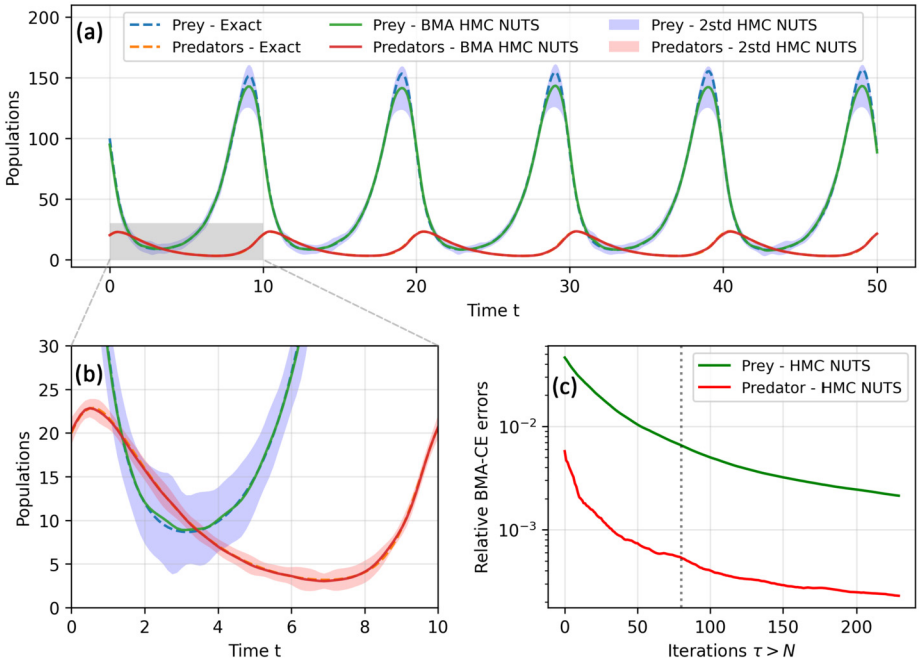

Fig. 10. Lotka-Volterra multiscale inference: (a) BMA predictions of the two-species populations along the physical time with their uncertainties, zoomed on (b). (c) Relative BMA-CE errors as defined in equation (28) for the neural network outputs $y = (u_\Theta, v_\Theta)$ plotted throughout the sampling iterations. The dotted vertical line marks the introduction of the ODE-likelihood terms in the sequential training.

chain traces from Fig. 9(b). This diagnostic rapidly drops as k increases, indicating fast mixing of the chains and a fast decorrelation of the samples from the non-informative initializations. Hence, it suggests efficient convergence of the AW-HMC chains toward the inverse parameter modes and reliable estimation of the posterior distributions.

In order to quantify the effectiveness in identifying the parameters, we also measure the relative error in the inference of the parameters $\Sigma_\Theta = \{\alpha_\Theta, \beta_\Theta, \delta_\Theta, \gamma_\Theta\}$

$$E_{\Sigma_\Theta} = \frac{|\Sigma_\Theta - \Sigma|}{\Sigma} \tag{31}$$

where the prediction is given by $\Sigma_\Theta = \dfrac{1}{N_s - N} \displaystyle\sum_{i=N}^{N_s} e^{\widetilde{\Sigma_\Theta}^{t_i}}$, and we show that these relative errors all scale around 5e−2 for the four inverse parameters. The predictive evolution of the species populations is displayed in Fig. 10, as a BMA on the neural network outputs $y = (u_\Theta, v_\Theta)$, and compared to the exact solutions in a qualitative and quantitative way. In this sense, we computed relative BMA cumulative errors for both the species, highlighting the convergence of the sampling, top right of Fig. 10. We see that the insertion of the ODE-residual likelihood terms in the two-step sequential training improves the convergence of the predictions when compared to pure data-based sampling.

This test case also reveals higher uncertainties on the evolution of the prey population characterized by effective standard deviations about four times greater (see Table 1). The enhanced uncertainty on these specific tasks is highlighted by smaller values of λ_0 and λ_2 at the end of the adaptive steps, compared to λ_1 and λ_3 in the potential energy (30). Therefore, the AW-HMC strategy benefits from its ability to adaptively weight the λ parameters to intrinsically characterize the task uncertainties based on their gradient variances.

4. Application to Computational Fluid Dynamics: stenotic blood flow

We illustrate the use of the methodology set out in Sect. 3.1 in a real-world problem from fluid mechanics, more precisely the study of inpainting and inverse problems on incompressible stenotic flows in asymmetric geometries. The objective is to demonstrate the generalization and performance of the present AW-HMC algorithm on more complex 2D geometries and nonlinear PDE dynamics under noise and sparsity of the data.

The measurement data are generated by randomly sampling the fully resolved Computational Fluid Dynamics (CFD) solutions on scattered locations. The direct numerical simulation of vascular flows in asymmetric stenotic vascular geometries is performed using a meshless solver based on the Discretization-Corrected Particle Strength Exchange (DC PSE) method as detailed in [7].

Table 1
Weight parameters λ_k obtained after the adaptive steps in the Lotka-Volterra multiscale inverse problem, for each of the sequential steps (top rows). Effective standard deviations $\widetilde{\sigma}_k$ resulting from the weight adaptations and computed as $\widetilde{\sigma}_k = \sqrt{1/\lambda_k}$ for each of the sequential steps (bottom rows). This highlights enhanced uncertainties on the tasks related to the prey species. The splitting of the sequential steps is detailed in Sect. 3.3.

Seq. step	λ_k			
	λ_0	λ_1	λ_2	λ_3
Data-fitting (step 1)	3.83e−2	1	–	–
Data-fitting + ODE tasks (step 2)	4.87e−2	1	9.16e−3	1.16e−1

Seq. step	$\widetilde{\sigma}_k$			
	$\widetilde{\sigma}_0$	$\widetilde{\sigma}_1$	$\widetilde{\sigma}_2$	$\widetilde{\sigma}_3$
Data-fitting (step 1)	5.109	1	–	–
Data-fitting + ODE tasks (step 2)	4.531	1	10.45	2.936

4.1. Inpainting problem with sparse and noisy data

Inpainting problems have drawn increasing interest in MRI or CT medical imaging as an opportunity to reduce artifacts and recover missing information by using deep learning approaches [36,3,59]. Although the usual inpainting framework incorporates only measurement data in the image processing, Zheng et al. investigated a physics-informed version of the problem by incorporating the underlying physics as indirect measurements [73]. The present section falls within the same context — the idea is to infer the whole flow reconstruction based on sparse and noisy measurements while imposing PDE constraints on some complementary collocation points.

The governing equations of the stenotic flow dynamic are written here in a velocity $\mathbf{u} = (u, v)$ and vorticity ω formulation in two dimensions, satisfying an incompressible steady-state Navier-Stokes equation given by

$$(\mathbf{u} \cdot \nabla)\omega = Re^{-1}\Delta\omega, \quad \text{in } \Omega \tag{32}$$

or equivalently

$$u\frac{\partial \omega}{\partial x} + v\frac{\partial \omega}{\partial y} = \frac{1}{Re}\Delta\omega, \quad \text{in } \Omega \tag{33}$$

where Re refers to the dimensionless Reynolds number, ω is the vorticity field $\omega = \dfrac{\partial v}{\partial x} - \dfrac{\partial u}{\partial y}$ and the incompressibility condition ensures $\nabla \cdot \mathbf{u} = 0$. We consider the 2D stenotic spatial domain $\Omega \subset [0, 10] \times [0, 1]$ and assume two different kinds of boundary conditions: 1) the stenosis upper and lower walls, denoted $\partial\Omega_1$, where we impose no-slip conditions such that $\mathbf{u}_{\partial\Omega_1} = 0$ and $\omega = (\nabla \times \mathbf{u})_{\partial\Omega_1}$ and 2) the inlet and outlet boundaries, denoted $\partial\Omega_2$, with a prescribed parabolic profile and Neumann condition, respectively, on the velocity in the main flow direction. These boundary conditions are detailed in Sect. 4.4 of the DC PSE article [7]. We also first consider that the Reynolds number is known and set to $Re = 200$ according to the CFD simulations, such that the set of parameters to infer Θ is restricted here to the neural network weights and bias.

The measurement dataset, $\mathcal{D}$, is composed of noisy vorticity data on $\mathcal{D}^{\partial_1}$ and $\mathcal{D}^{\partial_2}$, defined as in (12) respectively for sets $\partial\Omega_1$ and $\partial\Omega_2$, as well as on 1282 interior collocation points $\mathcal{D}^{\omega}$ that cover less than 2% of all the data required for the full vorticity field reconstruction on Ω. We finally defined the $\mathcal{D}^{\Omega}$ dataset as 6408 interior points representing 6% of the entire reconstructed data field, where we require that the PDE (33) be satisfied in a physically-constrained inpainting formulation. The multi-potential energy is then defined by:

$$\begin{aligned} U(\Theta) = &\frac{\lambda_0}{2\sigma_0^2}\left\|\omega_\Theta - \omega\right\|^2_{\mathcal{D}^{\omega}} + \frac{\lambda_1}{2\sigma_1^2}\left\|\omega_\Theta - \omega_{|\partial\Omega_1}\right\|^2_{\mathcal{D}^{\partial_1}} + \frac{\lambda_2}{2\sigma_2^2}\left\|\omega_\Theta - \omega_{|\partial\Omega_2}\right\|^2_{\mathcal{D}^{\partial_2}} \\ &+ \frac{\lambda_3}{2\sigma_3^2}\left\|u_n\frac{\partial\omega_\Theta}{\partial x} + v_n\frac{\partial\omega_\Theta}{\partial y} - \frac{1}{Re}\Delta\omega_\Theta\right\|^2_{\mathcal{D}^{\Omega}} + \frac{1}{2\sigma_\Theta^2}\|\Theta\|^2_{R_p} \end{aligned} \tag{34}$$

with u_n and v_n noisy evaluations of the velocity field on the $\mathcal{D}^{\Omega}$ set. We then used sequential training by adding the PDE-residual likelihood term in the second sampling phase, such that the AW-HMC parameters are given first by $N = 50$ and $N_s = 200$, and then $N = 50$ and $N_s = 250$ for a leapfrog path length $L = 150$ and time step $\delta t = 5\text{e}{-4}$. As for the previous benchmarks, we set all the $\sigma_\bullet$ equal to one and assume a centered normal distribution with the standard deviation $\sigma_\Theta = 10$ for the neural network parameters prior. The neural network is composed of 4 layers with 32 neurons per layer and is based on the hyperbolic tangent activation function. The velocity and vorticity CFD solutions $(\mathbf{u}, \omega)$ are both corrupted by additive Gaussian noise such that $\bullet_n = \bullet + \sigma\xi$, where $\xi \sim \mathcal{N}(0, \psi^2)$ is a vector of element-wise independent and identically-distributed Gaussian random numbers with mean zero and variance $\psi^2 = \text{Var}\{\bullet\}$, and σ refers to the level of added noise.

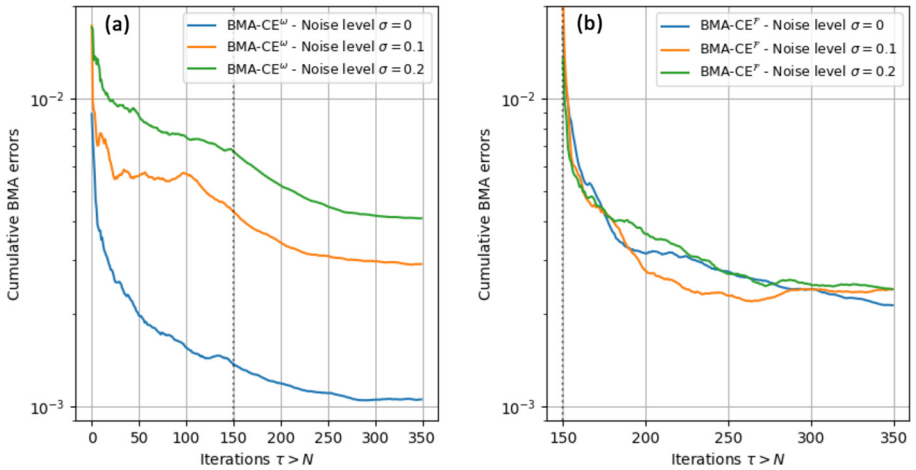

Fig. 11. Bayesian Model Average Cumulative Error diagnostics (BMA-CE), as defined in (35), for the vorticity physics-informed inpainting problem: (a) BMA-CE on the vorticity field prediction throughout the sampling iterations and for different noise levels. (b) BMA-CE on the PDE residual $\mathcal{F}$ satisfying (33) for different noise levels. The dotted vertical lines mark the introduction of the PDE constraint in the two-step sequential training.

In this physics-informed inpainting problem, we investigate the impact of the level of noise σ on the BMA predictions of the vorticity field, as well as on the physical constraint by extending the notion of BMA convergence to the PDE residual. Hence, we compute the BMA-CE diagnostics for the field ω and the PDE constraint based on

$$\begin{aligned} \text{BMA-CE}^{\omega}(\tau) &= \left\| \frac{1}{\tau - N} \sum_{i=N}^{\tau} P\left(\omega_{\Theta} \,|\, x, \Theta^{t_i}\right) - \omega \right\|^2 \\ \text{BMA-CE}^{\mathcal{F}}(\tau) &= \left\| \frac{1}{\tau - N} \sum_{i=N}^{\tau} P\left(\mathcal{F}(\omega_{\Theta}) \,|\, x, \Theta^{t_i}\right) \right\|^2 \end{aligned} \tag{35}$$

with $\mathcal{F}(\omega_{\Theta})$ the evaluation of the PDE from equation (33). The comparative curves for different noise levels are represented in Fig. 11 and show sampling convergence toward final BMA errors that scale about 1.05e−3, 2.9e−3 and 4.08e−3, respectively, for noise levels $\sigma = 0,\ 0.1$ and 0.2. In addition, we see that the PDE residual constraints converge independently to the noise level, reaching final BMA errors around 2e−3 in all cases.

To supplement the performance quantification of the inpainting formulation in recovering the entire vorticity field along with its uncertainty, we also use the Prediction Interval Coverage Probability (PICP) metric as defined by Yao et al. [71]. This consists of a quality indicator of the posterior approximation, which evaluates the percentage of the ground truth observations contained within 95% of the prediction interval, as given by:

$$PICP = \frac{1}{N_{\text{obs}}} \sum_{i=1}^{N_{\text{obs}}} \mathbb{1}_{(\omega^l_{\Theta})_i \le \omega_i \le (\omega^h_{\Theta})_i} \tag{36}$$

where ω^l_{Θ} and ω^h_{Θ} are respectively the 2.5% and 97.5% percentiles of the predictive distribution on the vorticity. In this case, the notation N_{obs} refers to the total number of observations in the predictive dataset, in other words, the grid resolution of the computational domain Ω. In our application, this PICP metric shows that more than 99% of the vorticity ground truth observations are covered by the posterior distribution of the neural network output ω_{Θ}, independently of the level of noise.

We also expect our self-weighted adaptation of λ_k to be able to capture noise sensitivity with respect to the value of σ, and intrinsic task sensitivities to noise level without imposing any a-priori on the noise level estimation. This is the key point of our methodology since we intentionally decouple σ_k in (34) from the noise magnitude, and rely on the self-weighted strategy to quantify their related uncertainties. On the contrary, when dealing with noisy measurement data in applications researchers frequently assume the fidelity of each sensor to be known and set the standard deviations σ_k accordingly. They can also be defined as additional learnable parameters to be inferred. The latter is usually subject to additional computational costs in *online* learning or requires alternative neural network formalism used as pre-training in *offline* learning [44]. In contrast, the strength of the AW-HMC methodology relies on its similar computational cost compared to classical BPINNs-HMC. Moreover, AW-HMC improves convergence by drawing attention to exploring the Pareto front with optimal integration time. Therefore, it can shorten overall sampling requirements making this a competitive strategy in terms of computational cost.

The results presented in Fig. 12 demonstrate the noise resistance of the AW-HMC approach and highlight sensitivity consideration with respect to the noise and tasks (see Table 2). We first noticed differences in the auto-adjustment of the

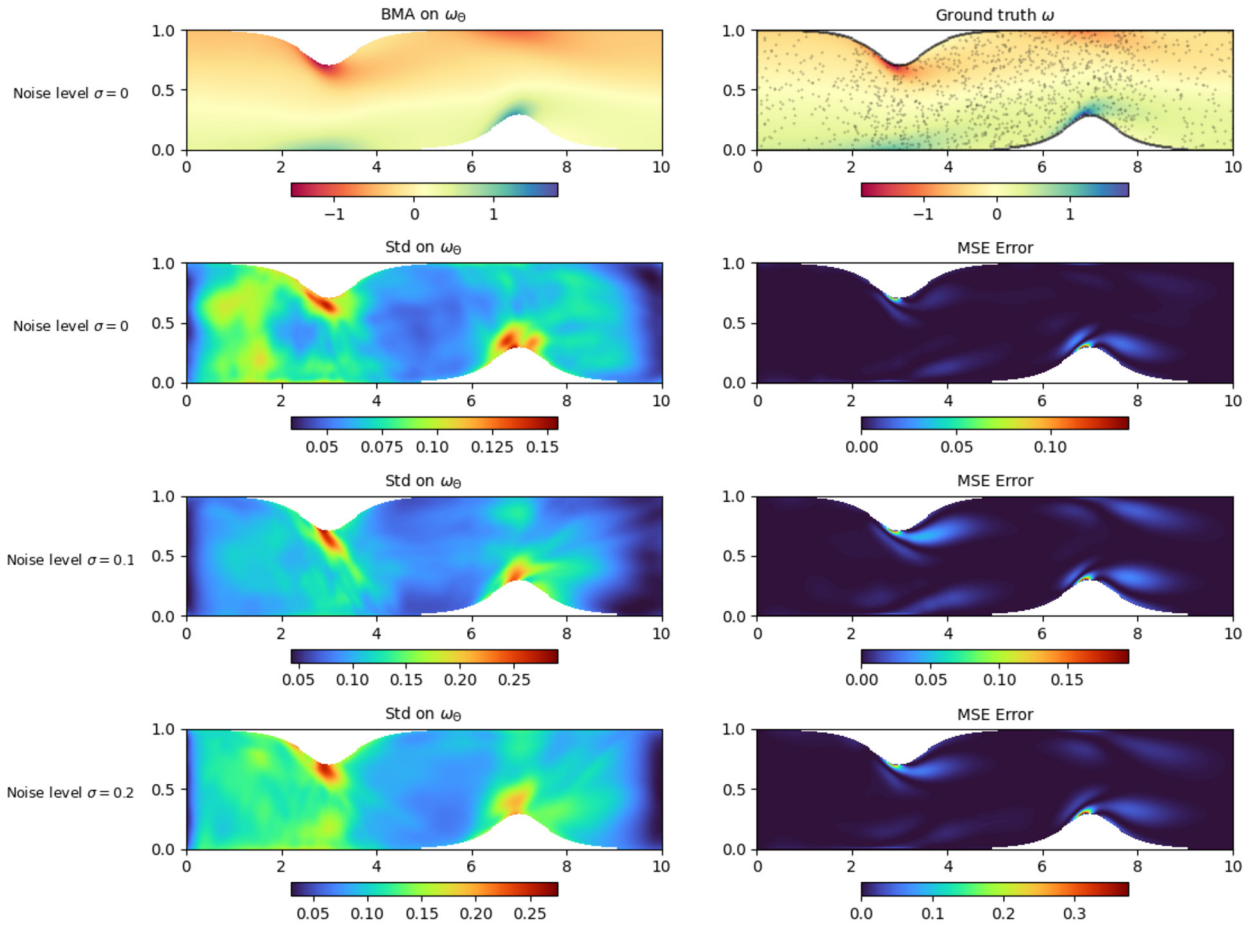

Fig. 12. Uncertainty quantification on the physics-informed inpainting problem: BMA prediction of the vorticity field ω_Θ in asymmetric stenosis without noise, compared to the ground truth solution ω (top row). The black dots on the exact field correspond to the training measurements of the dataset $\mathcal{D}$. Comparison of the uncertainty standard deviations (Std) and mean squared errors (MSE) on the predicted vorticity field ω_Θ for different noise levels ($\sigma = 0, 0.1, 0.2$), shown in the bottom rows.

Table 2
Final λ_k weight parameters for each σ noise level in the physics-informed inpainting problem (top rows). Effective $\widetilde{\sigma}_k$ standard deviations resulting from the weight adaptations and computed as $\widetilde{\sigma}_k = \sqrt{1/\lambda_k}$ for each noise level (bottom rows). This highlights the overall adaptation of the effective standard deviations to the noise magnitude and the task sensitivities to the noise level. In particular, the wall-boundary conditions associated with λ_1 present the highest noise sensitivity.

Noise level	λ_k			
	λ_0	λ_1	λ_2	λ_3
$\sigma = 0$	1	0.46	0.88	0.51
$\sigma = 0.1$	1	0.19	0.35	0.29
$\sigma = 0.2$	1	0.12	0.39	0.27

Noise level	$\widetilde{\sigma}_k$			
	$\widetilde{\sigma}_0$	$\widetilde{\sigma}_1$	$\widetilde{\sigma}_2$	$\widetilde{\sigma}_3$
$\sigma = 0$	1	1.47	1.07	1.40
$\sigma = 0.1$	1	2.29	1.69	1.86
$\sigma = 0.2$	1	2.89	1.60	1.92

lambda values relative to noise levels, leading to global enhanced uncertainties with increasing noise. We also observed various uncertainty adjustments depending on the sensitivity of the different tasks to the noise. In fact, the comparison of the local standard deviations on the vorticity field in Fig. 12 shows that the wall boundary conditions are the most sensitive to noise, automatically increasing the uncertainties in these areas. The inlet and outlet boundaries are rather less sensitive. This is highlighted by a lower adaptation of their uncertainties to the noise level. In short, this application has shown the ability of our new adaptive methodology to automatically adjust the weights, and with them the uncertainties, to the intrinsic task sensitivities to the noise and to adapt the uncertainty to noise magnitude itself.

4.2. Inverse problem with parameter estimation and latent field recovery

As a second CFD application, we consider a multi-objective flow inverse problem in an asymmetric and steep stenosis geometry. This aims to provide both a parameter estimation of the flow regime and recover a hidden field using our adaptively weighted strategy. Such considerations, motivated by real-world applications, use incomplete or corrupted measurement data in an attempt to derive additional information, which remains challenging or impractical to obtain straightforwardly.

With an emphasis on physical and biomedical problems, Raissi et al. investigated the extraction of hidden fluid mechanics quantities of interest from flow visualizations, using physics-informed deep learning [47,48]. The authors relied only on measurements of a passive scalar concentration that satisfied the incompressible Navier-Stokes equations, to infer the velocity and pressure fields in both external and internal flows.

In this direction, we focus on the velocity $\mathbf{u} = (u, v)$ and pressure p formulation of the stenotic flow dynamics such that the continuity and momentum governing steady-state equations are written:

$$\begin{cases} (\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + Re^{-1}\Delta\mathbf{u}, & \text{in } \Omega \\ \nabla\cdot\mathbf{u} = 0, & \text{in } \Omega \end{cases} \tag{37}$$

under the incompressibility condition on the stenotic domain $\Omega \subset [0, 10] \times [0, 1]$. We impose adherent boundary conditions on the wall interfaces such that $\mathbf{u}_{\partial\Omega_1} = 0$, and the following inlet/outlet boundary conditions respectively:

$$\begin{aligned} u = 4y - 4y^2,\ v = 0 \quad \forall (x, y) \in \{0\} \times [0, 1] \\ \frac{\partial u}{\partial x} = 0,\ v = 0 \quad \forall (x, y) \in \{10\} \times [0, 1]. \end{aligned} \tag{38}$$

The direct numerical simulation is performed using the DC-PSE formulation [7] with a Reynolds number set to $Re = 200$, as in the previous section. It is used to generate the observation data on Ω with a thin resolution. The $\mathcal{D}$ dataset is then composed of partial measurements of $\mathbf{u}$ randomly sampled to consider 9559 training points, representing less than 3% of the entire target resolution. The same collocation points are included to impose the PDE constraints, denoted $\mathcal{F}(\mathbf{u}) := (\mathcal{F}(u), \mathcal{F}(v))$, as well as the diverge-free condition $\mathcal{H}(\mathbf{u})$.

Finally, we set up the inverse problem by inferring the flow regime, considering the Reynolds number as an unknown model parameter $\Sigma = \{Re\}$. At the same time, we address the multitask problem to recover the latent pressure from the partial measurements of the velocity field and the fluid flow dynamics assumptions. The pressure field prediction, in particular, is adjusted throughout the sampling in such a way that its gradient satisfies the governing equations (37). As commonly established by the nature of the Navier-Stokes equation, the pressure is though not uniquely defined and, given the lack of precise boundary conditions on this field, is thus determined up to a constant. The predictions of each of the quantities of interest, namely the velocity and pressure, are then recovered on the original finer resolution in Fig. 13. As in Sect. 3.3, we select a log-normal prior distribution for the physical parameter and independent normal distributions for the neural network parameters, and we also use a sequential training approach, incorporating the PDE constraints in the second sampling phase.

The validation of the inference is first performed by computing the BMA-CE diagnostics for the velocity field components, the PDE constraints, and the incompressibility condition written in the same way as in equation (35). The results are provided in Fig. 14 and highlight the convergence of each term toward final BMA errors scaling respectively about $\text{BMA-CE}^{u}(N_s) = 6.4\text{e}{-3}$, $\text{BMA-CE}^{v}(N_s) = 1\text{e}{-3}$, $\text{BMA-CE}^{\mathcal{F}(\mathbf{u})}(N_s) = (4.2\text{e}{-2}, 4.7\text{e}{-2})$ and $\text{BMA-CE}^{\mathcal{H}(\mathbf{u})}(N_s) = 2.9\text{e}{-2}$. The Bayesian Model Average predictions of the velocity field are then compared in Fig. 13 with the ground truth observations providing local mean squared error (MSE) that are embedded in their uncertainties and show enhanced standard deviations at the regions with higher errors. The PICP metric also enables to estimate that more than 95% of the velocity field ground truth is recovered by the posterior distribution of $\mathbf{u}_\Theta$.

For the inverse parameter, we computed a BMA cumulative error based on the relative L1-norm defined as follows

$$\text{BMA-CE}^{Re}(\tau) = \frac{\left| \frac{1}{\tau}\sum_{i=1}^{\tau} Re_{\Theta^{t_i}} - Re \right|}{|Re|}, \qquad \forall \tau = 1 \ldots N_s \tag{39}$$

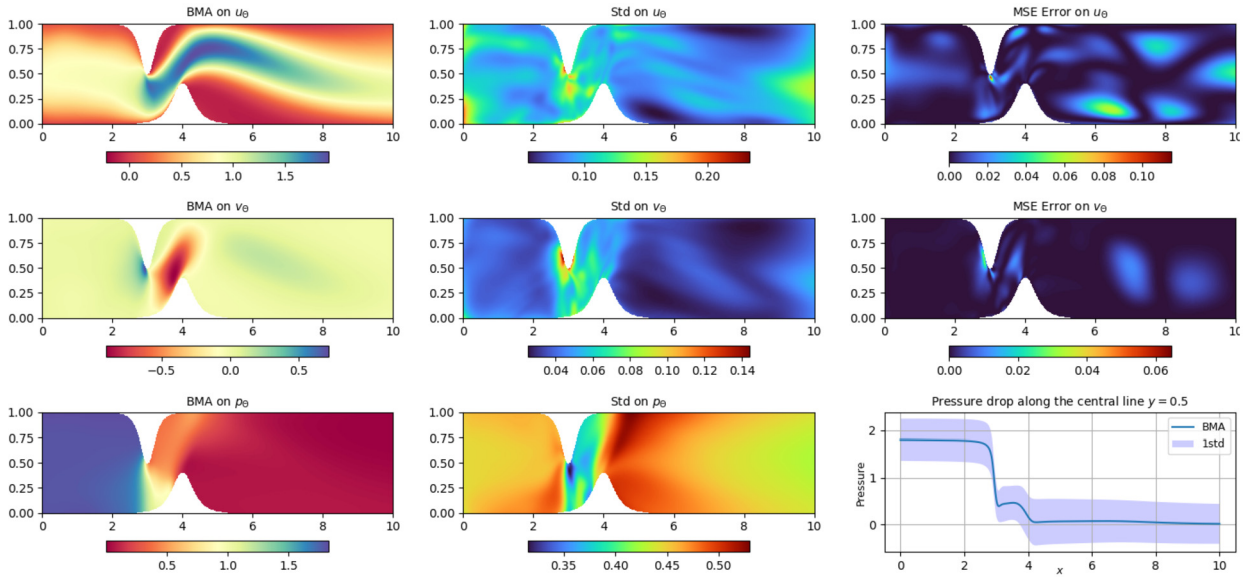

Fig. 13. Uncertainty quantification on the CFD inverse problem: BMA predictions of the velocity field $\mathbf{u}_\Theta = (u_\Theta, v_\Theta)$ in asymmetric stenosis along with their uncertainty standard deviations (Std) and mean squared errors (MSE), at the top. BMA and uncertainty on the inferred latent pressure field with the pressure evolution plotted along the central line $y = 0.5$, leading to an average pressure drop of 1.78 — bottom.

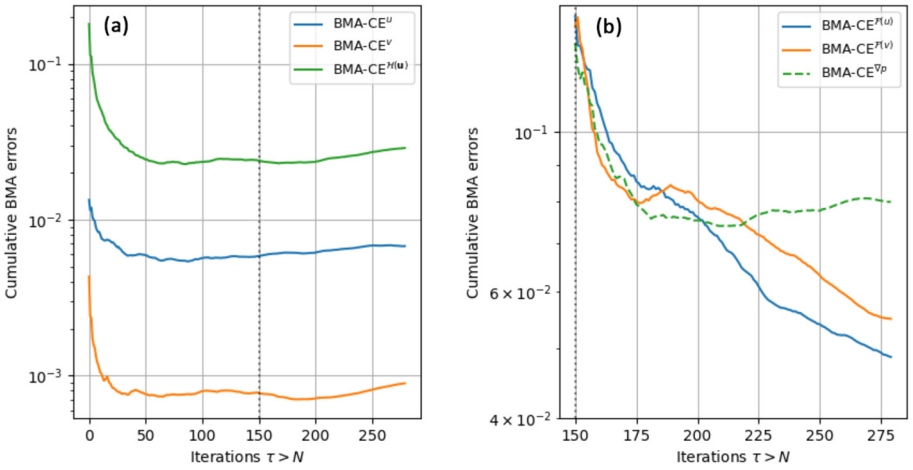

Fig. 14. Bayesian Model Average Cumulative Errors (BMA-CE) diagnostics for the CFD inverse problem: (a) BMA-CE throughout the sampling iterations for the velocity field components $\mathbf{u} = (u, v)$ and the divergence-free condition $\mathcal{H}(\mathbf{u})$. (b) BMA-CE on the PDE residuals $\mathcal{F}(u)$ and $\mathcal{F}(v)$. The dotted curve represents the a-posteriori checking of pressure gradient norm BMA-CE error as defined in equation (40). The dotted vertical lines, in both figures, delimit the two stages of the sequential training.

where $Re_{\Theta^{t_i}}$ refers to the prediction of the Reynolds number for the sample characterized by the parameters Θ^{t_i}. We show in Fig. 15 that this relative error converges, reaching at the end of the sampling a residual of 5.4e−2. We also represent here the histogram of the marginal posterior distribution of Re_Θ and its trajectory in the phase space illustrating the convergence toward its mode during the adaptive steps $\tau < N$. In fact, our approach leads to an estimate of the Reynolds number, inferred from the measurements data $\mathcal{D}$, which is consistent with the exact value and results in the predictive interval $Re_\Theta \in [182.82, 208.06]$.

The latent pressure field BMA, inferred up to constant, is illustrated in Fig. 13 with its uncertainty and is able to capture a sharp pressure drop — estimated in average to 1.78 — arising from the steep stenosis geometry. In fact, it has been emphasized by Sun et al. in symmetric geometries, that such pressure drops turn to become nonlinear as the stenotic geometry becomes narrower [56], which is in line with what we obtain in our asymmetric case. As the pressure ground truth is unknown in this application, we complement the validation of the inverse problem with a-posteriori checking on the pressure gradient. In this sense, we provide a PICP estimate on the pressure recovery which stands around 91% for its gradient norm, but also introduce the following posterior diagnostic on the pressure BMA-CE error:

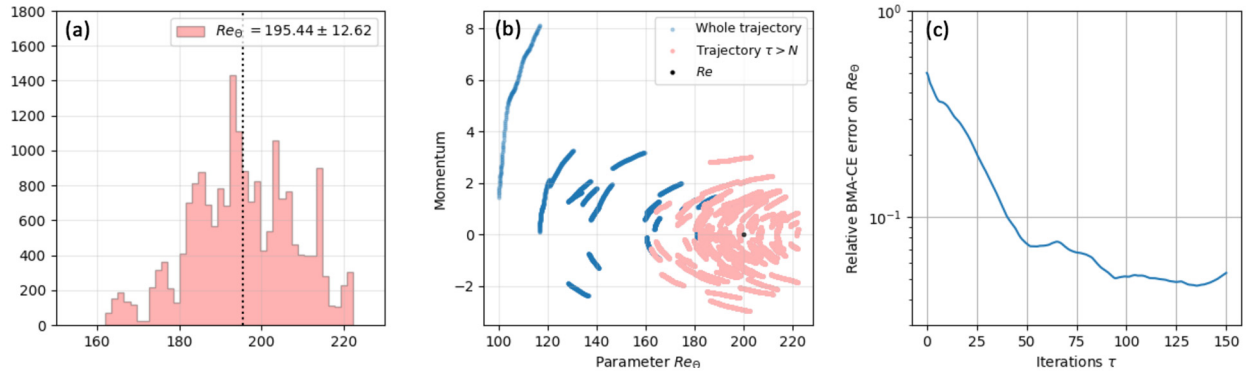

Fig. 15. Reynolds number posterior distribution and convergence diagnostic in CFD inverse problem: (a) Histogram of the marginal posterior distribution for the inverse Reynolds parameter. (b) Phase diagram of its trajectory throughout the sampling, with the adaptive steps trajectories (in blue) and effective sampling (in red). (c) BMA-CE error using the absolute relative norm as defined in (39). The relative BMA-CE error on Re_Θ is plotted over all the τ iterations of the second step sampling in the sequential training.

$$\text{BMA-CE}^{\nabla p}(\tau) = \left\| \frac{1}{\tau - N} \sum_{i=N}^{\tau} \left| P\left(\nabla p_\Theta \,|\, x, \Theta^{t_i}\right) \right| - |\nabla p| \right\|^2 \tag{40}$$

where $|\cdot|$ denotes the vector norm, and ∇p is the evaluation of the exact gradient pressure from equation (37). The results are plotted, in dotted line, throughout the sampling iterations in Fig. 14, and reach a residual error of 7.6e−2. This illustrates good agreement between the ground truth and the predictive pressure gradient arising from our adaptively-weighted strategy.

Overall, the present AW-HMC methodology relies on multitask sampling to identify the flow regime through partial measurements of the velocity field and thus handles a complex flow inverse problem with latent field recovery that satisfies nonlinear physical PDE constraints.

5. Concluding remarks

BPINNs have recently emerged as a promising deep-learning framework for data assimilation and a valuable tool for uncertainty quantification (UQ) [69]. This offers the opportunity to merge the predictive power of Physics-Informed Neural Networks (PINN) with UQ in a Bayesian inference framework using Markov Chain Monte Carlo (MCMC) sampling. This makes it possible to quantify the confidence in predictions under sparse and noisy data with physical model constraints, which is especially appealing for applications in complex systems. For this, Hamiltonian Monte Carlo has been established as a powerful MCMC sampler due to its ability to efficiently explore high-dimensional target distributions [4]. With it, BPINNs have extended the use of PINNs to a Bayesian UQ setting.

As we have shown here, BPINNs, however, share similar failure modes as PINNs: the multi-objective cost function translates to a multi-potential sampling problem in a BPINN. This presents the same difficulties in balancing the inference tasks and efficiently exploring the Pareto front as found in standard PINNs [49]. We illustrated this in a Sobolev training benchmark, which is prone to stiffness, disparate scales, and vanishing task-specific gradients. We emphasized that BPINNs are sensitive to the choice of the λ weights in the potential energy, which can possibly lead to biased predictions or inoperative sampling. Hence, the standard weighting strategy appears to be inefficient in multiscale problems and multitask inference, while it turns out to be unsustainable to manually tune the weights in a reproducible and reliable way. Recently proposed alternatives [44] are subject to additional hyper-parameter tuning or pre-training of the weights with a GAN, at the expense of increased computational complexity. Also, previous approaches mainly focused on measurement noise estimation and did not include physical model mis-specification concerns which are also critical, especially when UQ modeling is the goal.

Robust automatic weighting strategies are therefore essential to apply BPINNs to multiscale and multitask (inverse) problems and improve the reliability of the UQ estimates. Here, we have therefore proposed the AW-HMC BPINN formulation, which provides a plug-in automatic adaptive weighting strategy for standard BPINNs. AW-HMC effectively deals with multi-potential sampling, energy conservation instabilities, disparate scales, and noise in the data, as we have shown in the presented benchmarks.

We have shown that the presented strategy ensures a weighted posterior distribution well-fitted to explore the Pareto front, providing balanced sampling by ensuring appropriate adjustment of the λ weights based on Inverse Dirichlet weighting [34]. The weights can therefore directly be interpreted as training uncertainties, as measured by the variances of the task-specific training gradients. This leads to weights that are adjusted with respect to the model to yield the least sensitive multi-potential energy for BPINN HMC sampling. This results in improved convergence, robustness, and UQ reliability, as the sampling focuses on the Pareto front. This enables BPINNs to effectively and efficiently address multitask UQ.

The proposed method is also computationally more efficient than previous approaches, since it does not require additional hyper-parameters or network layers. This also ensures optimal integration time and convergence in the leapfrog training. This prevents time steps from tending to zero or becoming very small, avoiding a problem commonly encountered in No-U-Turn Sampling (NUTS) when attempting to avoid the pathologically divergent trajectories characteristic of HMC instabilities. The present methodology improves the situation, since the time step no longer needs to meet all of the stiff scaling requirements to ensure energy conservation. As a result, it shortens overall integration time and sample number requirements, combining computational efficiency with robustness against sampling instabilities.

Our results also show that AW-HMC reduces bias in the sampling, since it is able to automatically adjust the λ parameters, and with them the uncertainty estimates, according to the sensitivity of each term to the noise or inherent scaling. In classical approaches, this is prohibited by the bias and implicit prior introduced by manual weight tuning. In fact, we demonstrated the efficiency of the present method in capturing inverse parameters of different orders of magnitude in a multiscale problem, assuming completely independent priors with respect to the scaling. Previously, this would have been addressed by imposing prior distributions on these parameters that already rely on appropriate scaling. Otherwise, the classic BPINN formulation is prone to failure. The proposed adaptive weighting strategy avoids these issues altogether, performing much better in multiscale inverse problems.

We have demonstrated this in real-world applications from computational fluid mechanics (CFD) of incompressible flow in asymmetric 2D geometries. We showed the use of AW-HMC BPINNs for CFD inpainting and studies the impact of noise on the multi-potential energy. This highlighted the robustness of the present approach to noisy measurements, but also its ability to automatically adjust the λ values to accurately estimate the noise levels themselves. In this sense, we were able to show enhanced uncertainty with increasing noise, without any prior on the noise level itself, and to capture distinct intrinsic task sensitivities to the noise. Overall, this offers an effective alternative to automatically address multi-fidelity problems with measurements resulting from unknown heteroscedastic noise distributions.

Taken together, the present results render BPINNs a promising approach to scientific data assimilation. They now have the potential to effectively address multiscale and multitask inference problems, to couple UQ with physical priors, and to handle problems with sparse and noisy data. In all of these, the presented approach ensures efficient Pareto front exploration, the ability to correctly scale multiscale and stiff dynamics, and to derive unbiased uncertainty information from the data. Our approach involves only minimal assumptions on the noise distribution, the different problem scales, and the weights, and it is computationally efficient. This extends the application of BPINNs to more complex real-world problems that were previously not straightforwardly to address.

Applications we expect to particularly benefit from these improvements include porous media research, systems biology, and the geosciences, where BPINNs now offer promising prospects for data-driven modeling. They could support and advance efforts for the extraction and prediction of morphological geometries [42,54], upscaling and coarse-graining of material properties [1] and physical properties [51] directly from sample images. However, capturing these features from imperfect images remains challenging and is usually subject to uncertainties, e.g., due to unavoidable imaging artifacts. This either requires the development of homogenization-based approaches [21] to bridge scales and quantify these uncertainties [41] or the use of data assimilation to compensate for the partial lack of knowledge in the images. The present BPINNs formulation with AW-HMC offers a potential solution.

CRediT authorship contribution statement

The conceptualization has been collectively established. The computational code and GPU/HPC implementation has been made by SP. The article was written by SP and revised by SM, PP and IS. The positioning and results were discussed collectively. The mathematical analysis (probability and statistical analysis) was performed by SP. Funding was provided by PP and IS.

Acknowledgements

This work was partially supported by ANR Grant ANR-CE45-0022, E2S-UPPA, Carnot Institute ISIFoR P450902ISI, and Bundesministerium für Bildung und Forschung (BMBF) project ScaDS.AI.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Appendix A. Upper bound on the Inverse-Dirichlet weighting variance

The Inverse-Dirichlet Adaptively Weighted HMC algorithm, developed in Sect. 3.1, guarantees that the gradients of the multi-potential energy terms have balanced distributions throughout the sampling, as shown by their joint variance below:

$$\gamma^2 := \mathrm{Var}\{\lambda_k \nabla_\Theta \mathcal{L}_k\} \simeq \min_{t=0,\dots,K}(\mathrm{Var}\{\nabla_\Theta \mathcal{L}_t\}), \quad \forall k = 0, \dots, K. \tag{A.1}$$

In this section, we use a general case to demonstrate that γ^2 is upper bound and controlled by a reliability criterion which depends on the prediction errors or PDE residuals, the dispersion of their mean variability with respect to Θ and the setting of the $\sigma_\bullet$ values.

This first states the necessity to adequately set the σ parameters to avoid biased and imbalanced conditions on task gradient distributions, since these parameters critically and arbitrarily affect the gradient distributions control. This also highlights that manual tuning of the σ values may be an extremely sensitive task, difficult to achieve in practice. Therefore, in all the applications presented in this article, we chose to set these parameters uniformly and instead rely on the λ automatic adjustment to ensure, *inter alia*, the efficient exploration of the Pareto front. It ensures that these standard deviation parameters imply a strong constraint on each gradient distribution — with respect to Θ — and so, impact each task uncertainty.

For the sake of simplicity, we used two-task sampling with a data-fitting term from a field u and a PDE constraint, denoted $\mathcal{F}$, so the data set is decomposed into $\mathcal{D} = \mathcal{D}^u \cup \mathcal{D}^\Omega$, following the notations introduced in Sect. 2.2. The multi-potential energy thus reduces to:

$$U(\Theta) = \frac{\lambda_0}{2\sigma_0^2} \|u_\Theta - u\|^2_{\mathcal{D}^u} + \frac{\lambda_1}{2\sigma_1^2} \|\mathcal{F}(u_\Theta)\|^2_{\mathcal{D}^\Omega} + \frac{1}{2\sigma_\Theta^2} \|\Theta\|^2 := \sum_{k=0}^{K+1} \lambda_k \mathcal{L}_k(\Theta) \tag{A.2}$$

where we choose to keep the σ notation for the demonstration and restrain Θ to the neural network parameters, even if the following holds in an inverse problem paradigm. As a reminder, the measurement data used for the training $\mathcal{D}^u$ can differ from the collocation points where we impose the PDE constraint $\mathcal{D}^\Omega$ and their respective numbers are denoted N^u and N^Ω. With the notations from Sect. 2.2, the gradients of the two-tasks potential energy write respectively:

$$\begin{aligned}
\frac{\partial \mathcal{L}_0}{\partial \Theta_j}(\Theta) &= \frac{1}{\sigma_0^2 N^u} \sum_{i=0}^{N^u} \Big(u_\Theta(x_i) - u_i \Big) \frac{\partial u_\Theta}{\partial \Theta_j}(x_i) \\
\frac{\partial \mathcal{L}_1}{\partial \Theta_j}(\Theta) &= \frac{1}{\sigma_1^2 N^\Omega} \sum_{i=0}^{N^\Omega} \mathcal{F}\Big(u_\Theta(x_i) \Big) \frac{\partial \mathcal{F}(u_\Theta)}{\partial \Theta_j}(x_i)
\end{aligned} \tag{A.3}$$

for $\Theta \in \mathbb{R}^p$ and we can thus decompose the variances $\mathrm{Var}_\Theta\big[\nabla_\Theta \mathcal{L}_k\big]$, $k = 0, 1$ with respect to these gradients. To do so, we first compute their mean with respect to Θ and get respectively

$$\mathbb{E}_\Theta\big[\nabla_\Theta \mathcal{L}_0\big] = \frac{1}{N^p} \sum_{j=0}^{N^p} \frac{\partial \mathcal{L}_0}{\partial \Theta_j}(\Theta) = \frac{1}{\sigma_0^2 N^u} \sum_{i=0}^{N^u} \Big(u_\Theta(x_i) - u_i \Big) \mathbb{E}_\Theta\big[\nabla_\Theta u_\Theta\big](x_i) = \frac{1}{\sigma_0^2} \mathbb{E}_{\mathcal{D}^u}\big[(u_\Theta - u)\mathbb{E}_\Theta\big[\nabla_\Theta u_\Theta\big]\big] \tag{A.4}$$

and

$$\mathbb{E}_\Theta\big[\nabla_\Theta \mathcal{L}_1\big] = \frac{1}{\sigma_1^2} \mathbb{E}_{\mathcal{D}^\Omega}\big[\mathcal{F}(u_\Theta)\mathbb{E}_\Theta\big[\nabla_\Theta \mathcal{F}(u_\Theta)\big]\big] \tag{A.5}$$

with the special configuration $\nabla_\Theta \mathcal{F}(u_\Theta) = \mathcal{F}(\nabla_\Theta u_\Theta)$ if $\mathcal{F}$ is linear. Finally, we can extend it to the variance computations, as follows:

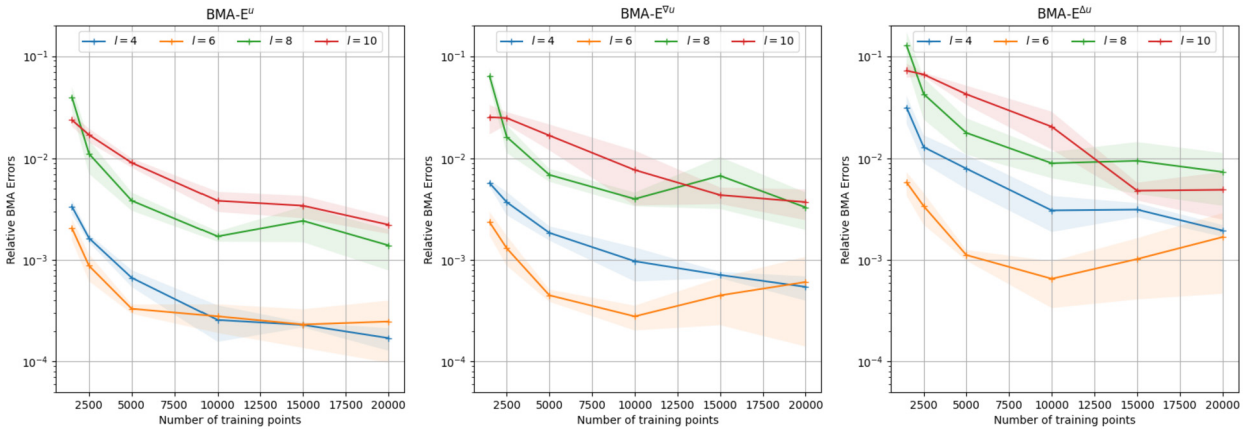

Fig. B.16. 2D Sobolev training benchmark: comparison of the relative BMA errors, as defined in (27), plotted with respect to the number of training points for each term of the multi-potential energy. We consider various shape complexities induced by the different values of l. The number of training points is increased until about 30% of the whole data set is reached, for 20000 training points.

$$
\begin{aligned}
\mathrm{Var}_\Theta\big[\nabla_\Theta \mathcal{L}_0\big] &= \frac{1}{N^p}\sum_{j=0}^{N^p}\left(\frac{\partial \mathcal{L}_0}{\partial \Theta_j} - \mathbb{E}_\Theta\big[\nabla_\Theta \mathcal{L}_0\big]\right)^2 \\
&= \frac{1}{N^p (N^u \sigma_0^2)^2}\sum_{j=0}^{N^p}\left[\sum_{i=0}^{N^u}\Big(u_\Theta(x_i) - u_i\Big)\left(\frac{\partial u_\Theta}{\partial \Theta_j}(x_i) - \mathbb{E}_\Theta\big[\nabla_\Theta u_\Theta\big](x_i)\right)\right]^2 \\
&= \frac{1}{(N^u \sigma_0^2)^2}\sum_{i=0}^{N^u}\sum_{k=0}^{N^u}\Big(u_\Theta(x_i) - u_i\Big)\Big(u_\Theta(x_k) - u_k\Big)\mathrm{Cov}_\Theta\big[\nabla_\Theta u_\Theta(x_i), \nabla_\Theta u_\Theta(x_k)\big] \\
&\leqslant \frac{1}{\sigma_0^4}\|u_\Theta - u\|^2_{\infty,\mathcal{D}^u}\mathrm{Cov}_\Theta\left[\frac{1}{N^u}\sum_{i=0}^{N^u}\nabla_\Theta u_\Theta(x_i), \frac{1}{N^u}\sum_{k=0}^{N^u}\nabla_\Theta u_\Theta(x_k)\right] \\
&= \frac{1}{\sigma_0^4}\|u_\Theta - u\|^2_{\infty,\mathcal{D}^u}\mathrm{Var}_\Theta\big[\mathbb{E}_{\mathcal{D}^u}\big[\nabla_\Theta u_\Theta\big]\big]
\end{aligned}
\tag{A.6}
$$

that provides an upper bound for the gradient variance of the data-fitting term. We then obtain, in the same way, the PDE constraint bound as:

$$
\mathrm{Var}_\Theta\big[\nabla_\Theta \mathcal{L}_1\big] \leqslant \frac{1}{\sigma_1^4}\|\mathcal{F}(u_\Theta)\|^2_{\infty,\mathcal{D}^\Omega}\mathrm{Var}_\Theta\big[\mathbb{E}_{\mathcal{D}^\Omega}\big[\nabla_\Theta \mathcal{F}(u_\Theta)\big]\big]. \tag{A.7}
$$

The notation $\|\cdot\|_{\infty,\mathcal{D}^\bullet}$ here refers to the discrete ℓ^∞ norm on the spatial domain composed of the $\mathcal{D}^\bullet$ training points, and $\mathbb{E}_{\mathcal{D}^\bullet}$ introduces the spatial mean on the corresponding data set. Hence, the gradient variances of the tasks are controlled by the crossed complex components $\mathrm{Var}_\Theta\mathbb{E}_{\mathcal{D}^\bullet}$ which can be interpreted as sensitivity terms evaluating the dispersion with respect to Θ of the gradient descent directions, averaged in space. Finally, since the $\sigma_\bullet$ values are uniformly set to one to avoid biased sampling, it means that the λ values are computed in such a way the joint variance of the gradient distributions is bounded by:

$$
\gamma^2 \leqslant \min\left\{\|u_\Theta - u\|^2_{\infty,\mathcal{D}^u}\mathrm{Var}_\Theta\big[\mathbb{E}_{\mathcal{D}^u}\big[\nabla_\Theta u_\Theta\big]\big],\ \|\mathcal{F}(u_\Theta)\|^2_{\infty,\mathcal{D}^\Omega}\mathrm{Var}_\Theta\big[\mathbb{E}_{\mathcal{D}^\Omega}\big[\nabla_\Theta \mathcal{F}(u_\Theta)\big]\big]\right\} \tag{A.8}
$$

which highlights the fact that the weights are adjusted with respect to the most likely task and thus improve the reliability in the uncertainty quantification. The present computations can straightforwardly be extended to more complex multi-potential energy terms for direct and inverse real-world problems, which concludes our analysis.

Appendix B. 2D Sobolev training benchmark

We extend the Sobolev benchmark used in Sect. 3.2 to 2D-training with the gradient and laplacian operators, with a target functional in the form:

$$
u(x, y) = \sum_{i=1}^{N_{rep}} A_x^i \cos\left(2\pi L^{-1} l_x^i x + \phi_x^i\right) A_y^i \sin\left(2\pi L^{-1} l_y^i y + \phi_y^i\right) \tag{B.1}
$$

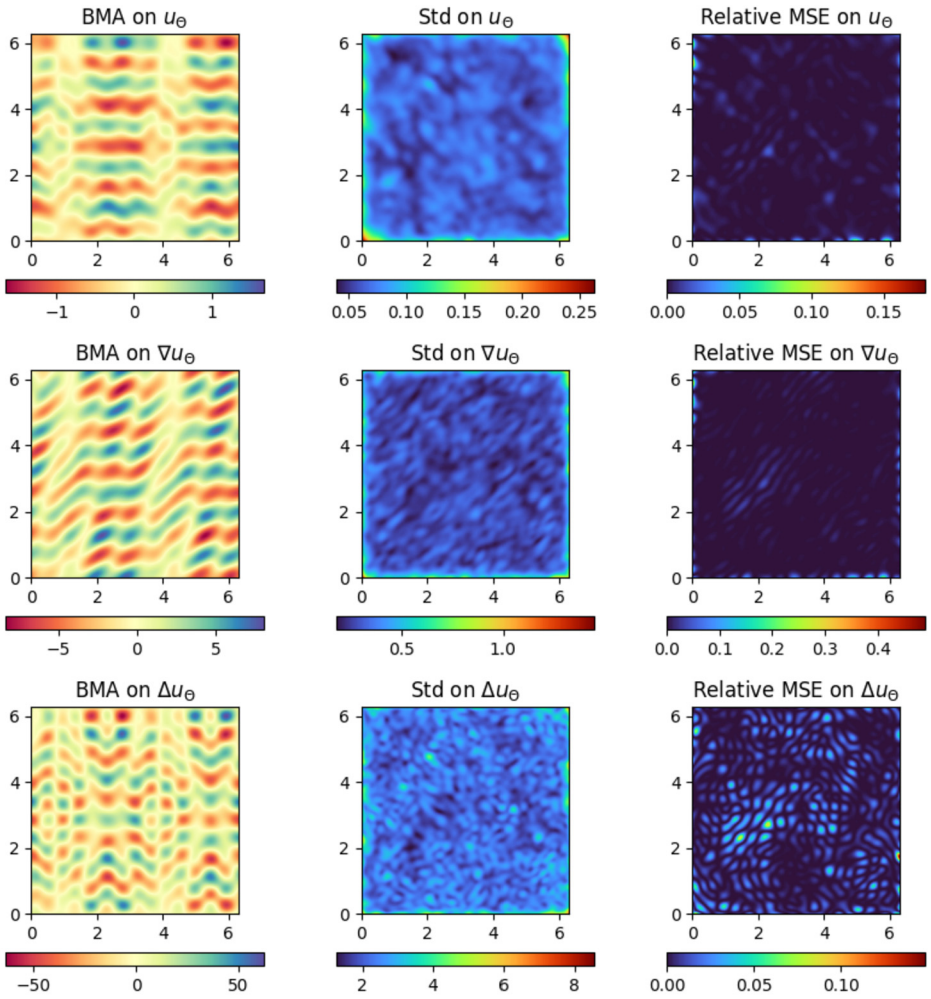

Fig. B.17. 2D Sobolev training benchmark: BMA predictions, predicted standard deviation, and relative BMA errors, presented locally for each term of the multi-objective potential energy. We worked on a limited case $l = 8$ with about 15% of training points. The global relative BMA errors, averaged over the entire domain, scale around $1.69e-3$, $3.42e-3$, and $5.6e-3$ respectively.

which enables us to deal with a wide range of shape complexities and sharp interfaces, in addition to the stiffness introduced by the higher-order derivatives. We set the domain size to $L = 2\pi$, the number of repetitions $N_{rep} = 5$, while the parameters A_x and A_y are independently and uniformly sampled from the interval $[-2, 2]$, as are ϕ_x and ϕ_y from $[0, 2\pi]$. In order to treat several shape complexities, we consider a range of parameter l such that the local length scales l_x and l_y are randomly sampled from the set $\{1, 2, ..., l\}$. The 2D spatial domain $[0, 2\pi]^2$ is covered by a uniform grid with a resolution of 256×256, along with randomly-selected training points. We then study both the impact of the functional complexity, by setting different values of l, and the number of training points on the Bayesian Model Average resulting from our AW-HMC methodology.

The results on the entire benchmark setup, presented in Fig. B.16, show a convergence trend with an increasing number of training points, and this independently of the l values, even though the relative BMA errors reach higher bounds with additional shape complexity. These relative BMA errors are computed according to equation (27) and are average versions of different repetitions of Sobolev sampling, simultaneously running in parallel. In fact, in order to deal with the stochastic-induced process that may arise from the sampling variabilities themselves, we performed several realizations starting with distinct initializations of the neural network Θ^{t_0} and momentum r^{t_0} parameters, which lead to different sampling realizations. We can potentially take into account these sampling variabilities to compute the standard deviation over these repetitions, as illustrated by the colored band in Fig. B.16.

We also represent in Fig. B.17 for each term of the multi-potential functional, respectively, their BMA predictions, their uncertainties based on the predictive standard deviations throughout the sampling, and the relative BMA errors in the case $l = 8$ and with 10000 training points, randomly sampled over the whole domain. The results here show enhanced uncertainties near the boundary walls where the higher errors are located and highlight the ability of our methodology to

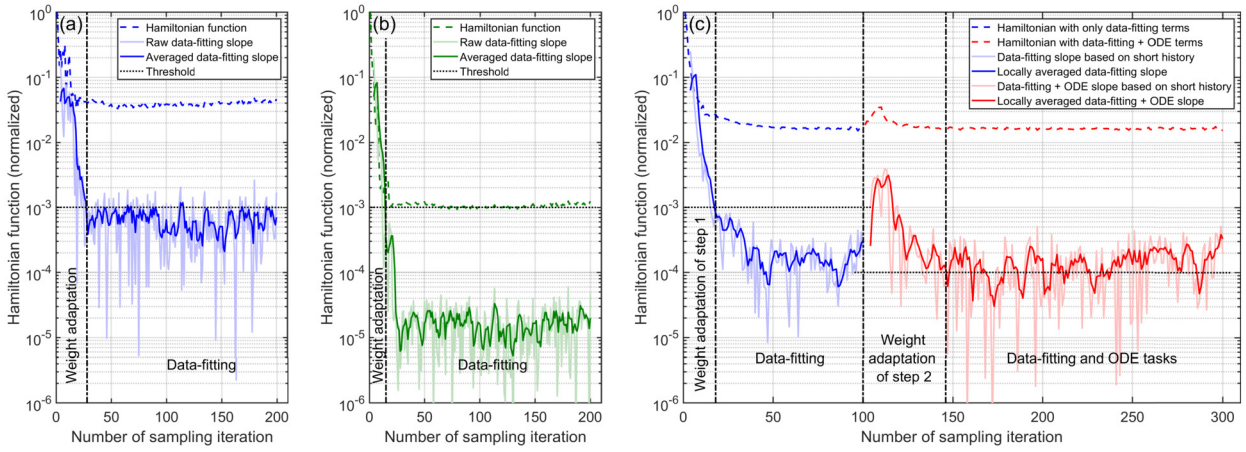

Fig. C.18. Hamiltonian averaged slope used as stopping criterion for weight adaptation. The vertical dashed lines depict the start and the end of the weight adaptation process. Each graph describes a different case: (a) Sobolev training with derivatives up to second order, (b) Sobolev training with derivatives up to third order, and (c) Lotka-Volterra inverse problem. The last example (c) has two sequential steps, first data-fitting alone in blue (with a preliminary weight adaptation involving only the fitting of the two-component data sets with $\lambda_2 = \lambda_3 = 0$ in equation (30)) and second data-fitting together with satisfying the ODE in red (with an updated weight adaptation since we consider the full potential energy in equation (30)).

capture complex shape fields of different orders of magnitude at the same time. We can also emphasize that such a 2D Sobolev training benchmark was previously unachievable with the classical BPINNs-HMC formulation.

Appendix C. Stopping criterion for weight adaptation

We detail the criterion used for stopping the weight adaptation procedure and activating the sampling from the weighted target distribution, with balanced tasks, which mainly explores the region of the highest posterior probability. This stopping criterion is based on a maximum number of iterations together with a convergence threshold.

Indeed, the stopping criterion for the weights adaptation, which we make coincide with the burn-in procedure, is reaching a maximum number of iterations $N_{\max}$ or ensuring the local variation $\overline{S}_\tau$ of the Hamiltonian function to be below a given threshold $S_{\min}$ (if there is no significant variation, there is no more evolution of weights and task values, so adaptation can be stopped).

The significant variation in the Hamiltonian evolution, given its strongly stochastic nature, is based on the locally averaged best slopes of the Hamiltonian function, whose statistical series is denoted H_i related to the number of iterations $X_i = i$. The best slopes S_τ are first computed by linear regression on the short history corresponding to the last p_{slope} values of the Hamiltonian function (light colors on Fig. C.18). Then, these slopes are themselves denoised by local averaging on the last p_{mean} values (solid colors on Fig. C.18) resulting in the following final expression of $\overline{S}_\tau$:

$$\overline{S}_\tau = \frac{1}{p_{\text{mean}}} \sum_{i=\tau-p_{\text{mean}}+1}^{\tau} S_i \quad \text{where} \quad S_\tau = \frac{\text{Cov}\left(\{H_i\}_{i=\tau-p_{\text{slope}}+1}^{\tau}, \{X_i\}_{i=\tau-p_{\text{slope}}+1}^{\tau}\right)}{\text{Var}\left(\{X_i\}_{i=\tau-p_{\text{slope}}+1}^{\tau}\right)}.$$

In practice, we choose $p_{\text{slope}} = p_{\text{mean}} = 4$. Moreover, it must be checked a-posteriori that there are lower bounds on the weights in order to verify that there is no singular behavior: if so, the objective and the Hamiltonian could decrease solely by decreasing of the weights toward 0. This check is especially important since we allow auto-adaptation of weights, which trivially doesn't occur when weights are given constant values.

For the Lotka-Volterra case in Sect. 3.3, the maximum number of adaptations is set to $N_{\max} = 50$. Two sequential steps are considered in this study:

- A data-fitting step, whose weight adaptation threshold is set to $S_{\min} = 1\text{e}{-3}$. The Hamiltonian function contains only the gap between the prediction and the observation (two components of the Lotka-Volterra dynamical system values/observations), plus the momentum. This corresponds to the potential energy given in equation (30) with $\lambda_2 = \lambda_3 = 0$, that is to say:

$$U(\Theta) = \frac{\lambda_0}{2\sigma_0^2} \|u_\Theta - u\|_{\mathcal{D}}^2 + \frac{\lambda_1}{2\sigma_1^2} \|v_\Theta - v\|_{\mathcal{D}}^2 + \frac{1}{2\sigma_\Theta^2} \|\Theta\|_{\mathbb{R}^{p+d}}^2.$$

- A second step involving both data-fitting and the constraint to satisfy the ODE, whose threshold is set lower at $S_{\min} = 1\text{e}{-3}$ due to good preconditioning from the first step. A new adaptation is still required, though, since the Hamiltonian

function contains two more terms, the two residual values of the ODE expression (plus the momentum). The resulting potential energy is then given in equation (30).

The stooping criteria described above lead to an exit after $N = 19$ adaptation iterations for step 1, and after $N = 48$ adaptive iterations for step 2, corresponding to the third dashed line in Fig. C.18(c).

For the Sobolev training in Sect. 3.2 and Appendix B, the threshold is also set to 1e−3 (there is only data-fitting for this benchmark problem) and the maximum number of adaptive iterations is set to $N_{\max} = 20$. To ensure a fair comparison between the different samplers, the number of iterations used for weight adaptation in Sect. 3.2 is manually set to $N = 20$. Nevertheless, as shown in Fig. C.18, the Sobolev training with derivatives up to the second order would exit after 28 iterations, and its formulation with derivatives up to the third order would exit after 15 iterations. It is noticeable that the 28 iterations of the adaptation process is a bit overestimated, due to the lag induced by the $p_{\text{slope}} + p_{\text{mean}} = 8$ backward averaging, so stopping after 20 iteration is totally acceptable. Fig. 6 confirms that the task weights indeed have lower bounds for Sobolev training up to third-derivative, so all tasks are meaningfully balanced. The weights stabilization observed for iterations $\tau > 15$ (in Fig. 6) also confirms that the present stopping criterion is relevant.

Appendix D. Failure of the usual methodologies on the Lotka-Volterra inverse problem

We consider the Lotka-Volterra inverse problem, as introduced in Sect. 3.3, to investigate the impact of multiscale dynamics on the usual methodologies, namely HMC with uniform weighting and NUTS. The sampling and leapfrog parameters are set accordingly to the AW-HMC test case, where N refers here to the burn-in and number of adaptive steps for the HMC and NUTS formulations, respectively. Therefore, we compare the different samplers assuming that 1) their time complexity is the same and 2) we are imposing no informative priors on the inverse parameter scaling. In fact, the first condition states that different leapfrog parameters might improve the inference of these conventional methodologies. However, it implies a noticeable decrease in the leapfrog time step δt, thus slower exploration of the energy levels. Hence, these methods require either an increase in the integration time — by increasing L — or using a large number of samples, to obtain suitable predictions. As a reminder, independently of this lack of efficiency in the posterior distribution sampling, poor choices on the weights of multi-potential energy can bias the sampler and deviate it from the Pareto front exploration. The second assumption is motivated by the willingness to address UQ on multiscale dynamics without any prior knowledge of the separate scales. This arises from an assertion by Linka et al. [26] indicating that sensitivity to scaling disrupts the performance of the BPINNs-HMC. Finally, we also consider sequential training to provide an appropriate basis for comparison between the different methods.

The results show a lack of convergence of the classical HMC with uniform weighting (in Fig. D.19 — top right) and also, a strong imbalance between the tasks. The relative BMA-CE errors effectively characterize an extremely poor convergence of the predator population with respect to the prey population, which translates directly into an inefficient BMA prediction for the two-species populations (in Fig. D.19 — bottom and top left). This failure mode is due essentially to the massive rejection of the samples (acceptance rate less than 1%) due to non-conservation of the Hamiltonian trajectories along the leapfrog steps. Hence, this confirms the lack of robustness of the BPINNs-HMC paradigm when facing instability issues due to multiscale dynamics.

The NUTS alternative also struggles to converge on this multiscale inverse problem and results in inadequate predictions, especially for the predator population. Here the reason is not the massive sample rejection but rather a prohibitive decrease in the time step, reaching $\delta t = 8.26\text{e}{-5}$ and $2.81\text{e}{-5}$, respectively, at the end of the adaptive steps — nearly corresponding to a ten-fold drop in the time step, compared to AW-HMC. The relative BMA-CE errors (in Fig. D.20 — top right) reveal that this time step adaptation is suitable for the convergence of the prey population since it appears to be the most sensitive task. This sensitivity should be understood in the sense that small variations with respect to Θ on the potential energy induce the strongest constraint on the Hamiltonian energy conservation. However, the time-step adaptation is not satisfactory for the predator population and even leads to inefficient forgetting of the neural network throughout the sequential training. This translates into misleading predictions on the evolution of the population (see Fig. D.20 — bottom and top left) and unsuccessful inference of the inverse parameters (Fig. D.21). The phase diagram of the inverse parameter trajectories demonstrates the difficulties of the NUTS sampler in adequately identifying the modes resulting from separate scales. Overall, the NUTS sampler suffers from a lack of convergence toward the Pareto front and a misleading inference of the inverse parameters, subject to weakly-informed priors, due to its inability to capture multiscale behaviors.

Appendix E. Characterization of the multi-potential energy in the CFD inverse problem

The CFD inverse problem, defined in Sect. 4.2, involves the recovery of the latent pressure field p_Θ in addition to the flow regime parameter — given by the Reynolds number Re_Θ — based upon partial measurements of the velocity field. The training dataset $\mathcal{D}$ used for the AW-HMC sampling is first decomposed into 9559 measurements of randomly-sampled $\mathbf{u}$, which respectively defined the $\mathcal{D}^{\mathbf{u}}$ and $\mathcal{D}^{\partial}$ sets of interior and boundary points. The same collocation points define $\mathcal{D}^{\Omega}$, where we impose the PDE constraints and the diverge-free condition. The steep stenosis geometry considered in this problem generates sharp gradients at the wall interface. The latter need to be adequately captured to obtain consistency in the inference of the latent pressure and inverse parameter. Hence, we complemented the training with some partial

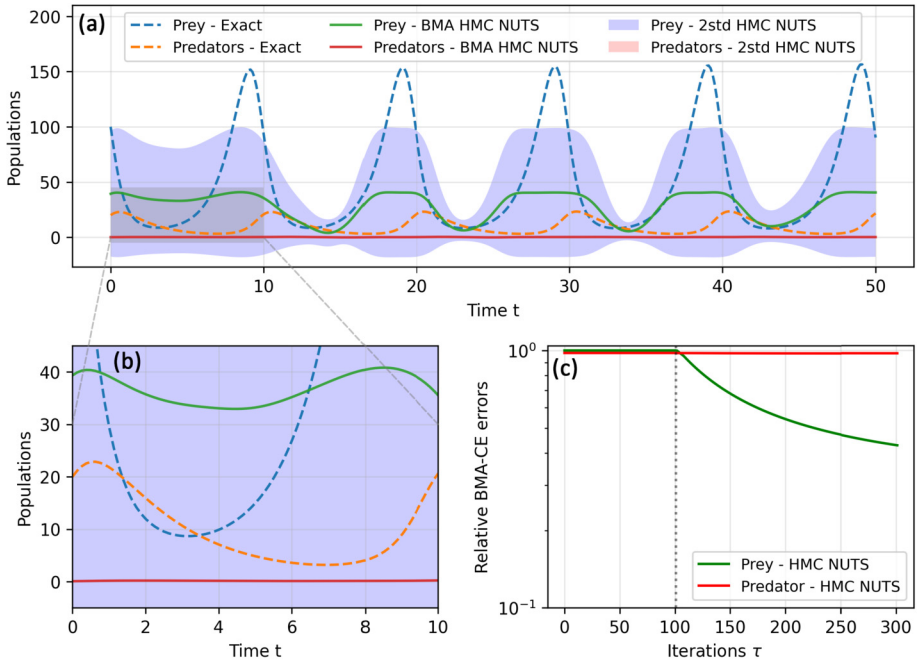

Fig. D.19. Failure mode of classical HMC, with uniform weighting, on the Lotka-Volterra multiscale inverse problem defined in Sect. 3.3. (a) BMA predictions for the two-species populations along the physical time with their uncertainties, with a zoomed version in (b). (c) Relative BMA-CE errors throughout the sampling iterations illustrate the lack of convergence of the method.

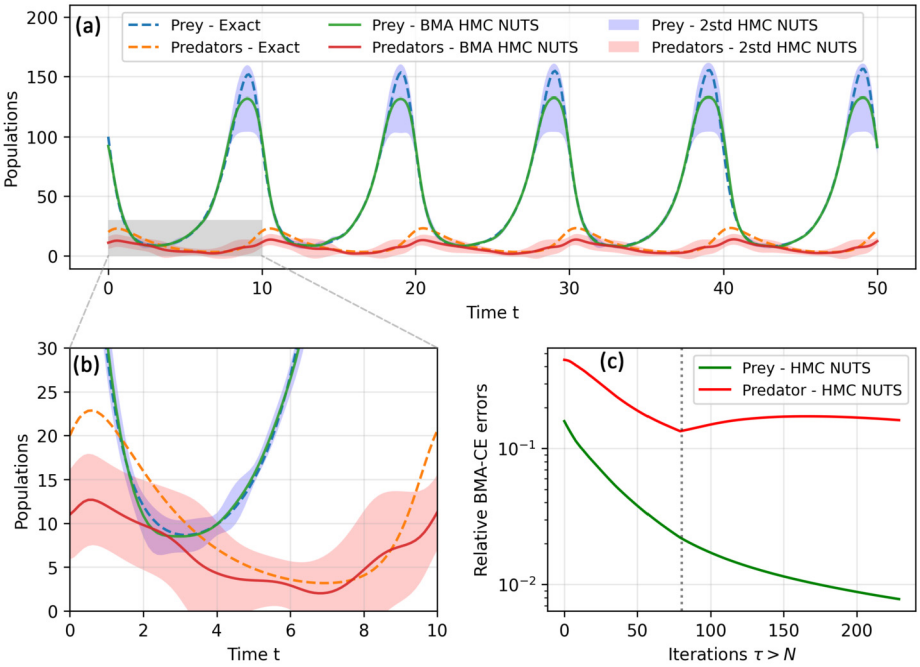

Fig. D.20. Failure mode of HMC with NUTS adaptation on the Lotka-Volterra multiscale inverse problem defined in Sect. 3.3. (a) BMA predictions for the two-species populations along the physical time with their uncertainties, with a zoomed version in (b). (c) Relative BMA-CE errors throughout the sampling iterations showing an imbalance between the tasks and preferential adaptation of the prey population.

measurements of the first-order derivatives of the velocity. This enables us to ensure that the convective terms, in the PDE constraints (37), are consistent with the velocity data and therefore infer the corresponding pressure field. The multi-potential energy is thus written as:

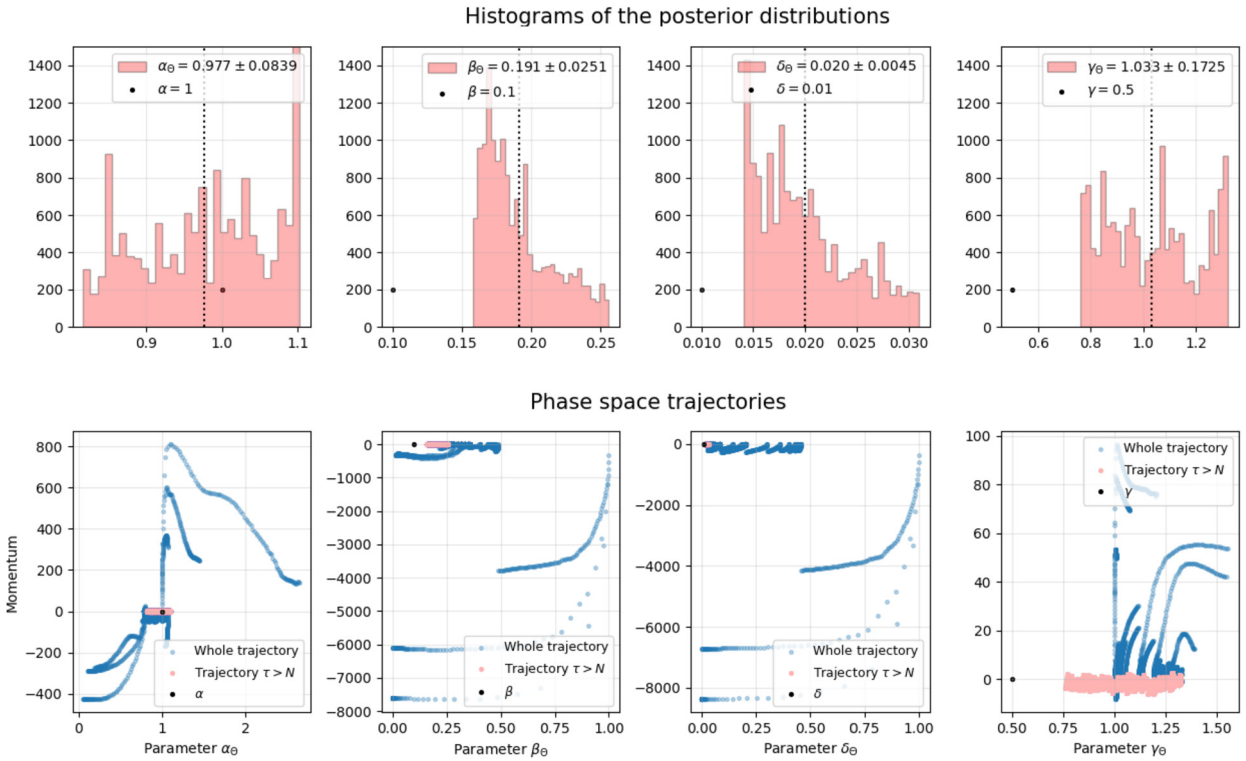

Fig. D.21. Failure mode of HMC with NUTS on the Lotka-Volterra multiscale inference. Top pictures show histograms of the marginal posterior distributions for the inverse parameters. Bottom phase diagrams of their trajectories throughout the sampling. The biased predictions from Fig. D.20 prevent proper inference of the inverse parameters, leading to random walk pathological behavior in the updated parameters.

$$
\begin{aligned}
U(\Theta) = &\frac{\lambda_0}{2\sigma_0^2}\left\|u_\Theta - u\right\|^2_{\mathcal{D}^{\mathbf{u}}} + \frac{\lambda_1}{2\sigma_1^2}\left\|v_\Theta - v\right\|^2_{\mathcal{D}^{\mathbf{u}}} + \frac{\lambda_2}{2\sigma_2^2}\left\|u_\Theta - u\right\|^2_{\mathcal{D}^{\partial}} + \frac{\lambda_3}{2\sigma_3^2}\left\|v_\Theta - v\right\|^2_{\mathcal{D}^{\partial}} \\
&+ \frac{\lambda_4}{2\sigma_4^2}\left\|\partial_x u_\Theta - \partial_x u\right\|^2_{\mathcal{D}^{\mathbf{u}}} + \frac{\lambda_5}{2\sigma_5^2}\left\|\partial_x v_\Theta - \partial_x v\right\|^2_{\mathcal{D}^{\mathbf{u}}} + \frac{\lambda_6}{2\sigma_6^2}\left\|\partial_x u_\Theta - \partial_x u\right\|^2_{\mathcal{D}^{\partial}} + \frac{\lambda_7}{2\sigma_7^2}\left\|\partial_x v_\Theta - \partial_x v\right\|^2_{\mathcal{D}^{\partial}} \\
&+ \frac{\lambda_8}{2\sigma_8^2}\left\|Re_\Theta^{-1}\Delta u_\Theta - (u_\Theta \partial_x u_\Theta + v_\Theta \partial_y u_\Theta) - \partial_x p_\Theta\right\|^2_{\mathcal{D}^{\Omega}} \\
&+ \frac{\lambda_9}{2\sigma_9^2}\left\|Re_\Theta^{-1}\Delta v_\Theta - (u_\Theta \partial_x v_\Theta + v_\Theta \partial_y v_\Theta) - \partial_y p_\Theta\right\|^2_{\mathcal{D}^{\Omega}} \\
&+ \frac{\lambda_{10}}{2\sigma_{10}^2}\left\|\nabla \cdot u_\Theta\right\|^2_{\mathcal{D}^{\Omega}} + \frac{\lambda_{11}}{2\sigma_{11}^2}\left\|\partial_y u_\Theta - \partial_y u\right\|^2_{\mathcal{D}^{\mathbf{u}}} + \frac{\lambda_{12}}{2\sigma_{12}^2}\left\|\partial_y u_\Theta - \partial_y u\right\|^2_{\mathcal{D}^{\partial}} + \frac{1}{2\sigma_\Theta^2}\|\Theta\|^2_{R_{p+1}}
\end{aligned}
\tag{E.1}
$$

where the notation $\|\cdot\|$ refers to either the RMS norm on $\mathcal{D}^\bullet$ or the usual Euclidean norm on $\mathbb{R}^{p+1}$.

References

[1] Naif Alqahtani, Fatimah Alzubaidi, Ryan T. Armstrong, Pawel Swietojanski, Peyman Mostaghimi, Machine learning for predicting properties of porous media from 2d X-ray images, J. Pet. Sci. Eng. 184 (2020) 106514.

[2] Naif J. Alqahtani, Yufu Niu, Ying Da Wang, Traiwit Chung, Zakhar Lanetc, Aleksandr Zhuravljov, et al., Super-resolved segmentation of X-ray images of carbonate rocks using deep learning, Transp. Porous Media 143 (2) (2022) 497–525.

[3] Karim Armanious, Vijeth Kumar, Sherif Abdulatif, Tobias Hepp, Sergios Gatidis, Bin Yang, ipA-MedGAN: inpainting of arbitrary regions in medical imaging, in: 2020 IEEE International Conference on Image Processing (ICIP), 2020, pp. 3005–3009.

[4] Michael Betancourt, A conceptual introduction to Hamiltonian Monte Carlo, arXiv:1701.02434, 2018.

[5] Michael Betancourt, Simon Byrne, Mark Girolami, Optimizing the integrator step size for Hamiltonian Monte Carlo, arXiv:1411.6669, 2014.

[6] Michael Betancourt, Simon Byrne, Sam Livingstone, Mark Girolami, The geometric foundations of Hamiltonian Monte Carlo, Bernoulli 23 (4) (2017) 2257–2298.

[7] George C. Bourantas, Bevan L. Cheeseman, Rajesh Ramaswamy, Ivo F. Sbalzarini, Using DC PSE operator discretization in Eulerian meshless collocation methods improves their robustness in complex geometries, Comput. Fluids 136 (2016) 285–300.

[8] Steven L. Brunton, Joshua L. Proctor, J. Nathan Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proc. Natl. Acad. Sci. 113 (15) (2016) 3932–3937.

[9] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, Andrew Rabinovich, GradNorm: gradient normalization for adaptive loss balancing in deep multitask networks, in: Proceedings of the 35th International Conference on Machine Learning, PMLR, 2018, pp. 794–803.

[10] Roberto Cipolla, Yarin Gal, Alex Kendall, Multi-task learning using uncertainty to weigh losses for scene geometry and semantics, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7482–7491.
[11] Adam D. Cobb, Brian Jalaian, Scaling Hamiltonian Monte Carlo inference for Bayesian neural networks with symmetric splitting, in: Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence, PMLR, 2021, pp. 675–685.
[12] Wojciech M. Czarnecki, Simon Osindero, Max Jaderberg, Grzegorz Swirszcz, Razvan Pascanu, Sobolev training for neural networks, in: Advances in Neural Information Processing Systems, vol. 30, Curran Associates, Inc., 2017.
[13] Marta D'Elia, Hang Deng, Cedric Fraces, Krishna Garikipati, Lori Graham-Brady, Amanda Howard, et al., Machine learning in heterogeneous porous materials, arXiv:2202.04137, 2022.
[14] Salar Fattahi, Somayeh Sojoudi, Data-driven sparse system identification, in: 2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton), IEEE, 2018, pp. 462–469.
[15] Marylou Gabrié, Grant M. Rotskoff, Eric Vanden-Eijnden, Adaptive Monte Carlo augmented with normalizing flows, Proc. Natl. Acad. Sci. 119 (10) (2022) e2109420119.
[16] Mark Girolami, Ben Calderhead, Riemann manifold Langevin and Hamiltonian Monte Carlo methods, J. R. Stat. Soc., Ser. B, Stat. Methodol. 73 (2) (2011) 123–214.
[17] Alex Graves, Practical variational inference for neural networks, in: Advances in Neural Information Processing Systems, vol. 24, Curran Associates, Inc., 2011.
[18] Matthew D. Hoffman, Andrew Gelman, The No-U-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo, J. Mach. Learn. Res. 15 (1) (2014) 1593–1623.
[19] Matthew D. Hoffman, Alexey Radul, Pavel Sountsov, An adaptive-MCMC scheme for setting trajectory lengths in Hamiltonian Monte Carlo, in: Proceedings of the 24th International Conference on Artificial Intelligence and Statistics, PMLR, 2021, pp. 3907–3915.
[20] David W. Hogg, Daniel Foreman-Mackey, Data analysis recipes: using Markov chain Monte Carlo, Astrophys. J. Suppl. Ser. 236 (1) (2018) 11.
[21] Laurène Hume, Philippe Poncet, A velocity-vorticity method for highly viscous 3d flows with application to digital rock physics, J. Comput. Phys. 425 (2021) 109910.
[22] Weiqi Ji, Weilun Qiu, Zhiyu Shi, Shaowu Pan, Sili Deng, Stiff-PINN: physics-informed neural network for stiff chemical kinetics, J. Phys. Chem. A 125 (36) (2021) 8098–8106.
[23] Shiwei Lan, Jeffrey Streets, Babak Shahbaba, Wormhole Hamiltonian Monte Carlo, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 2014, 2014, pp. 1953–1959.
[24] Daniel Levy, Matt D. Hoffman, Jascha Sohl-Dickstein, Generalizing Hamiltonian Monte Carlo with neural networks, in: International Conference on Learning Representations, 2018.
[25] Guang Lin, Yating Wang, Zecheng Zhang, Multi-variance replica exchange SGMCMC for inverse and forward problems via Bayesian PINN, J. Comput. Phys. 460 (2022) 111173.
[26] Kevin Linka, Amelie Schäfer, Xuhui Meng, Zongren Zou, George Em Karniadakis, Ellen Kuhl, Bayesian Physics Informed Neural Networks for real-world nonlinear dynamical systems, Comput. Methods Appl. Mech. Eng. (2022) 115346.
[27] Dehao Liu, Yan Wang, Multi-Fidelity Physics-Constrained Neural Network and its application in materials modeling, J. Mech. Des. 141 (12) (2019).
[28] Qiang Liu, Dilin Wang, Stein variational gradient descent: a general purpose bayesian inference algorithm, in: Advances in Neural Information Processing Systems, vol. 29, Curran Associates, Inc., 2016.
[29] Xu Liu, Wen Yao, Wei Peng, Weien Zhou, Bayesian physics-informed extreme learning machine for forward and inverse PDE problems with noisy data, arXiv:2205.06948, 2022.
[30] Samuel Livingstone, Michael Betancourt, Simon Byrne, Mark Girolami, On the geometric ergodicity of Hamiltonian Monte Carlo, Bernoulli 25 (4A) (2019) 3109–3138.
[31] Suryanarayana Maddu, Bevan L. Cheeseman, Christian L. Müller, Ivo F. Sbalzarini, Learning physically consistent differential equation models from data using group sparsity, Phys. Rev. E 103 (2021) 042310.
[32] Suryanarayana Maddu, Bevan L. Cheeseman, Ivo F. Sbalzarini, Christian L. Müller, Stability selection enables robust learning of differential equations from limited noisy data, Proc. R. Soc. A 478 (2262) (2022) 20210916.
[33] Suryanarayana Maddu, Dominik Sturm, Bevan L. Cheeseman, Christian L. Müller, Ivo F. Sbalzarini, STENCIL-NET: data-driven solution-adaptive discretization of partial differential equations, arXiv:2101.06182, 2021.
[34] Suryanarayana Maddu, Dominik Sturm, Christian L. Müller, Ivo F. Sbalzarini, Inverse Dirichlet weighting enables reliable training of physics informed neural networks, Mach. Learn.: Sci. Technol. 3 (1) (2022) 015026.
[35] Oren Mangoubi, Natesh S. Pillai, Aaron Smith, Does Hamiltonian Monte Carlo mix faster than a random walk on multimodal densities?, arXiv:1808.03230, 2018.
[36] José V. Manjón, José E. Romero, Roberto Vivo-Hernando, Gregorio Rubio, Fernando Aparici, Maria de La Iglesia-Vaya, et al., Blind MRI brain lesion inpainting using deep learning, in: Simulation and Synthesis in Medical Imaging, Springer International Publishing, 2020, pp. 41–49.
[37] Xuhui Meng, Hessam Babaee, George Em Karniadakis, Multi-fidelity Bayesian neural networks: algorithms and applications, J. Comput. Phys. 438 (2021) 110361.
[38] Joseph P. Molnar, Samuel J. Grauer, Flow field tomography with uncertainty quantification using a Bayesian physics-informed neural network, Meas. Sci. Technol. 33 (6) (2022) 065305.
[39] Jan Oldenburg, Finja Borowski, Alper Öner, Klaus-Peter Schmitz, Michael Stiehm, Geometry aware physics informed neural network surrogate for solving Navier–Stokes equation (GAPINN), Adv. Model. Simul. Eng. Sci. 9 (1) (2022) 8.
[40] Jaideep Pathak, Brian Hunt, Michelle Girvan, Zhixin Lu, Edward Ott, Model-free prediction of large spatiotemporally chaotic systems from data: a reservoir computing approach, Phys. Rev. Lett. 120 (Jan 2018) 024102.
[41] Sarah Perez, Peter Moonen, Philippe Poncet, On the deviation of computed permeability induced by unresolved morphological features of the pore space, Transp. Porous Media 141 (1) (2022) 151–184.
[42] Johan Phan, Leonardo C. Ruspini, Frank Lindseth, Automatic segmentation tool for 3D digital rocks by deep learning, Sci. Rep. 11 (1) (2021) 19123.
[43] Philippe Poncet, Roland Hildebrand, Georges-Henri Cottet, Petros Koumoutsakos, Spatially distributed control for optimal drag reduction of the flow past a circular cylinder, J. Fluid Mech. 599 (2008) 111–120.
[44] Apostolos F. Psaros, Xuhui Meng, Zongren Zou, Ling Guo, George Em Karniadakis, Uncertainty quantification in scientific machine learning: methods, metrics, and comparisons, J. Comput. Phys. 477 (2023) 111902.
[45] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred A. Hamprecht, et al., On the spectral bias of neural networks, in: Proceedings of the 36th International Conference on Machine Learning, PMLR, 2019, pp. 5301–5310.
[46] Maziar Raissi, Paris Perdikaris, George Em Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. 378 (2019) 686–707.
[47] Maziar Raissi, Alireza Yazdani, George Em Karniadakis, Hidden fluid mechanics: a Navier-Stokes informed deep learning framework for assimilating flow visualization data, arXiv:1808.04327, 2018.
[48] Maziar Raissi, Alireza Yazdani, George Em Karniadakis, Hidden fluid mechanics: learning velocity and pressure fields from flow visualizations, Science 367 (6481) (2020) 1026–1030.

[49] Franz M. Rohrhofer, Stefan Posch, Bernhard C. Geiger, On the Pareto front of physics-informed neural networks, arXiv:2105.00862, 2021.
[50] Vivekananda Roy, Convergence diagnostics for Markov chain Monte Carlo, Annu. Rev. Stat. Appl. 7 (1) (2020) 387–412.
[51] Javier E. Santos, Ying Yin, Honggeun Jo, Wen Pan, Qinjun Kang, Hari S. Viswanathan, et al., Computationally efficient multiscale neural networks applied to fluid flow in complex 3D porous media, Transp. Porous Media 140 (1) (2021) 241–272.
[52] Hayden Schaeffer, Learning partial differential equations via data discovery and sparse optimization, Proc. R. Soc. A, Math. Phys. Eng. Sci. 473 (2197) (2017) 20160446.
[53] Ozan Sener, Vladlen Koltun, Multi-task learning as multi-objective optimization, in: Advances in Neural Information Processing Systems, vol. 31, Curran Associates, Inc., 2018.
[54] Reza Shams, Mohsen Masihi, Ramin Bozorgmehry Boozarjomehry, Martin J. Blunt, A hybrid of statistical and conditional generative adversarial neural network approaches for reconstruction of 3D porous media (ST-CGAN), Adv. Water Resour. 158 (2021).
[55] Hwijae Son, Jin Woo Jang, Woo Jin Han, Hyung Ju Hwang, Sobolev training for physics informed neural networks, arXiv:2101.08932, 2021.
[56] Luning Sun, Han Gao, Shaowu Pan, Jian-Xun Wang, Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data, Comput. Methods Appl. Mech. Eng. 361 (2020) 112732.
[57] Luning Sun, Jian-Xun Wang, Physics-constrained bayesian neural network for fluid flow reconstruction with sparse and noisy data, Theor. Appl. Mech. Lett. 10 (3) (2020) 161–169.
[58] Shiliang Sun, Jing Zhao, Minghao Gu, Shanhu Wang, Variational hybrid Monte Carlo for efficient multi-modal data sampling, Entropy 25 (4) (2023).
[59] Minh-Trieu Tran, Soo-Hyung Kim, Hyung-Jeong Yang, Guee-Sang Lee, Multi-task learning for medical image inpainting based on organ boundary awareness, Appl. Sci. 11 (9) (2021).
[60] Remco van der Meer, Cornelis W. Oosterlee, Anastasia Borovykh, Optimally weighted loss functions for solving PDEs with neural networks, J. Comput. Appl. Math. 405 (2022) 113887.
[61] Pantelis R. Vlachas, Jaideep Pathak, Brian R. Hunt, Themistoklis P. Sapsis, Michelle Girvan, Edward Ott, et al., Backpropagation algorithms and Reservoir Computing in Recurrent Neural Networks for the forecasting of complex spatiotemporal dynamics, Neural Netw. 126 (2020) 191–217.
[62] Nikolaos N. Vlassis, WaiChing Sun, Sobolev training of thermodynamic-informed neural networks for interpretable elasto-plasticity models with level set hardening, Comput. Methods Appl. Mech. Eng. 377 (2021) 113695.
[63] Sifan Wang, Yujun Teng, Paris Perdikaris, Understanding and mitigating gradient flow pathologies in physics-informed neural networks, SIAM J. Sci. Comput. 43 (5) (2021) A3055–A3081.
[64] Sifan Wang, Xinling Yu, Paris Perdikaris, When and why PINNs fail to train: a neural tangent kernel perspective, J. Comput. Phys. 449 (2022) 110768.
[65] Andrew G. Wilson, Pavel Izmailov, Bayesian deep learning and a probabilistic perspective of generalization, Advances in Neural Information Processing Systems, vol. 33, Curran Associates, Inc., 2020, pp. 4697–4708.
[66] Anqi Wu, Sebastian Nowozin, Edward Meeds, Richard E. Turner, José Miguel Hernández-Lobato, Alexander L. Gaunt, Deterministic variational inference for robust Bayesian neural networks, in: 7th International Conference on Learning Representations, OpenReview.net, 2019.
[67] Liang Yan, Tao Zhou, Adaptive multi-fidelity polynomial chaos approach to Bayesian inference in inverse problems, J. Comput. Phys. 381 (2019) 110–128.
[68] Liang Yan, Tao Zhou, An adaptive surrogate modeling based on deep neural networks for large-scale Bayesian inverse problems, Commun. Comput. Phys. 28 (5) (2020) 2180–2205.
[69] Liu Yang, Xuhui Meng, George Em Karniadakis, B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data, J. Comput. Phys. 425 (2021) 109913.
[70] Yibo Yang, Paris Perdikaris, Adversarial uncertainty quantification in physics-informed neural networks, J. Comput. Phys. 394 (2019) 136–152.
[71] Jiayu Yao, Weiwei Pan, Soumya Shubhra Ghosh, Finale Doshi-Velez, Quality of uncertainty quantification for Bayesian neural network inference, arXiv: 1906.09686, 2019.
[72] Wentao Yuan, Qingtian Zhu, Xiangyue Liu, Yikang Ding, Haotian Zhang, Chi Zhang, Sobolev training for implicit neural representations with approximated image derivatives, in: Computer Vision – ECCV 2022, vol. 13675, Springer Nature Switzerland, 2022, pp. 72–88.
[73] Qiang Zheng, Lingzao Zeng, George Em Karniadakis, Physics-informed semantic inpainting: application to geostatistical modeling, J. Comput. Phys. 419 (2020) 109676.