Uncertainty-Aware Surrogate-based Amortized Bayesian Inference for Computationally Expensive Models

Anonymous authors
Paper under double-blind review

Abstract

Bayesian inference typically relies on a large number of model evaluations to estimate posterior distributions. Established methods like Markov Chain Monte Carlo (MCMC) and Amortized Bayesian Inference (ABI) can become computationally challenging. While ABI enables fast inference *after* training, generating sufficient training data still requires thousands of model simulations, which is infeasible for expensive models. Surrogate models offer a solution by providing *approximate* simulations at a lower computational cost, allowing the generation of large data sets for training. However, the introduced approximation errors and uncertainties can lead to overconfident posterior estimates. To address this, we propose Uncertainty-Aware Surrogate-based Amortized Bayesian Inference (UA-SABI) – a framework that combines surrogate modeling and ABI while explicitly quantifying and propagating surrogate uncertainties through the inference pipeline. Our experiments show that this approach enables reliable, fast, and repeated Bayesian inference for computationally expensive models, even under tight time constraints.

1 Introduction

Mathematical models are essential for simulating real-world processes, typically mapping parameters to observable data. Estimating these parameters from real-world observations, i.e., the inverse problem, is fundamental across scientific disciplines (e.g. Etz & Vandekerckhove, 2018; Von Toussaint, 2011; Hülsenbeck et al., 2001; Ellison, 2004). However, because models never perfectly capture reality and observed data are often sparse and imprecise, parameter estimation inherently involves uncertainty. Bayesian inference offers a systematic framework for estimating parameters while incorporating uncertainty in a statistically grounded manner (e.g. Gelman et al., 1995).

Markov Chain Monte Carlo (MCMC) methods are widely used for Bayesian inference to generate high-quality samples from the posterior distribution given fixed observations (Gilks et al., 1995). However, MCMC is computationally expensive and slow, rendering it impractical in scenarios where near-instant inference is required (Robert et al., 2018). Near-instant inference is, for example, necessary in adaptive robotic control or closed-loop medical devices, where parameters such as object mass or patient sensitivity must be estimated immediately to allow accurate prediction and safe action (e.g. Marlier, 2024; Tasoujian et al., 2020; Malagutti et al., 2023). Furthermore, each new set of observations requires restarting the entire process, further increasing costs when inference is needed for multiple datasets. Multiple inference runs may be needed for ongoing adaptation as conditions evolve or more data become available. Tracking the spread of a disease such as influenza or COVID-19 is one example of this. New case data continuously arrive, and separate datasets exist for different regions, requiring repeated Bayesian updates to infer transmission rates or reproduction numbers for each location (e.g. Radev et al., 2021; Yang et al., 2015). MCMC also relies on a known likelihood function that can be evaluated either analytically or numerically.

Amortized Bayesian Inference (ABI), a deep-learning-based approach originating from simulation-based inference (SBI), addresses these limitations (Cranmer et al., 2020; Radev et al., 2020; Lückmann et al., 2021). SBI methods are typically employed when evaluating the likelihood is infeasible, but simulations from the model are possible, allowing learning a mapping from observed data to the posterior of the model parameters

via simulated data. ABI in particular leverages generative neural networks to learn this mapping. Training data is generated through multiple model evaluations. Once trained, ABI enables near-instant posterior inference for new datasets, as the computational cost is incurred during training. Being likelihood-free makes ABI well-suited for complex problems involving noisy or high-dimensional data, where evaluating the likelihood is infeasible or unreliable. However, the learned posterior only approximates the true posterior, and high accuracy requires well-designed network architectures and extensive training data.

Both MCMC and ABI face limitations when the simulation model is computationally expensive, since both require a large number of model evaluations. MCMC requires many likelihood evaluations per generated posterior sample; ABI requires extensive model simulations to generate sufficient amounts of training data. As a result, if computational time is limited, obtaining a good posterior becomes infeasible for both approaches. Note that such computational effort can arise for different reasons – for instance, from a high-dimensional parameter space or from the complexity of the governing equations. In Earth sciences, for example, low-dimensional models can be computationally very expensive, particularly if they involve (coupled systems of) non-linear partial differential equations or long simulated time periods (e.g. Mohammadi et al., 2018; Hommel et al., 2015).

Our goal is to enable ABI for computationally expensive models to benefit from the aforementioned advantages that ABI offers. To this end, surrogate models are the crucial tool, allowing us to reduce the computational cost of expensive simulations for training data generation. Surrogate models (Sudret, 2008; Rasmussen & Williams, 2006), however, are only approximations (i.e., imperfect representations) of the reference simulation, and therefore not necessarily reliable. It is crucial to quantify the uncertainties associated with surrogate modeling, such as those arising from limited training data or any inherent inflexibility of the surrogate. These uncertainties then need to be propagated through the inference pipeline, ensuring consistent posterior estimation. Previous methods have integrated these uncertainties into MCMC methods with surrogate models (e.g. Lückmann et al., 2018; Zhang et al., 2020; Reiser et al., 2025). However, this requires many additional MCMC runs to incorporate the surrogate uncertainty, largely eliminating the computational advantages of surrogates.

We introduce Uncertainty-Aware Surrogate-based Amortized Bayesian Inference (UA-SABI), which enables ABI for computationally expensive models while accounting for the uncertainty inherent in surrogate approximations. By leveraging a surrogate, we eliminate the training error that arises from insufficient data during ABI training, as simulating from the surrogate is easy and fast. This comes at the cost of introducing a surrogate approximation error. However, unlike the ABI training error, the surrogate approximation error can be quantified and propagated, enabling reliable ABI for computationally expensive models.

For this purpose, we adapt and further develop the uncertainty estimation and propagation method for surrogate modeling proposed in Reiser et al. (2025). Our main novelty is to replace the MCMC with ABI for inference. This novelty and its benefits can be viewed from two perspectives: 1) Replacing MCMC with ABI substantially improves sampling efficiency, as parallel chains for draws from the surrogate posterior are no longer required. 2) UA-SABI enables ABI training with surrogate data while accounting for the uncertainty inherent in the surrogate approximation. This significantly reduces the required runs from the expensive simulation model while retaining the amortization property and, therefore, enabling ABI for computationally expensive models.

An illustration of the workflow, including training phases and inference, is given in Fig. 1. We demonstrate the workflow and substantiate the claimed advantages both theoretically and practically with a toy example and two real-world case studies.

2 Methods

2.1 Amortized Bayesian Inference (ABI)

This work focuses on using ABI to infer the parameters of computationally expensive simulation models. The typical ABI procedure (Radev et al., 2020) can be divided into a training phase and an inference

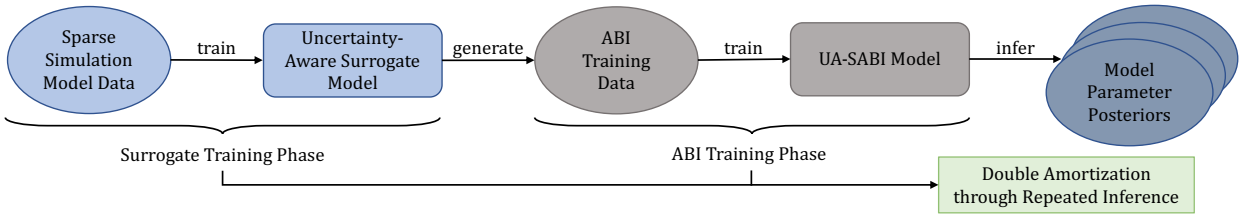


Figure 1: Illustrative workflow of UA-SABI training and inference.

phase. Training includes simulating data and then training a neural posterior estimator (NPE) on that data (Papamakarios & Murray, 2016; Durkan et al., 2019; Greenberg et al., 2019b).

ABI Training Data Generation Training data is generated using a simulation model $M = M(\mathbf{x}, \boldsymbol{\omega})$, typically a realistic first-principles model. M depends on inputs $\mathbf{x}$ and parameters $\boldsymbol{\omega}$ and yields an output $\mathbf{y}$. Additionally, it can contain stochastic parts omitted in the following to simplify the notation. To learn the mapping from observations to posterior draws, a training set $D_B = \{(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^{N_B}$ of N_B samples must first be generated. This can be performed by simulating N_B pairs of parameters, inputs and outputs from the simulation model:

$$\mathbf{y}^{(i)} = M(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}) \quad \text{with} \quad (\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}) \sim p(\mathbf{x}, \boldsymbol{\omega}). \tag{1}$$

Neural Posterior Estimation After creating the training dataset, the NPE, including a summary network $S_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$ and an inference network $I_{\boldsymbol{\varphi}}(\mathbf{s})$, must be trained. The goal of $S_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$ is to embed (multiple) model inputs and outputs $(\mathbf{x}, \mathbf{y})$ of potentially variable length (e.g., a measurement series) in a fixed-length vector $\mathbf{s}$ that serves as input to $I_{\boldsymbol{\varphi}}(\mathbf{s})$. Then, $I_{\boldsymbol{\varphi}}(\mathbf{s})$ generates samples from the approximate posterior $q_{\boldsymbol{\varphi}}(\boldsymbol{\omega} \mid \mathbf{s})$. Here, $\boldsymbol{\theta}$ and $\boldsymbol{\varphi}$ are learnable coefficients in $S_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$ and $I_{\boldsymbol{\varphi}}(\mathbf{s})$. In our case studies, we use coupling flows (Papamakarios et al., 2021) as inference networks. Coupling flows have been empirically and theoretically shown to be expressive and allow for fast evaluation and sampling (Draxler et al., 2025). However, ABI is generally flexible in its choice for the inference network and works with any generative neural network (e.g. Ardizzone et al., 2019; Padmanabha & Zabaras, 2021; Denker et al., 2021; Wildberger et al., 2023; Schmitt et al., 2024b). Thus, the summary and inference networks are modular and can be independently replaced or adapted to suit the specific problem at hand.

The summary and inference networks are trained jointly by minimizing the Kullback-Leibler (KL) divergence between the true posterior $p(\boldsymbol{\omega} \mid \mathbf{x}, \mathbf{y})$ and the approximate posterior $q_{\boldsymbol{\varphi}}(\boldsymbol{\omega} \mid \mathbf{s})$:

$$L(\boldsymbol{\varphi}, \boldsymbol{\theta}) = \mathbb{E}_{p(\mathbf{y})}[\mathrm{KL}(p(\boldsymbol{\omega} \mid \mathbf{x}, \mathbf{y}) \mid q_{\boldsymbol{\varphi}}(\boldsymbol{\omega} \mid \mathbf{s}))] \approx \sum_{i=1}^{N_B} (-\log\, q_{\boldsymbol{\varphi}}(\boldsymbol{\omega}^{(i)} \mid S_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}))) \tag{2}$$

This expectation is approximated via the training data D_B obtained from the (potentially expensive) simulation model (Radev et al., 2020).

2.2 Uncertainty-Aware Surrogate Modeling

Uncertainty-aware surrogate modeling is introduced in and adapted from Reiser et al. (2025). A surrogate model $\widetilde{M}$ aims to approximate the behavior of a simulation model M with negligible computational cost. The surrogate model is typically parametrized by learnable surrogate coefficients, denoted as $\mathbf{c}$. The simulation model output can then be approximated by

$$\mathbf{y} = M(\mathbf{x}, \boldsymbol{\omega}) \approx \widetilde{M}_{\mathbf{c}}(\mathbf{x}, \boldsymbol{\omega}) = \widetilde{\mathbf{y}}. \tag{3}$$

However, due to limited simulation data from M used for training $\widetilde{M}_{\mathbf{c}}$, the surrogate coefficients $\mathbf{c}$ can only be estimated with substantial epistemic uncertainty. For the same reason, the expressibility of $\widetilde{M}_{\mathbf{c}}$ must

be restricted, i.e., the complexity of $\widetilde{M}_{\mathbf{c}}$ must be kept relatively low. This introduces an approximation error, denoted as ϵ, which captures the misspecification of the surrogate. In practice, one usually has to expect a non-negligible approximation error due to limitations in model capacity, optimization challenges, or regularization constraints. We assume the approximation error $\epsilon \sim p(\epsilon \mid \sigma)$ to follow a distribution parametrized by σ, leading to the error-adjusted surrogate output:

$$\widetilde{\mathbf{y}}_\epsilon = f(\widetilde{\mathbf{y}}, \epsilon) \quad \text{with} \quad \epsilon \sim p(\epsilon \mid \sigma). \tag{4}$$

This setup defines the surrogate likelihood

$$p(\widetilde{\mathbf{y}}_\epsilon \mid \widetilde{\mathbf{y}}, \sigma) = p(\widetilde{\mathbf{y}}_\epsilon \mid \mathbf{x}, \boldsymbol{\omega}, \mathbf{c}, \sigma), \tag{5}$$

that approximates the (usually unknown) likelihood of the simulation model.

Using **sparse** training data ($N_T \ll N_B$) obtained from the simulation model $D_T = \{(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^{N_T}$, with $\mathbf{y}^{(i)} = M(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)})$ and $(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)})$ sampled from a given prior, we infer the surrogate model coefficients $\mathbf{c}$ and σ via Bayesian inference. This enables capturing both the surrogate model's epistemic uncertainty and the irreducible approximation error. For this purpose, we consider the likelihood equation 5 and the prior $p(\mathbf{c}, \sigma)$. The joint posterior over the surrogate parameters and the approximation noise, given the training data, is then

$$p(\mathbf{c}, \sigma \mid D_T) \propto \prod_{i=1}^{N_T} p(\mathbf{y}^{(i)} \mid \mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}, \mathbf{c}, \sigma)\, p(\mathbf{c}, \sigma), \tag{6}$$

assuming an i.i.d. sampled approximation error. This assumption serves as a simple stand-in, since surrogate construction is not the focus of this work. It suffices to demonstrate that even a basic error model outperforms making no error assumption at all. In general, UA-SABI is not limited to the i.i.d. case; case-specific surrogate error models are fully compatible with the method.

The conditions under which $\widetilde{M}$ is considered optimal are implied by the assumed distributions of $\mathbf{x}$ and $\boldsymbol{\omega}$ in D_T and the specified error model $p(\epsilon \mid \sigma)$. Sampling-based algorithms such as MCMC can be used to estimate the surrogate posterior equation 6. For the considered surrogates, this is tractable since their likelihoods are fast to evaluate, and the surrogate posterior – compared to model parameter posteriors across datasets – is estimated only once. While this formulation fully captures surrogate uncertainty, we now temporarily simplify the setup by omitting this uncertainty to introduce an uncertainty-*unaware* baseline method – Surrogate-based ABI. This allows us to isolate and demonstrate the benefit of propagating surrogate uncertainty through to inference.

2.3 Surrogate-based ABI (SABI)

The quality of the NPE depends on a sufficient training budget D_B to ensure that $p(\boldsymbol{\omega} \mid \mathbf{x}, \mathbf{y})$ is well approximated via $q_{\boldsymbol{\varphi}}(\boldsymbol{\omega} \mid S_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y}))$. However, for the expensive models considered here, generating sufficient ABI training data becomes infeasible. To mitigate this, we propose using a surrogate model of the expensive simulator to efficiently generate sufficient training data. Surrogate-based ABI (SABI) thus differs from standard ABI only in the generation of training data. Instead of using the expensive model, a (point) surrogate model with a point estimate $\overline{\mathbf{c}}$ is used to generate the training data $D_B = \{(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}, \widetilde{\mathbf{y}}^{(i)})\}_{i=1}^{N_B}$:

$$\widetilde{\mathbf{y}}^{(i)} = \widetilde{M}_{\overline{\mathbf{c}}}(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}) \quad \text{with} \quad (\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}) \sim p(\mathbf{x}, \boldsymbol{\omega}). \tag{7}$$

For the point surrogate, the median of the posterior in Eq. (6) can be used as it is robust to outliers and invariant under monotonic transformations, but also the mean is a valid option.

However, in general, $\widetilde{M}_{\mathbf{c}}$ is only an approximation of the simulation model M. Therefore, the SABI posterior trained with surrogate data $p(\boldsymbol{\omega} \mid \mathbf{x}, \widetilde{\mathbf{y}})$ will be systematically different from the true posterior $p(\boldsymbol{\omega} \mid \mathbf{x}, \mathbf{y})$, as no surrogate uncertainties or errors are considered in the ABI training process. Specifically, neither the epistemic uncertainty of $\widetilde{M}_{\mathbf{c}}$, nor its irreducible approximation error is propagated. As we show in our experiments, this can lead to highly inaccurate posterior approximations.

2.4 Uncertainty-Aware Surrogate-based ABI (UA-SABI)

To ensure reliable inference, the uncertainty of the surrogate model needs to be propagated through the ABI training and inference procedure, as illustrated in Fig. 1. This results in our proposed Uncertainty-Aware Surrogate-based Amortized Bayesian Inference (UA-SABI) approach.

Instead of generating training data with the point surrogate model, UA-SABI utilizes the uncertainty-aware surrogate model:

$$\widetilde{\mathbf{y}}_\epsilon^{(i)} \sim p(\widetilde{\mathbf{y}}_\epsilon \mid \mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}, \mathbf{c}^{(i)}, \sigma^{(i)}) \quad \text{with} \quad (\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}) \sim p(\mathbf{x}, \boldsymbol{\omega}),\ (\mathbf{c}^{(i)}, \sigma^{(i)}) \sim p(\mathbf{c}, \sigma \mid D_T). \tag{8}$$

To propagate uncertainty to the ABI training, we now additionally draw the surrogate coefficients $\mathbf{c}^{(i)}$ and the approximation error parameter $\sigma^{(i)}$ from the surrogate posterior equation 6. These coefficients are then used to evaluate the surrogate model and obtain $\widetilde{\mathbf{y}}^{(i)}$. The pair $(\widetilde{\mathbf{y}}^{(i)}, \sigma^{(i)})$ is subsequently used to generate a sample $\widetilde{\mathbf{y}}_\epsilon^{(i)}$ from the distribution of the error-adjusted surrogate output. We now approximate the KL divergence in Eq. (2) as:

$$L(\boldsymbol{\varphi}, \boldsymbol{\theta}) \approx \sum_{i=1}^{N_B} (-\log\, q_{\boldsymbol{\varphi}}(\boldsymbol{\omega}^{(i)} \mid S_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}, \widetilde{\mathbf{y}}_\epsilon^{(i)}))) \tag{9}$$

A pseudocode summarizing the UA-SABI training procedure can be found in Appendix A.

3 Related Work

Multi-Fidelity Approaches in SBI Multi-fidelity methods leverage models or simulations of varying accuracy and cost to accelerate inference, combining cheap, approximate simulations with a smaller number of expensive, high-fidelity runs. In Approximate Bayesian Computation (ABC), such strategies reduce computational burden: Warne et al. (2018) apply multi-level Monte Carlo variance reduction to rejection sampling in ABC, improving efficiency. Prescott & Baker (2020) introduce a multi-fidelity ABC approach that combines early acceptance and rejection of parameter samples and selectively runs high-fidelity simulations based on low-fidelity outputs, balancing computational cost and accuracy. Building on this, Prescott & Baker (2021) integrate the multi-fidelity strategy with sequential Monte Carlo sampling, further enhancing efficiency by more effectively exploring the parameter space for MF-ABC proposals. More recent work by Warne et al. (2022) integrates low- and high-fidelity simulations within a multilevel Monte Carlo framework to accelerate SBI for partially observed stochastic processes, using inexpensive approximations to preselect parameters and high-fidelity simulations to refine posterior estimates, while Prescott et al. (2024) dynamically select between low- and high-fidelity models based on predictive accuracy, optimizing resource allocation. In multi-fidelity simulation-based Inference (MF-SBI) (Krouglova et al., 2025) a neural posterior estimator is first pre-trained on large amounts of low-fidelity data and then fine-tuned with a smaller set of high-fidelity simulations. While sharing similar goals, none of the aforementioned methods explicitly take the uncertainty of the low-fidelity simulations into account. UA-SABI's uncertainty-awareness guards against overconfidence in regions with scarce data – a protection which appears not to be in place for the low-fidelity training regions in MF-SBI, for example. Further, unlike many of the multi-fidelity methods, UA-SABI does not require domain expertise for the construction of the surrogate.

Sequential Neural Posterior Estimation (S-NPE) S-NPE iteratively refines posterior estimates by adaptively choosing simulations that are most informative, improving efficiency. Papamakarios & Murray (2016) use initial simulations to fit a Bayesian conditional density estimator for the posterior and then sample parameters from the current posterior estimate to guide the next round of simulations. Similarly, Toni et al. (2009) introduced Sequential ABC (ABC-SMC), which iteratively refines posterior distributions by progressively tightening acceptance thresholds. Kulkarni & Moritz (2023) build on ABC-SMC, leveraging massively parallel architectures, such as GPUs, to more efficiently explore the parameter space. Lueckmann et al. (2017) sequentially train a Bayesian mixture-density network. Greenberg et al. (2019a) developed Automatic Posterior Transformation, which employs flexible flow-based density estimators and dynamically updated proposal distributions. Deistler et al. (2022) introduce truncated S-NPE, restricting simulations to

truncated regions around high-probability areas. However, UA-SABI and S-NPE pursue related but distinct goals: UA-SABI focuses on fully amortized inference, enabling fast posterior estimation for new datasets through one-time training, whereas S-NPE sequentially refines the posterior on the same dataset to improve accuracy and sample efficiency across rounds. Additionally, the referenced S-NPE procedures implement an active learning strategy for simulation selection, assuming data are generated iteratively, while UA-SABI can also operate on pre-existing simulation datasets (offline training).

Gaussian Process Surrogates in SBI Several methods employ Gaussian Process (GP) surrogates to improve sampling efficiency in likelihood-free inference. Meeds & Welling (2014) construct a GP to emulate the expensive simulator and use its predictive mean and uncertainty to estimate the acceptance probability in ABC, avoiding simulator calls when the surrogate is sufficiently confident. Wilkinson (2014) fit a GP to the ABC log-likelihood to prune implausible parameter regions before sampling, greatly reducing simulation cost. Bayesian Optimization likelihood-free inference (Gutmann & Corander, 2016) fits a GP to the discrepancy between simulated and observed data and uses Bayesian optimization to adaptively choose simulator parameters, concentrating effort on informative regions and enabling accurate posterior estimation with few simulations. As in UA-SABI, these approaches apply surrogates, focusing specifically on GPs. However, they employ the surrogate to intelligently select simulation parameters for subsequent likelihood-free inference, whereas UA-SABI uses the surrogate itself to directly generate training data and is agnostic to the surrogate model class.

Structure-Aware SBI Methods Several computational strategies exploit the structure of simulators or information about the inference problem to improve efficiency in SBI. Compositional simulation-based inference for time series (Gloeckler et al., 2025) breaks long, high-dimensional time series into local transitions, performing inference on each step and composing the results to recover a global posterior. Cost-aware simulation-based inference (Bharti et al., 2025) tackles heterogeneous simulation costs by prioritizing cheaper and more informative simulations using a combination of rejection and importance sampling, introducing a cost function to guide allocation of simulation resources. In contrast to UA-SABI, structure-aware methods exploit knowledge about the simulator, whereas UA-SABI treats the simulator as a black box and requires no such knowledge.

Neural Likelihood Estimation Neural likelihood estimation is a likelihood-free inference approach that models the likelihood function directly. Sequential neural likelihood (Papamakarios et al., 2019) uses autoregressive neural flows and focuses simulations on high-posterior regions for efficient inference. Bayesian synthetic likelihood (Price et al., 2018) approximates the likelihood of summary statistics with a multivariate normal, incorporating their uncertainty to provide robust and accurate inference. Jointly Amortized Neural Approximation (JANA) (Radev et al., 2023a) combines neural likelihood and posterior estimation. In addition to learning a neural posterior, it simultaneously learns a neural likelihood to enable marginal likelihood estimation, which is essential for tasks such as Bayesian model selection. However, its objective differs slightly, as it aims to learn both the likelihood, serving as a surrogate model, and the posterior jointly. Such a highly parameterized neural likelihood requires a sufficient amount of training data (Frazier et al., 2024). While UA-SABI also requires simulation data to train a surrogate, it achieves effective inference with far fewer simulations compared to neural likelihood estimation, making it more practical for very expensive models.

Low-Budget Simulation-based Inference with Bayesian Neural Networks Delaunoy et al. (2024) also focus on the challenge of limited training data in ABI. They employ Bayesian Neural Networks (BNNs), which model uncertainty via distributions over network parameters. However, selecting an appropriate distribution and defining a meaningful prior for weights and biases remains speculative and can undermine posterior reliability, as noted in (Delaunoy et al., 2024). Moreover, BNNs still demand substantial training data and compute resources (~25,000 GPU hours in Delaunoy et al. (2024)), thus only shifting computational resources from evaluating expensive simulation models to training BNNs.

Uncertainty-Aware Surrogate-Based Inference Reiser et al. (2025) consider the same setting, replacing a computationally expensive model with an uncertainty-aware surrogate. However, their inference

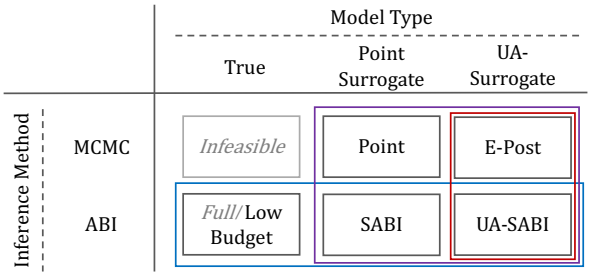


Figure 2: Context of UA-SABI and its alternatives under tight computational constraints.

approach differs: they run MCMC separately for each observation set and for every surrogate sample. This results in a computational cost that scales with both the number of observations and the number of surrogate samples, making their method significantly more expensive. Their proposed method, E-Post, marginalizes over the surrogate coefficient posterior $p(\mathbf{c}, \sigma \mid D_T)$ to obtain the posterior of the parameters of interest $\boldsymbol{\omega}$:

$$p(\boldsymbol{\omega} \mid \mathbf{y}) = \iint p(\boldsymbol{\omega} \mid \mathbf{y}, \mathbf{c}, \sigma)\, p(\mathbf{c}, \sigma \mid D_T)\, d\mathbf{c}\, d\sigma. \tag{10}$$

In practice, this integral is approximated via a Monte Carlo integration with a sufficient number of samples from the surrogate posterior $p(\mathbf{c}, \sigma \mid D_T)$.

During inference with UA-SABI, this explicit marginalization is not required, as the NPE is trained over the entire posterior space of both the outputs and the surrogate parameters. This results in a form of double amortization of the surrogate and ABI training costs through repeated inference. Since UA-SABI and E-Post share the same overall objective and framework, E-Post with MCMC inference serves as a benchmark for UA-SABI. A formal proof of asymptotic equivalence is given in Proposition 1 in Appendix B.

4 Case Studies

We evaluate and benchmark UA-SABI against its uncertainty-unaware counterpart SABI, the corresponding MCMC-based methods, and (if possible) standard ABI using a toy example and two real-world case studies. *All code and data will be made publicly available upon acceptance.*

4.1 Objectives

We aim to validate the parameter posterior obtained with UA-SABI by assessing its quality and justifying the computational effort required to train the surrogate by reducing the effort to train the ABI model. Our central research question is "Can we train ABI with surrogate data, do we have to, and, if so, how should we account for the uncertainty introduced by the surrogate?". To keep focus on that, we deliberately remove unrelated complexities. Specifically, in our real-world case studies, we use computationally expensive but low-dimensional problems. High-dimensional models present significant challenges for surrogate training, as they typically require substantially more data to adequately capture the model's behavior. Additionally, inference on high-dimensional problems becomes more complex due to multi-modality and non-identifiability issues that complicate posterior estimation. This would make result interpretation less transparent and potentially obscure insights about the proposed approach itself.

We want to show the advantage of using surrogate data instead of scarce simulation data, and demonstrate the importance of quantifying and propagating the uncertainty of the surrogate model. To do so, we compare UA-SABI, SABI, and standard ABI for the true model – trained with full and low budget (shown in blue in Fig. 2). To validate the posteriors produced by these methods, we also compare them with those obtained using the corresponding MCMC-based approach (shown in purple in Fig. 2). Further, to highlight the efficiency of our method, we compare the runtimes of UA-SABI and E-Post (shown in red in Fig. 2).

To assess posterior calibration for our first two case studies, we generate multiple synthetic ground truth parameters and perform inference on the simulated datasets. For each dataset, we compute the rank of the ground truth within the posterior samples and summarize the results using empirical cumulative distribution function (ECDF) difference plots – a procedure known as simulation-based calibration (SBC) checking (Talts et al., 2020; Säilynoja et al., 2022; Modrák et al., 2023). To compare runtimes, we estimated how many observation sets are needed before the accumulated computational cost of repeated E-Post inference runs exceeds the total cost of UA-SABI, including both training and its repeated (quasi-instant) inference.

General Setup For all case studies, we use polynomial chaos expansion (PCE) for the surrogate model (Wiener, 1938; Sudret, 2008; Oladyshkin & Nowak, 2012), as it is a fast-to-evaluate and fast-to-train surrogate and a flexible black-box model which can be applied to (almost) arbitrary simulation models. In principle, PCE is just one example among many possible choices – any uncertainty-aware surrogate could be used and replaced in a modular fashion depending on the application domain. A deterministic PCE constructs the surrogate through a spectral projection onto orthogonal (w.r.t. $p(\mathbf{x}, \boldsymbol{\omega})$) polynomial basis functions, expressed as

$$\widetilde{M}_{\mathbf{c}}(\mathbf{x}, \boldsymbol{\omega}) = \sum_{j=0}^{J} c_j \cdot \Psi_j(\mathbf{x}, \boldsymbol{\omega}), \tag{11}$$

with $\boldsymbol{\Psi} = \{\Psi_j\}_{j=0}^{J}$ being the multivariate orthogonal polynomial basis and $\mathbf{c} = \{c_j\}_{j=0}^{J}$ the corresponding coefficients. The number of expansion terms J is computed via the standard truncation scheme (Sudret, 2008). For our case studies, we choose a relatively low polynomial degree, as we operate in a data-scarce regime and aim to avoid overfitting. In the real-world case studies, the underlying models are highly computationally expensive, so a higher polynomial degree would require additional model runs that could exceed the available computational budget. In general, selecting the appropriate surrogate complexity is important but challenging to determine a priori, as it depends on the specific application.

To construct a Bayesian PCE (Shao et al., 2017; Bürkner et al., 2023), we define a prior distribution over the surrogate coefficients $p(\mathbf{c})$ and the approximation error parameter $p(\sigma)$. We choose a normal likelihood:

$$p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\omega}, \mathbf{c}, \sigma) = \mathcal{N}(M(\mathbf{x}, \boldsymbol{\omega}) \mid \widetilde{M}_{\mathbf{c}}(\mathbf{x}, \boldsymbol{\omega}), \sigma). \tag{12}$$

Employing Hamiltonian Monte Carlo (Hoffman & Gelman, 2014) yields samples from the surrogate posterior $p(\mathbf{c}, \sigma \mid D_T)$ (Bürkner et al., 2023). The resulting Bayesian PCEs serve as our uncertainty-aware surrogate.

4.2 Case Study 1: LogSin Model

We first evaluate our surrogate-based ABI approaches in a simple synthetic scenario: a simulation model with one parameter ω and a one-dimensional input x:

$$y = M(x, \omega) = \omega \log(x) + \sin(0.05x) + 0.01x + 1. \tag{13}$$

4.2.1 Setup

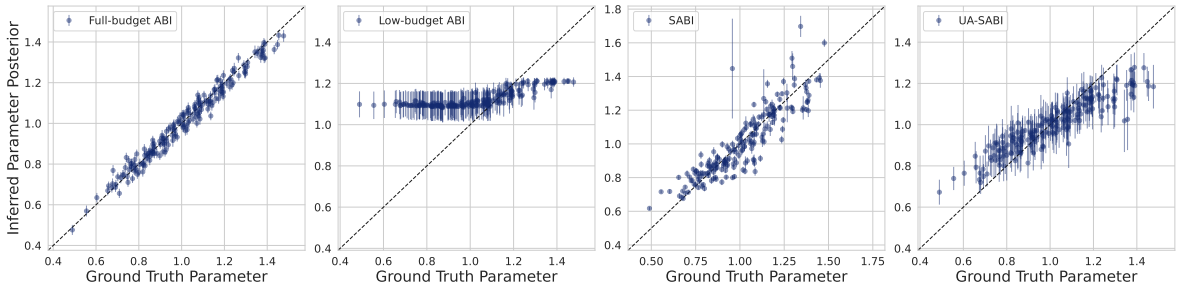
We generate surrogate training data by evaluating the simulation model M at the first 16 points of a two-dimensional Sobol sequence (Sobol', 1967), scaled to $[1, 200]$ for x and $[0.6, 1.4]$ for ω. The resulting input-output pairs are used to train a Bayesian PCE. This introduces significant surrogate uncertainty and approximation error, as a perfect match to the simulation model is unattainable. The surrogate's posterior is sampled using HMC via *Stan* (Carpenter et al., 2017; Stan Development Team, 2024). We employ 4 chains, each with 1,000 warm-up and 250 sampling iterations, yielding 1,000 surrogate posterior samples in total to be propagated.

To perform inference on observation sets, we sample 200 ground truth parameters ω^* from a prior $p(\omega) = \mathcal{N}(1, 0.2)$ and generate four $(\mathbf{x}, \mathbf{y})$ observations for each. We compare the inference results of our proposed surrogate-based ABI to surrogate-based MCMC methods. For ABI, we used an equivariant deep set summary network (Zaheer et al., 2017) and a coupling flow inference network. We employ online training, where newly

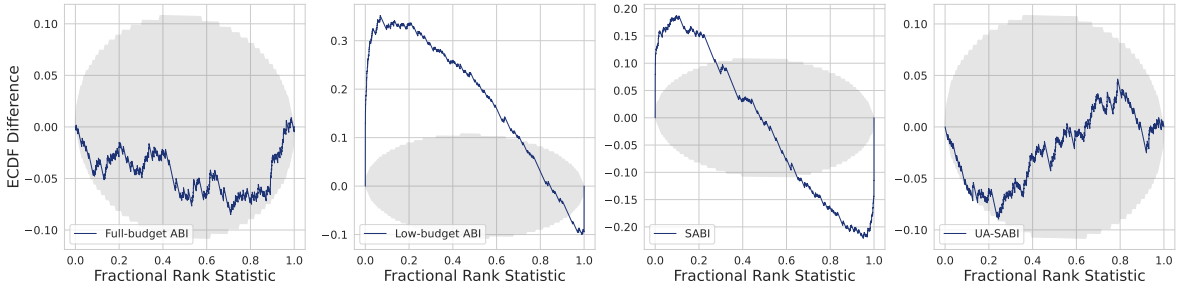
sampled surrogate outputs are used at each iteration. All ABI models were trained for 100 epochs, with a batch size of 64 and 128 batches per epoch, using *BayesFlow* (Radev et al., 2023b). For given observations, ABI generates 4,000 posterior samples via the inference network. Surrogate-based MCMC (Point) draws 4,000 samples using 4 chains (1,000 warm-up and 1,000 sampling iterations). For E-Post, we run MCMC separately for each surrogate posterior draw (1,000 warm-up, 4 sampling iterations per run), a process that is embarrassingly parallelizable but still computationally expensive. Further computational details are given in Appendix C.

4.2.2 Impact of Uncertainty Propagation

First, we compare the performance of our two surrogate-based methods, SABI and UA-SABI, against each other. Then we compare them against a low-budget ABI, trained with the same simulation data as used for surrogate training, and a full-budget ABI trained on sufficient simulation data. Figure 3a shows corresponding recovery plots. They show the posterior median (circles) and median deviation (vertical lines) for any given ground truth value ω^*.



(a) Recovery plots.



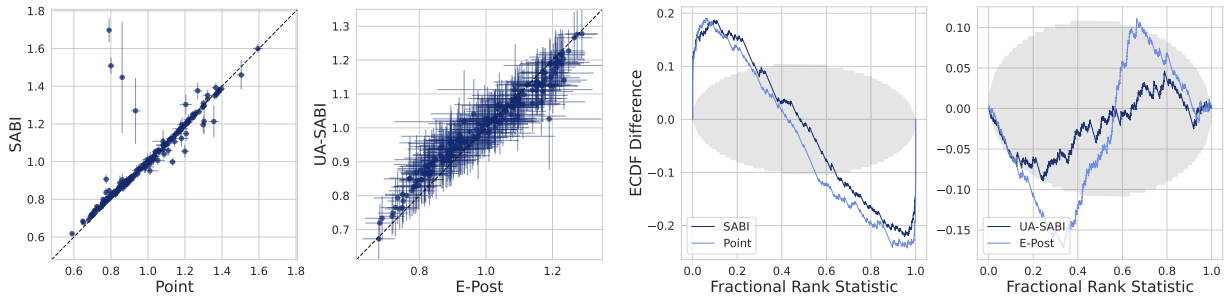
(b) ECDF difference plots. Only full-budget ABI and UA-SABI are well calibrated.

Figure 3: LogSin recovery plots (top) and ECDF difference plots (bottom) for full-budget ABI, low-budget ABI, SABI, and UA-SABI (from left to right) over 200 ground truth samples. In the ECDF difference plots, empirical ranks are shown in blue, 95% confidence bands assuming calibration are shown in grey.

Figure 3a shows that a low-budget ABI model is unable to recover the ground truth values, particularly towards the boundaries of the parameter domain. Using a surrogate as part of SABI or UA-SABI, it becomes possible to generate sufficient ABI training data. Assigning the inter- and extrapolation tasks to the surrogate, specifically designed for this purpose, produces better outcomes than relying on the NPE to implicitly interpolate the posterior from scarce data. However, without propagating surrogate uncertainty (i.e., using SABI), estimated posterior uncertainty remains minimal and does not cover the true values, thus producing overconfident posteriors. Conversely, incorporating surrogate uncertainty via UA-SABI leads to an increase in posterior uncertainty, now better covering the true values. Indeed, the ECDF difference plots showing calibration in Fig. 3b confirm that both low-budget ABI and SABI produce severely miscalibrated posteriors, while UA-SABI and full-budget ABI yield well-calibrated posteriors. However, achieving this calibration with full-budget ABI requires substantially more training data, highlighting the efficiency of UA-SABI under limited simulation budgets.

4.2.3 Validation of Parameter Posterior

We validate the correctness of the surrogate-based ABI approaches and benchmark their performance against a reference solution. To this end, we compare the two surrogate-based ABI methods, SABI and UA-SABI, against the corresponding surrogate-based MCMC methods, Point and E-Post. We plot the posterior medians (circles) and their median deviations (lines) for the ABI methods along with the corresponding MCMC results, as well as the ECDF differences for all the methods in Fig. 4.



(a) Recovery plots.

(b) ECDF difference plots.

Figure 4: LogSin recovery plots (left) and ECDF difference plots (right) comparing ABI methods to corresponding MCMC methods over 200 ground truth samples. In the ECDF difference plots, empirical ranks are shown in blue, 95% confidence bands assuming calibration are shown in grey.

When comparing the ABI posteriors to the MCMC full reference solution, we observe that for all methods, the ABI results align closely with their MCMC counterparts. This validates the reliability and confirms the correctness of the ABI methods. Comparing the ECDF differences between the ABI methods and their corresponding MCMC baselines shows that both Point and SABI produce similarly overconfident posteriors, while UA-SABI yields well-calibrated posteriors. In contrast, E-Post tends to produce even slightly underconfident posteriors.

4.2.4 Runtime Comparison

Next, we justify the computational effort required to train UA-SABI. Specifically, we aim to determine the break-even point – that is, the number of observation sets (i.e., measurement series) after which training an UA-SABI and performing (quasi-instant) inference becomes more efficient than repeatedly rerunning E-Post. Beyond that point, the one-time training cost of UA-SABI is amortized. All experiments were performed on a standard laptop equipped with an Intel Core i7-1185G7 CPU. For ABI, we consider both the upfront training phase and the inference time for the given sets as part of the total runtime. For E-Post, we measure the inference runtime while parallelized across 8 cores. Unlike for UA-SABI, E-Post runtimes depend on the number of cores used for parallelization.

In Fig. 6a, we show the runtime of UA-SABI and E-Post for $\{5, 10, 15, 20\}$ inference runs. We observe that UA-SABI's runtime is nearly constant for the number of inference runs, while E-Post scales linearly. Based on this comparison, UA-SABI is already justified after around 9 inference runs.

4.3 Case Study 2: Carbon Dioxide (CO_2) Storage Model

In the second case study, we test our method on a virtual benchmark that represents a real-world problem. Specifically, we consider a CO_2 storage benchmark (Köppel et al., 2019). In this test case, a non-linear hyperbolic partial differential equation models the two-phase flow of CO_2 in brine. It describes CO_2 injection, plume migration, pressure build-up, and the influence of uncertain porous medium properties in a deep saline aquifer. Given the CO_2 saturation as measurements, our aim is to infer three parameters of interest: injection rate of CO_2 (IR), relative permeability degree in the fractional flux function (PM), and porosity (PR) of the formation. This test case allows us to analyze the performance of UA-SABI in inference for hydrosystem models. The underlying physical model exhibits strong discontinuities at certain combinations

of parameter values and space/time coordinates, which inherently limit the approximation performance of PCE-based surrogate models. However, this scenario provides an opportunity to test UA-SABI under challenging conditions where the surrogate model is strongly misspecified. Five separate Bayesian PCE models with sparsity-inducing priors (Bürkner et al., 2023) are trained for five different instants at days $\{20, 40, 60, 80, 100\}$ and at a fixed location 30m from the injection well. A similar setup was studied in (Oladyshkin et al., 2020). We show parameter posterior results for porosity, the most informative parameter of the CO_2 storage model.

4.3.1 Setup

In general, the setup for the CO_2 storage model follows the same structure as that described in Section 4.2.1. The priors of the input parameters are given as IR $\sim 6.4\times10^{-4}\times(1+\text{Beta}(4,2))$, PM $\sim 2\times\text{Beta}(1.25, 1.25)+2$, and PR $\sim \text{Beta}(2.4, 9)$ (Köppel et al., 2019). For PCE training, we considered 64 Sobol sequence evaluations scaled to the input parameter prior ranges and constructed the polynomial basis with arbitrary polynomial chaos (aPC) based on the priors (Oladyshkin & Nowak, 2012; Bürkner et al., 2023).

In contrast to case study 1, we used an offline ABI training set, where the training data is pre-generated from 10^4 parameter prior draws and fixed during the training process, allowing for direct comparison with a standard full-budget ABI trained on pre-generated simulation data. Further computational details are given in Appendix C.

4.3.2 Validation of Parameter Posterior

We compare the inferred posterior samples obtained from a full-budget ABI ($N_B = 10^4$), a low-budget ABI ($N_B = 64$), SABI, and UA-SABI. Full-budget ABI was feasible due to the already available data (Köppel et al., 2017; Oladyshkin et al., 2020). Therefore, recovery plots were chosen to validate the quality of our results relative to standard ABI (Fig. 5).

Recovery plots in Fig. 5a again show that low-budget ABI struggles to recover the ground truth parameter, particularly near domain boundaries. SABI also produces poor estimates and, moreover, fails to capture uncertainty in the inferred parameter posteriors. This overconfidence results in miscalibrated posteriors, confirmed by the two corresponding ECDF difference plots in Fig. 5b.

In contrast, UA-SABI performs comparably to full-budget ABI while accounting for the additional uncertainty introduced by using a surrogate to generate training data. According to the ECDF difference plots, it produces a well-calibrated posterior for porosity. Additionally, recovery plots between ABI and MCMC for surrogate-based methods and ECDF difference plots for MCMC-based methods are provided in Appendix D. Despite the model's discontinuities and the resulting surrogate misspecification, our method successfully quantifies and propagates surrogate uncertainty to the parameter posterior.

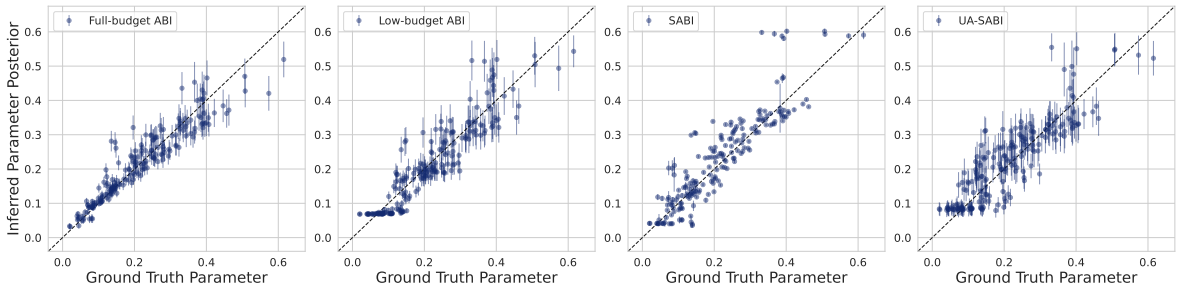
4.3.3 Runtime Comparison

Also, for the CO_2 storage model, we compare the runtimes, following the same approach as in Section 4.2.4. The experiments were run on a computing cluster with two AMD EPYC 7551 CPUs (totaling 64 physical cores) to speed up E-Post through parallelization.

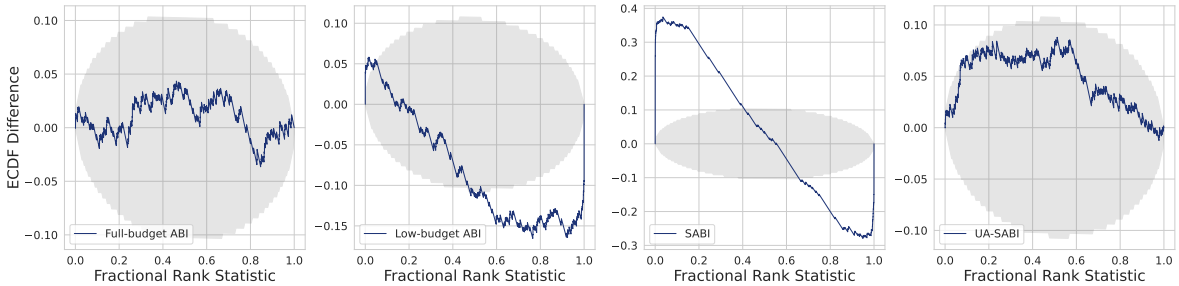
Figure 6b presents the measured runtimes of UA-SABI and E-Post for $\{5, 6, 7, 8\}$ inference runs, whereby E-Post was parallelized on 16 cores. We observe a break-even point between 6 and 7 inference runs, indicating that UA-SABI becomes justified after 7 runs. Despite using more cores for E-Post, the training costs are amortized earlier for a more expensive model.

4.4 Case Study 3: Microbially Induced Calcite Precipitation Model

In this case study, we transition from benchmarks to realistic conditions by using real measurement data. Case studies 1 and 2 introduced a toy example and a virtual benchmark, respectively. Case study 3 now focuses on posterior inference from observed measurements, where the primary goal is to evaluate the appli-

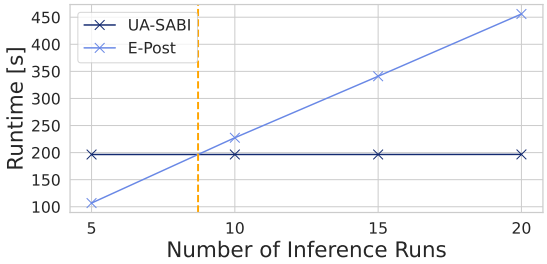


(a) Recovery plots of porosity.

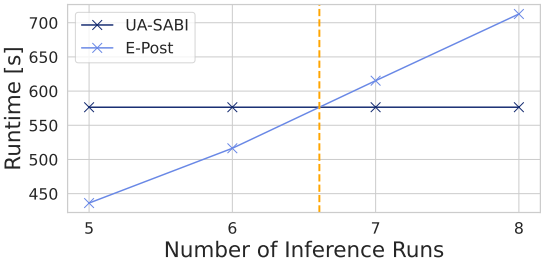


(b) ECDF difference plots of porosity. Only full-budget ABI and UA-SABI are well calibrated.

Figure 5: CO_2 recovery plots (top) and ECDF difference plots (bottom) for full-budget ABI, low-budget ABI, SABI, and UA-SABI (from left to right) over 200 ground truth samples. In the ECDF difference plots, empirical ranks are shown in blue, 95% confidence bands assuming calibration are shown in grey.



(a) Runtimes for LogSin for $\{5, 10, 15, 20\}$ runs.



(b) Runtimes for CO_2 for $\{5, 6, 7, 8\}$ runs.

Figure 6: Comparison of runtimes for UA-SABI (training and inference) and E-Post (inference) with break-even point.

cability of UA-SABI under real-world conditions. A runtime comparison for the MICP model on synthetic data is added in Appendix E.4.

The underlying model predicts microbially induced calcite precipitation (MICP). MICP is a reactive transport process, here taking place inside a porous medium. It includes at least biofilm, calcite, and the unreactive solid matrix as solid phases, water, and, in some cases, an additional fluid phase such as gas. Relevant in the aqueous phase are dissolved calcium, inorganic carbon, and urea. The complete set of relevant species is detailed in Hommel et al. (2015). The process is mediated by S. pasteurii, a bacterium that produces the enzyme urease. Urease catalyzes the hydrolysis of urea into ammonia and carbonic acid. In aqueous solution, ammonia consumes hydrogen ions (H^+), which raises the pH. The increase in pH alters carbonate equilibria: carbonic acid dissociates further, releasing additional H^+ and carbonate ions. This raises the concentration of dissolved carbonate species. When calcium ions are present, they combine with carbonate ions and trigger the precipitation of calcite within the pore space.

The experiment is performed on a 61cm sand-filled column with a 2.54cm diameter. Bacteria are first injected from the bottom, followed by an overnight no-flow period to allow biofilm formation. Biofilm growth is then stimulated with a 24h substrate injection. Afterwards, two pore volumes of a 0.33mol/L calcium–urea solution are injected at 10ml/min, repeated every 24h. Each injection is followed by a no-flow period for mineralization, and then a substrate injection to reactivate the biofilm (Hommel et al., 2015). This cycle is repeated 30 times. A more detailed description can be found in Hommel et al. (2015).

Measurements of the volume fraction of calcite on eight spatial points located at 3.81, 11.43, 19.05, 26.67, 34.29, 41.91, 49.53, and 57.15cm distance from the bottom are available. After the last cycle, we build our surrogate based on four parameters of interest: the coefficient for preferential attachment to biomass ($c_{a,1}, [\mathrm{s}^{-1}]$), the coefficient for attachment to arbitrary surfaces ($c_{a,2}, [\mathrm{s}^{-1}]$), the dry mass density of biofilm ($\rho_f, [\mathrm{kg/m}^3]$), and the enzyme content of biomass ($k_{ub}, [\mathrm{kg/kg}]$).

The MICP case provides an opportunity to test UA-SABI on a model that is highly computationally expensive yet low-dimensional in parameter space, and for which real-world measurement data are available. We again employ PCE as the surrogate modeling technique, due to its demonstrated success for this model (Scheurer et al., 2021). In total, eight Bayesian PCEs are trained, one for each measurement location.

4.4.1 Setup

In general, the setup for the MICP model follows the same structure as that described in the former case studies. The priors of the input parameters are given as $c_{a,1} \sim \mathcal{U}(10^{-10}, 10^{-7})$, $c_{a,2} \sim \mathcal{U}(10^{-10}, 10^{-6})$, $\rho_f \sim \mathcal{U}(1, 15)$, and $k_{ub} \sim \mathcal{U}(10^{-5}, 5 \cdot 10^{-4})$ (Scheurer et al., 2021). For PCE training, 25 model evaluations within the prior ranges of input parameters are available, which are not necessarily optimal for surrogate training. We constructed the polynomial basis with aPC based on the available model evaluations (Oladyshkin & Nowak, 2012; Bürkner et al., 2023).

As in case study 2, we used an offline ABI training set of 10^4 parameter prior draws. In contrast to case study 2, no pre-generated simulation data are available for these draws. Instead, output data can only be generated through the surrogate; thus, no comparison with a full-budget ABI is possible. Further computational details are given in Appendix C.

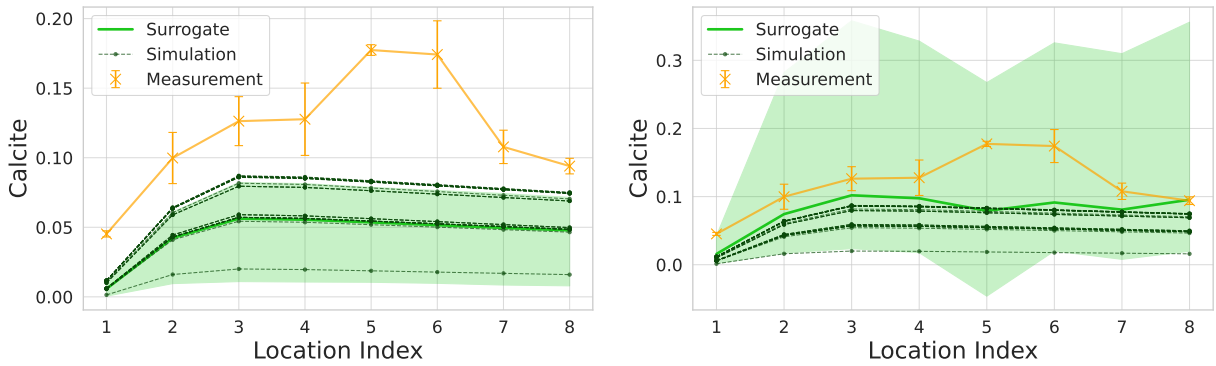
4.4.2 Validation of Surrogate Model and Parameter Inference Setup

Since inference will be performed using real measurement data, the first step is to evaluate how accurately the surrogate represents the underlying system. Figure 7a presents the measurement data, the model evaluations used for surrogate training, and the surrogate predictions obtained from 10^4 prior samples across the eight locations. The results indicate that the simulated model data—and consequently the surrogate—do not cover the region occupied by the measurement data. This makes the measurement data out-of-distribution (OOD) relative to the simulation model's a priori predictions - a phenomenon that is suspected to occur frequently in dynamic systems modeling, although rarely investigated in detail.

Neural networks are known to generalize poorly outside their training domain, resulting in poor posteriors obtained via ABI (Schmitt et al., 2024a), which is shown in Appendix E.1. Thus, we seek to transfer this challenge to the surrogate instead. To this end, we found that the surrogate priors used for parameter inference can be adjusted to ensure that the measurement data fall within the surrogate's support. This approach leverages the generalization properties of the low-dimensional, regularized surrogate, avoiding dependence on the poor generalization abilities of neural networks, which are used within the ABI model during inference.

The results of a sensitivity analysis (more detailed in Appendix E.2) on the surrogate using Sobol indices (Sobol', 1990) shows that the surrogate's output at all locations is almost only sensitive to k_{ub}.

In Fig. 8, the surrogate output is shown exemplarily at location index 3 by varying k_{ub}. It can be observed that the uncertainty of the surrogate increases when evaluating OOD inputs. Based on these results, we adapt the parameter inference prior range for k_{ub}, such that $k_{ub} \sim \mathcal{U}(10^{-5}, 5 \cdot 10^{-3})$. This adjustment results in a prior predictive that covers the measurement data, as illustrated in Fig. 7b.



(a) Surrogate output with original prior for $k_{ub} \sim \mathcal{U}(10^{-5}, 5 \cdot 10^{-4})$.

(b) Surrogate output with adjusted prior for $k_{ub} \sim \mathcal{U}(10^{-5}, 5 \cdot 10^{-3})$.

Figure 7: Measurement data for calcite, 25 model evaluations, and surrogate model output obtained with original prior (left) and adjusted prior (right) for k_{ub} of the MICP model across eight locations. Surrogate predictions are obtained from 10^4 prior samples; median and 90% confidence intervals are shown.

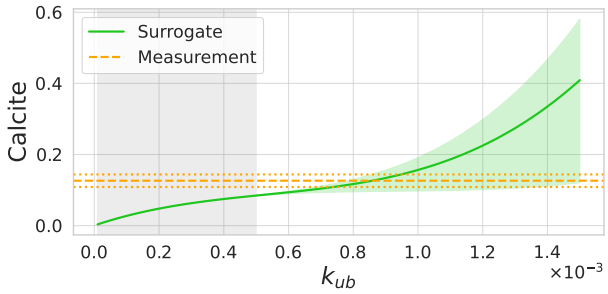


Figure 8: MICP surrogate output at location index 3 over k_{ub}. The output is obtained by evaluating the uncertainty-aware surrogate for $k_{ub} \in [10^{-5}, 5 \cdot 10^{-3}]$ and keeping the other parameters fixed at their means. Median and 90% confidence intervals of the surrogate output are shown in green, the region of the original prior is shown in grey.

4.4.3 Comparison of Parameter Posterior

Since parameter inference is carried out on real measurement data, no ground truth is available. We therefore compare only the inferred posteriors obtained using low-budget ABI ($N_B = 25$), SABI, UA-SABI, Point, and E-Post. For easy visual comparison, we applied kernel density estimation (KDE) with Scott's bandwidth method to the posterior samples from each approach. Since k_{ub} is the only sensitive parameter, we present its posterior estimates in Fig. 9 to avoid redundancy, while posterior estimates for the remaining parameters are provided in Appendix E.3.

Figure 9 demonstrates that low-budget ABI, which is shown as a point estimate due to quasi-nonexistent variance in the posterior, exhibits a pronounced bias and substantially underestimates posterior uncertainty. This confirms again that training an ABI model with very limited data is not meaningful. Comparing the surrogate-based methods further highlights that explicitly quantifying surrogate uncertainty and propagating it through the inference process is reflected in the wider posterior estimates. To assess the validity of the obtained posteriors and the appropriateness of the quantified and propagated uncertainty, Fig. 10 presents the posterior predictive distributions for calcite across the four surrogate-based methods.

Figure 10a and Fig. 10b clearly illustrate that not quantifying and propagating uncertainty results in overconfident posterior predictives. In contrast, as shown in Fig. 10c and Fig. 10d, explicitly accounting for uncertainty produces confidence intervals that successfully cover the measurement data. Both methods yield

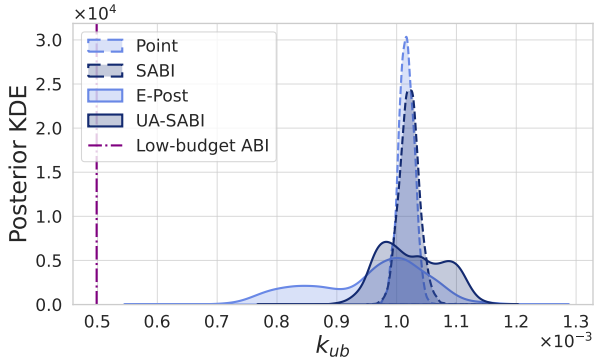
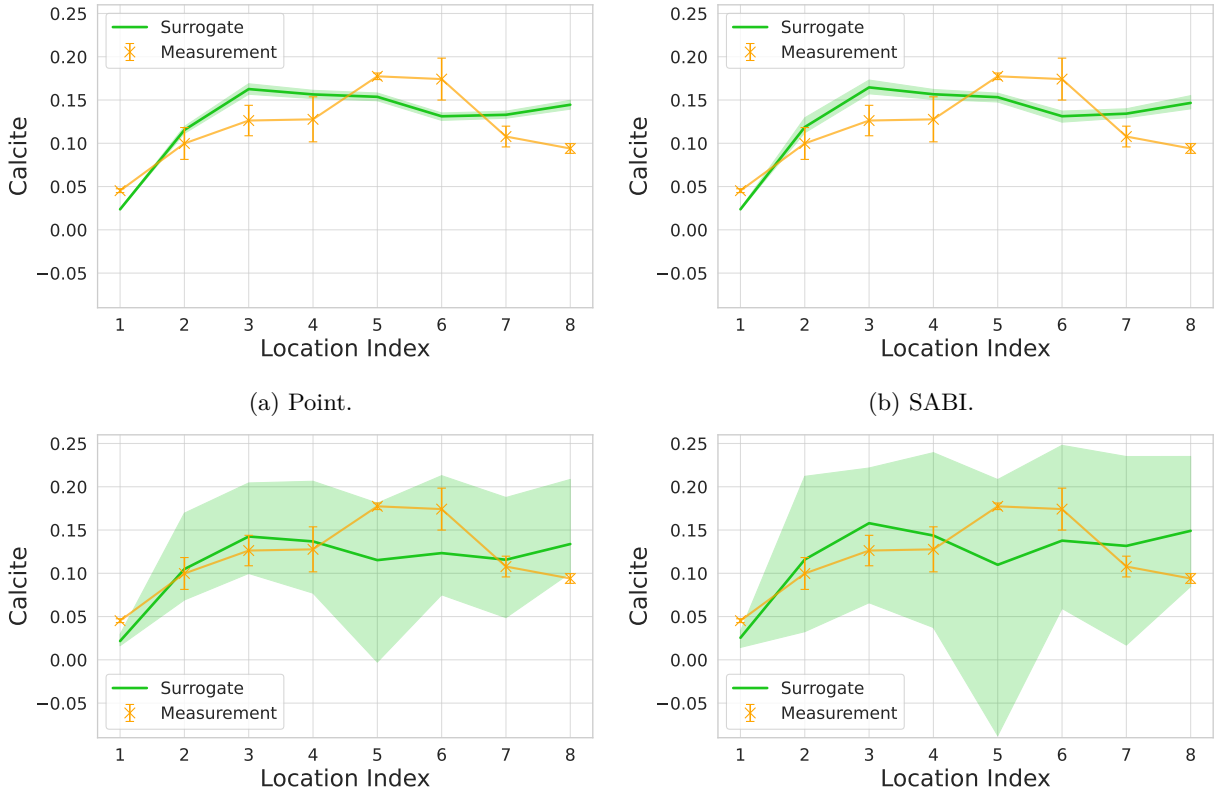


Figure 9: KDEs of 4,000 parameter posterior samples for k_{ub} of the MICP model obtained with low-budget ABI, Point, E-Post, SABI, and UA-SABI.



(a) Point.

(b) SABI.

(c) E-Post.

(d) UA-SABI.

Figure 10: MICP posterior predictives for calcite for the four surrogate-based methods. Surrogate predictions are obtained from 4,000 posterior samples, median and 90% confidence intervals are shown.

comparable intervals, with UA-SABI producing a slightly broader interval as a higher posterior density in the out-of-distribution region of the most sensitive parameter k_{ub} (Fig. 9) yields increased surrogate predictive uncertainty.

Overall, this demonstrates that quantifying and propagating the uncertainty of a surrogate model enables ABI for highly computationally expensive models. Even when the available model simulations are subop-

timal for surrogate training and far from the measurement data, accounting for uncertainty and promoting generalization in the surrogate model leads to more trustworthy results.

5 Summary and Outlook

In this work, we introduced Uncertainty-Aware Surrogate-based Amortized Bayesian Inference (UA-SABI) – a framework designed to enable efficient and reliable ABI for computationally expensive models. UA-SABI combines surrogate modeling and ABI while explicitly quantifying and propagating surrogate uncertainties through the inference process. This addresses a core limitation of existing approaches: while surrogate models can reduce the cost of generating training data for ABI, ignoring their approximation error often leads to overconfident and misleading posteriors. By incorporating uncertainty awareness, UA-SABI enables better-calibrated and reliable inference, even under tight computational constraints.

We validated UA-SABI in both a simple toy example and two real-world problems of modeling CO_2 storage and MICP, highlighting its ability to produce well-calibrated posterior estimates that match those obtained with MCMC-based methods. Our experiments demonstrate the importance of sufficient training data, as low-budget ABI produces erroneous posteriors. Also, they show the importance of uncertainty propagation when using a surrogate: Surrogate-based ABI results in overconfident posteriors, whereas UA-SABI correctly reflects model uncertainty in its predictions. Moreover, we showed that the upfront computational effort of training UA-SABI is quickly offset in scenarios involving repeated inference, with amortization becoming beneficial already after a few inference runs. Additionally, the MICP experiment poses a challenging real-world scenario by having 1) only a limited number of model runs available, and 2) real measurement data that are not adequately captured by these model runs.

Overall, UA-SABI offers an efficient and reliable solution in settings that require repeated inference for computationally expensive models. In general, we conclude that using surrogates to generate training data can be an effective strategy for ABI when simulations are scarce or computationally expensive. Moreover, explicitly accounting for surrogate uncertainty in the training data improves ABI results, particularly by reducing overconfidence.

In this work, we applied UA-SABI only to low-dimensional, computationally expensive problems to keep results interpretable and avoid challenges that would distract from our scope - such challenges as they appear in surrogate training and parameter inference for high-dimensional models with only scarce available data. Future work should test the method on higher-dimensional parameter spaces to further evaluate its scalability and performance. Additionally, the surrogate is technically misspecified, as the simulator is not fully contained within the surrogate class. Thus, future work could integrate UA-SABI with recently proposed methods for detecting and mitigating model misspecification in ABI (Schmitt et al., 2024a; Dellaporta et al., 2022; Elsemüller et al., 2025), which are all naturally compatible with our framework.

References

Lynton Ardizzone, Carsten Lüth, Jakob Kruse, Carsten Rother, and Ullrich Köthe. Guided Image Generation with Conditional Invertible Neural Networks, 2019. URL https://arxiv.org/abs/1907.02392.

Ayush Bharti, Daolang Huang, Samuel Kaski, and François-Xavier Briol. Cost-aware Simulation-based Inference, 2025. URL https://arxiv.org/abs/2410.07930.

Paul-Christian Bürkner, Ilja Kröker, Sergey Oladyshkin, and Wolfgang Nowak. A Fully Bayesian Sparse Polynomial Chaos Expansion Approach with Joint Priors on the Coefficients and Global Selection of Terms. *Journal of Computational Physics*, 488:112210, 2023. doi: 10.1016/j.jcp.2023.112210.

Bob Carpenter, Andrew Gelman, Matthew D. Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A Probabilistic Programming Language. *Journal of Statistical Software*, 76(1):1—-32, 2017. doi: 10.18637/jss.v076.i01.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The Frontier of Simulation-based Inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020. doi: 10.1073/pnas.1912789117.

Michael Deistler, Pedro J. Goncalves, and Jakob H. Macke. Truncated Proposals for Scalable and Hassle-free Simulation-based Inference. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 23135–23149. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/9278abf072b58caf21d48dd670b4c721-Paper-Conference.pdf.

Arnaud Delaunoy, Maxence de la Brassinne Bonardeaux, Siddharth Mishra-Sharma, and Gilles Louppe. Low-Budget Simulation-based Inference with Bayesian Neural Networks, 2024. URL https://arxiv.org/abs/2408.15136.

Charita Dellaporta, Jeremias Knoblauch, Theodoros Damoulas, and François-Xavier Briol. Robust Bayesian Inference for Simulator-based Models via the MMD Posterior Bootstrap. In G. Camps-Valls, F. J. R. Ruiz, and I. Valera (eds.), *International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 943–970. PMLR, 2022. URL https://proceedings.mlr.press/v151/dellaporta22a.html.

Alexander Denker, Maximilian Schmidt, Johannes Leuschner, and Peter Maass. Conditional Invertible Neural Networks for Medical Imaging. *Journal of Imaging*, 7(11):243, 2021. doi: 10.3390/jimaging7110243.

Felix Draxler, Stefan Wahl, Christoph Schnörr, and Ullrich Köthe. On the Universality of Volume-Preserving and Coupling-Based Normalizing Flows, 2025. URL https://arxiv.org/abs/2402.06578.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural Spline Flows. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/7ac71d433f282034e088473244df8c02-Paper.pdf.

Aaron M. Ellison. Bayesian Inference in Ecology. *Ecology Letters*, 7(6):509–520, 2004. doi: 10.1111/j.1461-0248.2004.00603.x.

Lasse Elsemüller, Valentin Pratz, Mischa von Krause, Andreas Voss, Paul-Christian Bürkner, and Stefan T. Radev. Does Unsupervised Domain Adaptation Improve the Robustness of Amortized Bayesian Inference? A Systematic Evaluation, 2025. URL https://arxiv.org/abs/2502.04949.

Alexander Etz and Joachim Vandekerckhove. Introduction to Bayesian Inference for Psychology. *Psychonomic Bulletin & Review*, 25:5–34, 2018. doi: 10.3758/s13423-017-1262-3.

David T. Frazier, Ryan P. Kelly, Christopher C. Drovandi, and David J. Warne. The Statistical Accuracy of Neural Posterior and Likelihood Estimation, 2024. URL https://arxiv.org/abs/2411.12068.

Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald B. Rubin. *Bayesian Data Analysis*. Chapman & Hall/CRC, 1995. ISBN 9780429258411.

Walter R. Gilks, Sylvia Richardson, and David Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC, 1995. ISBN 9780429170232.

Manuel Gloeckler, Shoji Toyota, Kenji Fukumizu, and Jakob H. Macke. Compositional Simulation-based Inference for Time Series, 2025. URL https://arxiv.org/abs/2411.02728.

David Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic Posterior Transformation for Likelihood-Free Inference. In K. Chaudhuri and R. Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2404–2414. PMLR, 2019a. URL https://proceedings.mlr.press/v97/greenberg19a.html.

David S. Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic Posterior Transformation for Likelihood-free Inference. In K. Chaudhuri and R. Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2404–2414. PMLR, 2019b. URL https://proceedings.mlr.press/v97/greenberg19a.html.

Michael U. Gutmann and Jukka Corander. Bayesian Optimization for Likelihood-Free Inference of Simulator-Based Statistical Models. *Journal of Machine Learning Research*, 17(125):1–47, 2016. URL http://jmlr.org/papers/v17/15-017.html.

Matthew D. Hoffman and Andrew Gelman. The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(47):1593–1623, 2014. URL https://www.jmlr.org/papers/volume15/hoffman14a/hoffman14a.pdf.

Johannes Hommel, Ellen Lauchnor, Adrienne Phillips, Robin Gerlach, Alfred B Cunningham, Rainer Helmig, Anozie Ebigbo, and Holger Class. A Revised Model for Microbially Induced Calcite Precipitation: Improvements and New Insights Based on Recent Experiments. *Water Resources Research*, 51(5):3695–3715, 2015. doi: 10.1002/2014WR016503.

John P. Hülsenbeck, Fredrik Ronquist, Rasmus Nielsen, and Jonathan P. Bollback. Bayesian Inference of Phylogeny and its Impact on Evolutionary Biology. *Science*, 294(5550):2310–2314, 2001. doi: 10.1126/science.1065889.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Markus Köppel, Fabian Franzelin, Ilja Kröker, Sergey Oladyshkin, Dominik Wittwar, Gabriele Santin, Andrea Barth, Bernard Haasdonk, Wolfang Nowak, Dirk Pflüger, and Christian Rohde. Datasets and executables of data-driven uncertainty quantification benchmark in carbon dioxide storage, 2017.

Markus Köppel, Fabian Franzelin, Ilja Kröker, Sergey Oladyshkin, Gabriele Santin, Dominik Wittwar, Andrea Barth, Bernard Haasdonk, Wolfgang Nowak, Dirk Pflüger, and Christian Rohde. Comparison of Data-Driven Uncertainty Quantification Methods for a Carbon Dioxide Storage Benchmark Scenario. *Computational Geosciences*, 23:339–354, 2019. doi: 10.1007/s10596-018-9785-x.

Anastasia N. Krouglova, Hayden R. Johnson, Basile Confavreux, Michael Deistler, and Pedro J. Gonçalves. Multifidelity Simulation-based Inference for Computationally Expensive Simulators, 2025. URL https://arxiv.org/abs/2502.08416.

Sourabh Kulkarni and Csaba Andras Moritz. Improving Effectiveness of Simulation-based Inference in the Massively Parallel Regime. *IEEE Transactions on Parallel and Distributed Systems*, 34(4):1100–1114, 2023. doi: 10.1109/TPDS.2023.3238045.

Loïc Le Gratiet, Stefano Marelli, and Bruno Sudret. *Metamodel-based Sensitivity Analysis: Polynomial Chaos Expansions and Gaussian Processes*, pp. 1289–1325. Springer, 2017. ISBN 978-3-319-12385-1. doi: 10.1007/978-3-319-12385-1_3.

Jan-Matthis Lückmann, Giacomo Bassetto, Theofanis Karaletsos, and Jakob H. Macke. Likelihood-free Inference with Emulator Networks. In F. J. R. Ruiz, C. Zhang, D. Liang, and T. D. Bui (eds.), *Proceedings of The 1st Symposium on Advances in Approximate Bayesian Inference*, volume 96 of *Proceedings of Machine Learning Research*, pp. 32–53. PMLR, 2018. URL https://proceedings.mlr.press/v96/lueckmann19a.html.

Jan-Matthis Lückmann, Jan Bölts, David S. Greenberg, Pedro Goncalves, and Jakob Macke. Benchmarking Simulation-based Inference. In A. Banerjee and K. Fukumizu (eds.), *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 343–351. PMLR, 2021. URL https://proceedings.mlr.press/v130/lueckmann21a.html.

Jan-Matthis Lueckmann, Pedro J. Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H. Macke. Flexible Statistical Inference for Mechanistic Models of Neural Dynamics. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/addfa9b7e234254d26e9c7f2af1005cb-Paper.pdf.

Nicolò Malagutti, Grace McGinness, and Dilip A. Nithyanandam. Real-Time Personalised Pharmacokinetic-Pharmacodynamic Modelling in Propofol Anesthesia Through Bayesian Inference. In *2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 1–6, 2023. doi: 10.1109/EMBC40787.2023.10339991.

Norman Marlier. *Simulation-based Inference for Robotic Grasping.* PhD thesis, Universite de Liege (Belgium), 2024.

Edward Meeds and Max Welling. GPS-ABC: Gaussian Process Surrogate Approximate Bayesian Computation, 2014. URL https://arxiv.org/abs/1401.2838.

Martin Modrák, Angie H. Moon, Shinyoung Kim, Paul-Christian Bürkner, Niko Huurre, Kateřina Faltejsková, Andrew Gelman, and Aki Vehtari. Simulation-based Calibration Checking for Bayesian Computation: The Choice of Test Quantities Shapes Sensitivity. *Bayesian Analysis*, pp. 1–28, 2023. doi: 10.1214/23-BA1404.

Farid Mohammadi, Rebekka Kopmann, Anneli Guthke, Sergey Oladyshkin, and Wolfgang Nowak. Bayesian Selection of Hydro-morphodynamic Models under Computational Time Constraints. *Advances in Water Resources*, 117:53–64, 2018. doi: 10.1016/j.advwatres.2018.05.007.

Sergey Oladyshkin and Wolfgang Nowak. Data-Driven Uncertainty Quantification Using the Arbitrary Polynomial Chaos Expansion. *Reliability Engineering & System Safety*, 106:179–190, 2012. doi: 10.1016/j.ress.2012.05.002.

Sergey Oladyshkin, Farid Mohammadi, Ilja Kröker, and Wolfgang Nowak. Bayesian^3 Active Learning for the Gaussian Process Emulator using Information Theory. *Entropy*, 22(8):890, 2020. doi: 10.3390/e22080890.

Govinda A. Padmanabha and Nicholas Zabaras. Solving Inverse Problems Using Conditional Invertible Neural Networks. *Journal of Computational Physics*, 433:110194, 2021. doi: 10.1016/j.jcp.2021.110194.

George Papamakarios and Iain Murray. Fast ϵ -free Inference of Simulation Models with Bayesian Conditional Density Estimation. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/file/6aca97005c68f1206823815f66102863-Paper.pdf.

George Papamakarios, David Sterratt, and Iain Murray. Sequential Neural Likelihood: Fast Likelihood-free Inference with Autoregressive Flows. In Kamalika Chaudhuri and Masashi Sugiyama (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 837–848. PMLR, 2019. URL https://proceedings.mlr.press/v89/papamakarios19a.html.

George Papamakarios, Eric Nalisnick, Danilo J Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing Flows for Probabilistic Modeling and Inference. *Journal of Machine Learning Research*, 22 (57):1–64, 2021. URL http://jmlr.org/papers/v22/19-1028.html.

Thomas P. Prescott and Ruth E. Baker. Multifidelity Approximate Bayesian Computation. *SIAM/ASA Journal on Uncertainty Quantification*, 8(1):114–138, 2020. doi: 10.1137/18M1229742.

Thomas P. Prescott and Ruth E. Baker. Multifidelity approximate Bayesian computation with sequential Monte Carlo parameter sampling. *SIAM/ASA Journal on Uncertainty Quantification*, 9(2):788–817, 2021. doi: 10.1137/20M1316160.

Thomas P. Prescott, David J. Warne, and Ruth E. Baker. Efficient Multifidelity Likelihood-free Bayesian Inference with Adaptive Computational Resource Allocation. *Journal of Computational Physics*, 496: 112577, 2024. doi: 10.1016/j.jcp.2023.112577.

Leah F Price, Christopher C Drovandi, Anthony Lee, and David J Nott. Bayesian Synthetic Likelihood. *Journal of Computational and Graphical Statistics*, 27(1):1–11, 2018. doi: 10.1080/10618600.2017.1302882.

Stefan T. Radev, Ulf K. Mertens, Andreas Voss, Lynton Ardizzone, and Ullrich Köthe. BayesFlow: Learning Complex Stochastic Models with Invertible Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, 33(4):1452–1466, 2020. doi: 10.1109/TNNLS.2020.3042395.

Stefan T. Radev, Frederik Graw, Simiao Chen, Nico T. Mutters, Vanessa M. Eichel, Till Bärnighausen, and Ullrich Köthe. OutbreakFlow: Model-based Bayesian Inference of Disease Outbreak Dynamics with Invertible Neural Networks and its Application to the COVID-19 Pandemics in Germany. *PLoS Computational Biology*, 17(10):e1009472, 2021. doi: 10.1371/journal.pcbi.1009472.

Stefan T. Radev, Marvin Schmitt, Valentin Pratz, Umberto Picchini, Ullrich Köthe, and Paul-Christian Bürkner. JANA: Jointly Amortized Neural Approximation of Complex Bayesian Models. In R. J. Evans and I. Shpitser (eds.), *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pp. 1695–1706. PMLR, 2023a. URL https://proceedings.mlr.press/v216/radev23a.html.

Stefan T. Radev, Marvin Schmitt, Lukas Schumacher, Lasse Elsemüller, Valentin Pratz, Yannik Schälte, Ullrich Köthe, and Paul-Christian Bürkner. BayesFlow: Amortized Bayesian Workflows with Neural Networks. *Journal of Open Source Software*, 8(89):5702, 2023b. doi: 10.21105/joss.05702.

Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2006. ISBN 026218253X.

Philipp Reiser, Javier E. Aguilar, Anneli Guthke, and Paul-Christian Bürkner. Uncertainty Quantification and Propagation in Surrogate-based Bayesian Inference. *Statistics and Computing*, 35(3):66, 2025. doi: 10.1007/s11222-025-10597-8.

Christian P. Robert, Víctor Elvira, Nick Tawn, and Changye Wu. Accelerating MCMC Algorithms. *Wiley Interdisciplinary Reviews: Computational Statistics*, 10(5):e1435, 2018. doi: 10.1002/wics.1435.

Teemu Säilynoja, Paul-Christian Bürkner, and Aki Vehtari. Graphical Test for Discrete Uniformity and Its Applications in Goodness-of-Fit Evaluation and Multiple Sample Comparison. *Statistics and Computing*, 32(32), 2022. doi: 10.1007/s11222-022-10090-6.

Stefania Scheurer, Aline Schäfer Rodrigues Silva, Farid Mohammadi, Johannes Hommel, Sergey Oladyshkin, Bernd Flemisch, and Wolfgang Nowak. Surrogate-based Bayesian Comparison of Computationally Expensive Models: Application to Microbially Induced Calcite Precipitation. *Computational Geosciences*, 25(6): 1899–1917, 2021. doi: 10.1007/s10596-021-10076-9.

Marvin Schmitt, Paul-Christian Bürkner, Ullrich Köthe, and Stefan T. Radev. Detecting Model Misspecification in Amortized Bayesian Inference with Neural Networks. In U. Köthe and C. Rother (eds.), *Pattern Recognition*, pp. 541–557. Springer Nature Switzerland, 2024a. doi: 10.1007/978-3-031-54605-1_35.

Marvin Schmitt, Valentin Pratz, Ullrich Köthe, Paul-Christian Bürkner, and Stefan T. Radev. Consistency Models for Scalable and Fast Simulation-based Inference. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 126908–126945. Curran Associates, Inc., 2024b. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/e58026e2b2929108e1bd24cbfa1c8e4b-Paper-Conference.pdf.

Qian Shao, Anis Younes, Marwan Fahs, and Thierry A. Mara. Bayesian Sparse Polynomial Chaos Expansion for Global Sensitivity Analysis. *Computer Methods in Applied Mechanics and Engineering*, 318:474–496, 2017. doi: 10.1016/j.cma.2017.01.033.

Ilya M. Sobol'. On the Distribution of Points in a Cube and the Approximate Evaluation of Integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4):86–112, 1967. doi: 10.1016/0041-5553(67)90144-9.

Il'ya Meerovich Sobol'. On Sensitivity Estimation for Nonlinear Mathematical Models. *Matematicheskoe modelirovanie*, 2(1):112–118, 1990.

Stan Development Team. Stan Modeling Language Users Guide and Reference Manual, 2024. URL http://mc-stan.org/. Version 2.36.

Bruno Sudret. Global Sensitivity Analysis Using Polynomial Chaos Expansions. *Reliability Engineering & System Safety*, 93(7):964–979, 2008. doi: 10.1016/j.ress.2007.04.002.

Sean Talts, Michael Betancourt, Daniel Simpson, Aki Vehtari, and Andrew Gelman. Validating Bayesian Inference Algorithms with Simulation-based Calibration, 2020. URL https://arxiv.org/abs/1804.06788.

Shahin Tasoujian, Saeed Salavati, Matthew A. Franchek, and Karolos M. Grigoriadis. Robust Delay-Dependent LPV Synthesis for Blood Pressure Control with Real-Time Bayesian Parameter Estimation. *IET Control Theory & Applications*, 14(10):1334–1345, 2020. doi: 10.1049/iet-cta.2019.0651.

Tina Toni, David Welch, Natalja Strelkowa, Andreas Ipsen, and Michael P.H. Stumpf. Approximate Bayesian Computation Scheme for Parameter Inference and Model Selection in Dynamical Systems. *Journal of the Royal Society Interface*, 6(31):187–202, 2009. doi: 10.1098/rsif.2008.0172.

Udo Von Toussaint. Bayesian Inference in Physics. *Reviews of Modern Physics*, 83(3):943–999, 2011. doi: 10.1103/RevModPhys.83.943.

David J. Warne, Ruth E. Baker, and Matthew J. Simpson. Multilevel Rejection Sampling for Approximate Bayesian Computation. *Computational Statistics & Data Analysis*, 124:71–86, 2018. doi: 10.1016/j.csda.2018.02.009.

David J. Warne, Thomas P. Prescott, Ruth E. Baker, and Matthew J. Simpson. Multifidelity Multilevel Monte Carlo to Accelerate Approximate Bayesian Parameter Inference for Partially Observed Stochastic Processes. *Journal of Computational Physics*, 469:111543, 2022. doi: 10.1016/j.jcp.2022.111543.

Norbert Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60(4):897–936, 1938. doi: 10.2307/2371268.

Jonas Wildberger, Maximilian Dax, Simon Buchholz, Stephen Green, Jakob H. Macke, and Bernhard Schölkopf. Flow Matching for Scalable Simulation-based Inference. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 16837–16864. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/3663ae53ec078860bb0b9c6606e092a0-Paper-Conference.pdf.

Richard Wilkinson. Accelerating ABC Methods using Gaussian Processes. In S. Kaski and J. Corander (eds.), *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, pp. 1015–1023. PMLR, 2014. URL https://proceedings.mlr.press/v33/wilkinson14.html.

Wan Yang, Marc Lipsitch, and Jeffrey Shaman. Inference of Seasonal and Pandemic Influenza Transmission Dynamics. *Proceedings of the National Academy of Sciences*, 112(9):2723–2728, 2015. doi: 10.1073/pnas.1415012112.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R. Salakhutdinov, and Alexander J. Smola. Deep Sets. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/f22e4747da1aa27e363d86d40ff442fe-Paper.pdf.

Jiangjiang Zhang, Qiang Zheng, Dingjiang Chen, Laosheng Wu, and Lingzao Zeng. Surrogate-based Bayesian Inverse Modeling of the Hydrological System: An Adaptive Approach Considering Surrogate Approximation Error. *Water Resources Research*, 56(1):e2019WR025721, 2020. doi: 10.1029/2019WR025721.

A Pseudocode

Algorithm 1 Training of Surrogate and UA-SABI

Surrogate Training Phase

Require: Simulation model $M(\mathbf{x}, \boldsymbol{\omega})$, prior $p(\mathbf{c}, \sigma)$, likelihood $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\omega}, \mathbf{c}, \sigma)$

Ensure: Posterior $p(\mathbf{c}, \sigma \mid D_T)$

Choose inputs and parameters $\{(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)})\}_{i=1}^{N_T}$
for $i = 1, \ldots, N_T$ **do**
    Evaluate simulation model $\mathbf{y}^{(i)} = M(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)})$
end for
Construct training dataset $D_T = \{(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}, \mathbf{y}^{(i)})\}_{i=1}^{N_T}$
Perform Bayesian inference for surrogate parameters

$$p(\mathbf{c}, \sigma \mid D_T) \propto \prod_{i=1}^{N_T} p(\mathbf{y}^{(i)} \mid \mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}, \mathbf{c}, \sigma)\, p(\mathbf{c}, \sigma)$$

UA-SABI Training Phase

Require: Prior $p(\mathbf{x}, \boldsymbol{\omega})$, posterior $p(\mathbf{c}, \sigma \mid D_T)$

Ensure: Trained summary and inference network parameters $\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\varphi}}$

for each epoch **do**
    for $i = 1, \ldots, N_B$ **do**
        Sample inputs and parameters from prior $(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)}) \sim p(\mathbf{x}, \boldsymbol{\omega})$
        Sample surrogate parameters from posterior $(\mathbf{c}^{(i)}, \sigma^{(i)}) \sim p(\mathbf{c}, \sigma \mid D_T)$
        Evaluate surrogate $\widetilde{\mathbf{y}}^{(i)} = \widetilde{M}_{\mathbf{c}^{(i)}}(\mathbf{x}^{(i)}, \boldsymbol{\omega}^{(i)})$
        Sample corrected surrogate output $\widetilde{\mathbf{y}}_{\epsilon}^{(i)} \sim p(\widetilde{\mathbf{y}}_{\epsilon} \mid \widetilde{\mathbf{y}}^{(i)}, \sigma^{(i)})$
        Pass $(\mathbf{x}^{(i)}, \widetilde{\mathbf{y}}_{\epsilon}^{(i)})$ through summary network $\mathbf{s}^{(i)} = S_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}, \widetilde{\mathbf{y}}_{\epsilon}^{(i)})$
        Pass $(\boldsymbol{\omega}^{(i)}, \mathbf{s}^{(i)})$ through inference network $q_{\boldsymbol{\varphi}}(\boldsymbol{\omega}^{(i)} \mid \mathbf{s}^{(i)})$
        Compute loss from Eq. (2) and update network parameters $\boldsymbol{\theta}, \boldsymbol{\varphi}$
    end for
end for

B Proofs

Proposition 1. *The posterior targeted by UA-SABI is the same as the E-Post posterior.*

Proof. In UA-SABI, as per Eq. (8), we sample from the joint distribution

$$p(\boldsymbol{\omega}, \mathbf{y}, \mathbf{c}, \sigma \mid D_T) = p(\mathbf{y} \mid \boldsymbol{\omega}, \mathbf{c}, \sigma)\, p(\boldsymbol{\omega})\, p(\mathbf{c}, \sigma \mid D_T) \tag{14}$$

to train the conditional neural density estimator $q(\boldsymbol{\omega} \mid \mathbf{y})$. We only condition on the data $\mathbf{y}$ and treat the surrogate parameters $(\mathbf{c}, \sigma)$ as nuisance parameters ignored by $q(\boldsymbol{\omega} \mid \mathbf{y})$. This means that we (implicitly) integrate over the distribution of $(\mathbf{c}, \sigma)$. Hence, UA-SABI targets the following posterior:

$$p_{\mathrm{UA-SABI}}(\boldsymbol{\omega} \mid \mathbf{y}, D_T) \propto \iint p(\mathbf{y} \mid \boldsymbol{\omega}, \mathbf{c}, \sigma)\, p(\boldsymbol{\omega})\, p(\mathbf{c}, \sigma \mid D_T)\, d\mathbf{c}\, d\sigma \tag{15}$$

On the other hand, E-Post targets the following posterior, as per Eq. (10):

$$p_{\mathrm{E-Post}}(\boldsymbol{\omega} \mid \mathbf{y}, D_T) = \iint p(\boldsymbol{\omega} \mid \mathbf{y}, \mathbf{c}, \sigma)\, p(\mathbf{c}, \sigma \mid D_T)\, d\mathbf{c}\, d\sigma \tag{16}$$

$$\propto \iint p(\mathbf{y} \mid \boldsymbol{\omega}, \mathbf{c}, \sigma)\, p(\boldsymbol{\omega})\, p(\mathbf{c}, \sigma \mid D_T)\, d\mathbf{c}\, d\sigma, \tag{17}$$

which shows $p_{\mathrm{UA-SABI}}(\boldsymbol{\omega} \mid \mathbf{y}, D_T) = p_{\mathrm{E-Post}}(\boldsymbol{\omega} \mid \mathbf{y}, D_T)$. □

C Additional Case Study Details

C.1 Surrogate Models

In case study 1, we train a Bayesian PCE with 2-dimensional Legendre polynomials and a maximum total degree of 3, resulting in $J = 10$ polynomials. We set a normal prior for the surrogate coefficients, $p(\mathbf{c}) = \mathcal{N}(0, 5)$, and a half-normal prior for the approximation error parameter, $p(\sigma) = \text{Half-}\mathcal{N}(0.5)$.

In case study 2 and 3, we consider a Bayesian PCE with 3- and 4-dimensional aPC polynomials for the CO_2 model (see Section 4.3) and a maximum total degree of 3, resulting in $J = 19$ and $J = 35$ polynomials respectively. Following Bürkner et al. (2023), we place a sparsity-inducing R2D2 prior on the surrogate coefficients $\mathbf{c}$ with $R^2 \sim \text{Beta}(0.5, 2)$. For the approximation error we set the prior as $p(\sigma) = \text{Half-}\mathcal{N}(0.1)$.

C.2 Neural Posterior Estimation

For the first two case studies, we used the same NPE setup. The summary network $S_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$ is a *DeepSet* Zaheer et al. (2017) composed of two hidden layers, each containing 10 neurons. It outputs a 10-dimensional summary vector. For the inference network $I_{\boldsymbol{\varphi}}(\mathbf{s})$, we employ a coupling flow as implemented in Radev et al. (2023b). For the third case study, we did not use a summary network as the measurement locations in the experiment were fixed. For the inference network $I_{\boldsymbol{\varphi}}(\mathbf{s})$, we used a spline coupling flow (Durkan et al., 2019) as implemented in Radev et al. (2023b). The training process employed a cosine learning rate scheduler with an initial learning rate of 5×10^{-4} and a minimum learning rate fraction of $\alpha = 10^{-6}$. The scheduler operated over a total of 12,800 steps, corresponding to 128 batches per epoch over 100 epochs. All NPE models were trained using the Adam optimizer (Kingma & Ba, 2017) for 100 epochs.

D Additional Results for Case Study 2: CO_2 Storage Model

This section shows the recovery and ECDF difference plots for porosity of the CO_2 storage model shown for SABI vs. Point and UA-SABI vs. E-Post.

In Fig. 11 we show that SABI and Point as well as UA-SABI and E-Post yield similar results in both recovery and calibration. The notably wide intervals observed for Point, especially compared to SABI in the recovery plots suggest convergence problems in the MCMC, likely due to the low standard deviation in the likelihood. These convergence problems may explain the inconsistencies between SABI and Point. In comparison, their uncertainty-aware counterparts UA-SABI and E-Post produce highly similar estimates.

E Additional Results for Case Study 3: MICP Model

E.1 Parameter Posterior with Original Priors

Performing inference with the original prior for k_{ub} leads, regardless of the inference method, to point-like posteriors at the upper boundary of the prior. The reason is the same for all methods: the maximal value of k_{ub} yields the highest output values. Since the measurement data are always higher than the surrogate output under the original prior, the posterior collapses to a point-like estimate at the maximum admissible value of k_{ub}, as exemplified in Fig. 8.

E.2 Sensitivity Analysis

We conducted a sensitivity analysis on the surrogate to identify the parameters to which the surrogate output is most sensitive. Sobol indices (Sobol', 1990) decompose the output variance into contributions from each parameter (including their interactions), with the total Sobol index of a parameter quantifying its overall influence on the output variance. Figure 13 presents the distributions of the total Sobol indices (Le Gratiet et al., 2017) for each parameter at each location, noting that each location has its own surrogate. Since the Bayesian PCE yields samples of the coefficients, the corresponding Sobol indices are also obtained in the

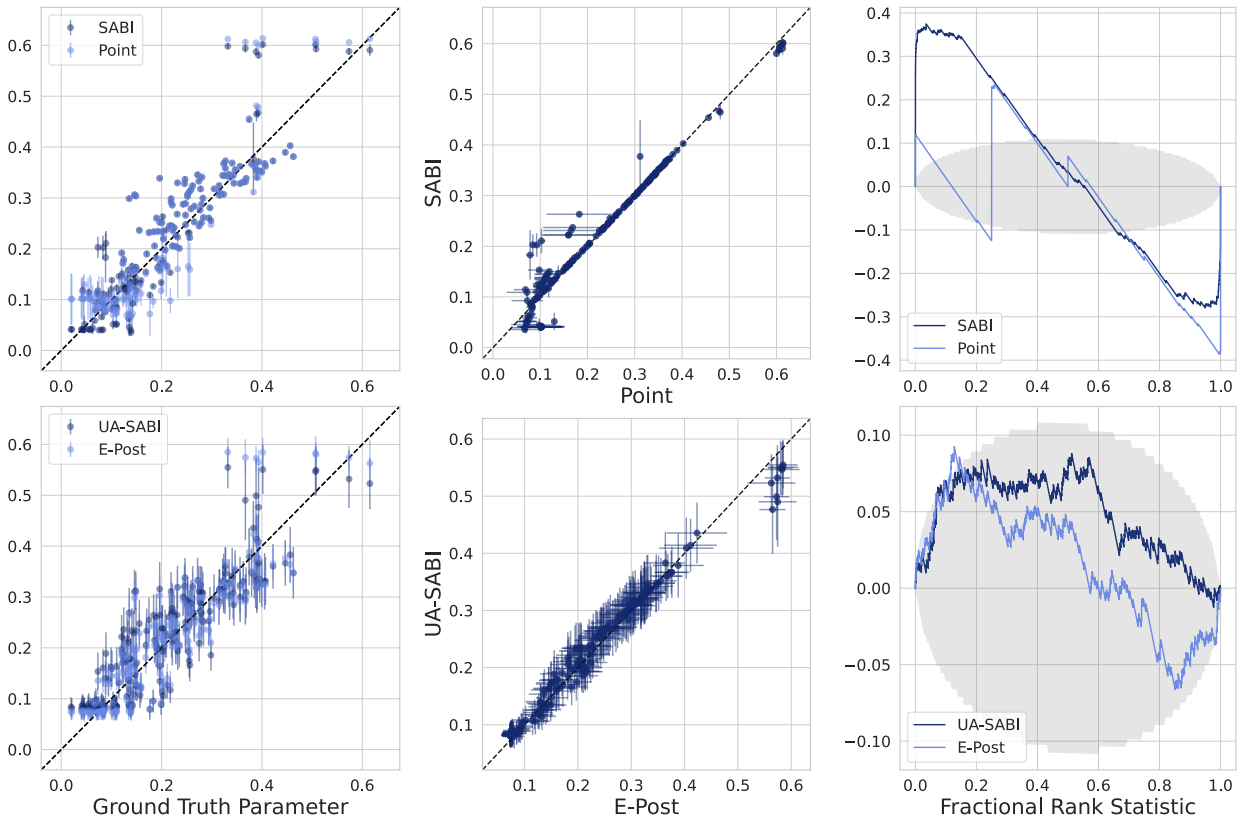


Figure 11: CO_2 recovery plots comparing to ground truth and MCMC full reference solution, ECDF difference plots (from left to right) for 200 ground truth samples: SABI vs. Point (top) and UA-SABI vs. E-Post (bottom). For ECDF difference plots, empirical ranks are shown in blue, 95% confidence bands assuming calibration are shown in grey.

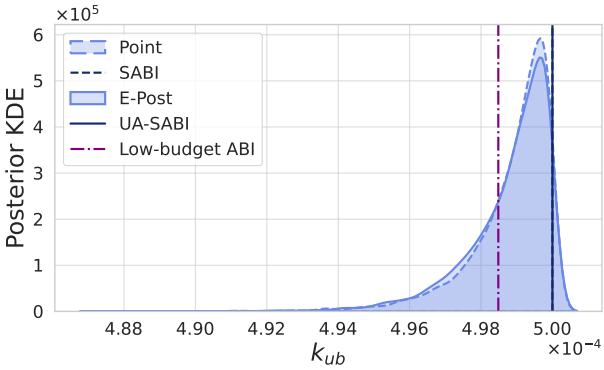


Figure 12: KDEs of 4,000 parameter posterior samples for k_{ub} of the MICP model obtained with low-budget ABI, Point, E-Post, SABI and UA-SABI using the original parameter prior.

form of distributions. Figure 13 clearly shows that almost all the variance in the output at all locations is determined by k_{ub}.

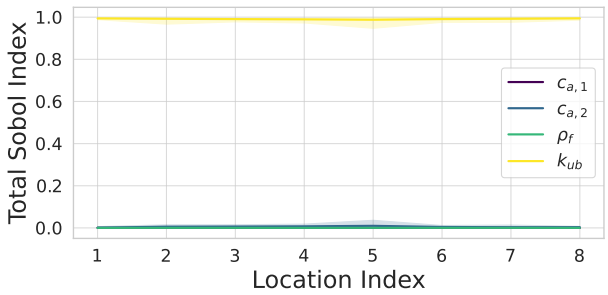


Figure 13: Distribution of total Sobol indices over all locations for the four parameters $c_{a,1}, c_{a,2}, \rho_f, k_{ub}$ of the MICP model.

E.3 Parameter Posterior for Remaining Parameters

Figure 14 shows the KDEs of 4,000 parameter samples for the 3 remaining parameters, $c_{a,1}, c_{a,2}$, and ρ_f, obtained with low-budget ABI, Point, E-Post, SABI, and UA-SABI. Two distinct scenarios emerge: 1) the posterior collapses to a point estimate (often at the boundary of the prior), or 2) the posterior remains close to the prior distribution. Point-like posteriors are produced by low-budget ABI or Point, while surrogate-based methods yield posteriors resembling the prior. Both outcomes can be explained by the lack of sensitivity of the output to these parameters, as shown by the total Sobol indices in Fig. 13. This missing sensitivity makes reliable inference very difficult.

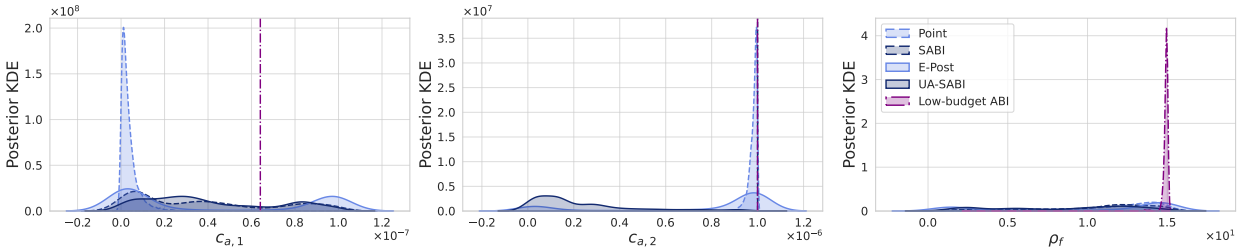


Figure 14: KDEs of 4,000 parameter posterior samples for the remaining parameters $c_{a,1}, c_{a,2}$, and ρ_f of the MICP model obtained with low-budget ABI, Point, E-Post, SABI and UA-SABI.

E.4 Runtime Comparison

To compare runtimes as in case studies 1 and 2, inference must be performed on multiple datasets. Since only one dataset of real measurements is available, we conducted the runtime analysis on synthetic datasets. The experiments were also run on a computing cluster with two AMD EPYC 7551 CPUs (totaling 64 physical cores) to speed up E-Post through parallelization.

Figure 15 shows the measured runtimes of UA-SABI and E-Post for $2, 3, 4, 5$ inference runs, with E-Post parallelized across 16 cores. We observe a break-even point between 3 and 4 runs, meaning that UA-SABI becomes advantageous starting from 4 runs. Despite E-Post leveraging more cores, the training costs of UA-SABI are amortized earlier when the model is more expensive.

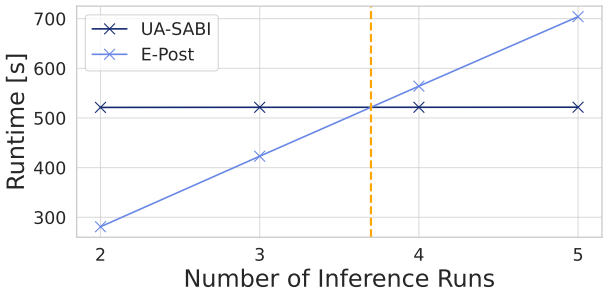


Figure 15: Comparison of runtimes for MICP for {2,3,4,5} runs for UA-SABI (training and inference) and E-Post (inference) with break-even point.