
Cost-effective Reduced-Order Modeling via Bayesian Active Learning

Amir Hossein Rahmati¹ Nathan M. Urban² Byung-Jun Yoon^{1,2} Xiaoning Qian^{1,2}

¹ Texas A&M University, College Station, TX ² Brookhaven National Laboratory, Upton, NY

{amir_hossein_rahmati, bjyoon, xqian}@tamu.edu
{nurban, byoon, xqian1}@bnl.gov

Abstract

Machine Learning surrogates have been developed to accelerate solving systems dynamics of complex processes in different science and engineering applications. To faithfully capture governing systems dynamics, these methods rely on large training datasets, hence restricting their applicability in real-world problems. In this work, we propose BayPOD-AL, an active learning framework based on an uncertainty-aware Bayesian proper orthogonal decomposition (POD) approach, which aims to effectively learn reduced-order models from high-fidelity full-order models representing complex systems. Experimental results on predicting the temperature evolution over a rod demonstrate BayPOD-AL’s effectiveness in suggesting the informative data and reducing computational cost related to constructing a training dataset compared to other uncertainty-guided active learning strategies. Furthermore, we demonstrate BayPOD-AL’s generalizability and efficiency by evaluating its performance on a dataset of higher temporal resolution than the training dataset.

1 Introduction

Many real-world decision-making problems benefit from proper modeling of high-dimensional complex systems dynamics. While traditional physics-principled computational models based on differential equations provide high-fidelity solutions, their prohibitive computational cost hinders their applications [Boluki et al., 2024]. With the abundance of data, from both high-fidelity simulations and real-world measurements, machine learning (ML) surrogate models have become one of the exciting emerging solutions to learn the underlying governing dynamics of many real-world problems [Boluki et al., 2024, Verma, 2020]. These surrogates have enabled efficient modeling of complex processes, instead of solely depending on solving their corresponding time-consuming, computationally expensive Ordinary or Partial Differential equation systems (ODEs/PDEs). More specifically, recent studies have been investigating the development of ML methods to accelerate these computations, in a spectrum from purely data-driven ML surrogates, physics-informed neural networks (PINNs), to more recent hybrid models such as ML-augmented reduced-order models [Raissi et al., 2019, Swischuk et al., 2019, Guo et al., 2022, Hirsh et al., 2021, Lagaris et al., 1998]. For instance, Rudy et al. [2016] utilize sparse regression to learn a system’s governing PDEs while Raissi et al. [2017], Raissi [2018], Zhu and Zabaras [2018] consider black-box neural network models trained by “physics-informed” loss functions. Although these methods center their attention on providing accurate solutions, they often need retraining with a change of system settings, including parameters as well as initial and boundary conditions. In DeGennaro et al. [2019], the authors proposed a two-step method to infer the model parameter posterior enabling uncertainty

quantification after differential equation system identification. A Bayesian framework for deriving reduced-order models (ROMs), BayPOD [Boluki et al., 2024], has been recently developed in an end-to-end manner based on proper orthogonal decomposition (POD), motivated by the idea of developing ML-augmented ROMs with embedded physics constraints [Swischuk et al., 2019]. ROM methods aim to derive physics-principled surrogates of high-fidelity complex models in significantly reduced lower-dimensional space to reduce the computational load while maintaining the desired accuracy [Hesthaven and Ubbiali, 2018]. By formulating ROM learning in a Bayesian framework, BayPOD is capable of quantifying the uncertainty in addition to providing approximate high-fidelity differential equation solutions. This is a pivotal feature in scenarios where there is little observed data available, not atypical in science and engineering applications.

Although these endeavors have facilitated new ML surrogates for more efficient modeling and forecasting of complex processes, they often require considerably large training datasets to achieve satisfactory performances. Acquiring such training data from high-fidelity full-order models (FOMs) has heavy computational demands. To develop more data-efficient ML surrogates for ROM learning of complex systems, inspired by recent advancements of Active Learning (AL) in the ML community [Wang and Shang, 2014, Ash et al., 2020, Houlsby et al., 2011, Wu et al., 2022, Settles, 2009, Ren et al., 2021], we here aim to bridge this gap by developing active learning for ROMs instead of constructing ROMs based on large batches of randomly generated FOM data. More specifically, via iteratively suggesting the most informative data we intend to improve sample efficiency while preserving the underlying surrogate model’s performance, reliability, and interoperability.

To develop and evaluate active learning for ROMs, we focus on BayPOD as the learned surrogate models by BayPOD come with their inherent uncertainty quantification capabilities in the adopted Bayesian learning framework. Here we promote a robust AL framework, BayPOD-AL, designed to showcase the feasibility and effectiveness of active learning for ROMs. With quantified uncertainty in BayPOD, we explore different uncertainty-based active learning strategies to utilize the estimated uncertainty for efficient guidance of the AL procedure. Recent studies in ML have suggested that uncertainty-based AL (UAL) methods can be unreliable in improving sample efficiency while optimizing the model’s performance under specific scenarios [Munjal et al., 2022, Saifullah et al., 2022, Hacohen et al., 2022, Rahmati et al., 2024]. Exploring and evaluating different UAL strategies for ROM learning is critical to help understand and prevent potential performance degradation under the new ROM learning settings. In particular, as we focus on modeling complex systems behavior with ROMs, there is an inherent mismatch between the ROM surrogates and the systems to approximate. In Rahmati et al. [2024], the authors suggested error-based acquisition functions to alleviate the issues caused by the model mismatch. When developing ROMs for differential equation systems, the Mean Squared Error (MSE) is often considered as the target criterion. In this study, we propose and evaluate *BayPOD-UAL* that is guided by an acquisition function depending solely on the estimated uncertainty and *BayPOD-EAL* that relies on the estimated error. BayPOD-EAL, by taking advantage of the results in Savvides et al. [2024], is expected to achieve higher sample efficiency due to its objective-driven formulation directly targeting reducing MSE for ROM learning [Yoon et al., 2013, Boluki et al., 2019, Yoon et al., 2021]. Throughout our experiments, we demonstrate the effectiveness of BayPOD-AL in improving sample efficiency when learning a BayPOD surrogate predicting the temperature evolution over a rod. Finally, by investigating its performance on a temporally high-resolution dataset, we further show its efficiency and robustness.

2 Active Learning for Reduced-Order Models

Before delving into active learning for ROMs, we first briefly review ROMs, especially the family of POD-based ROMs, including BayPOD.

2.1 Reduced-Order Models

As mentioned in Section 1, solving the full-order models (FOMs) to provide the high-fidelity solutions of the governing ODEs/PDEs is prohibitively expensive. Reduced-order models (ROMs) are designed with the objective of reducing the computational cost by estimating FOM solutions in a lower-dimensional space, subject to keeping the information loss to a minimum [Benner et al., 2015, Besselink et al., 2013, Penzl, 2006, Swischuk et al., 2019, Pant et al., 2021]. Compared to pure data-driven “black-box” ML surrogates, learning ROMs of differential equation systems, naturally allows

integrating the underlying scientific principles. We focus on proper orthogonal decomposition (POD), one of the most widely used model reduction methods to derive low-dimensional representations of the high-dimensional system states [Swischuk et al., 2019]. Following the notations from Boluki et al. [2024], consider a function $f : \mathcal{X} \times \mathcal{T} \times \mathcal{P} \rightarrow \mathbb{R}$, where \mathcal{X} , the spatial domain, \mathcal{T} , the time domain, and \mathcal{P} , the input domain are mapped to a physical field. Denoting a snapshot $\mathbf{f}(t; \mathbf{p}) \in \mathbb{R}^{n_x}$ as the discretized spatial domain at time t and input model parameters \mathbf{p} , with n_t different time points and n_p different inputs, the POD bases can be obtained via singular value decomposition (SVD). Specifically, defining $F = [\mathbf{f}(t_i; \mathbf{p}_j)] \in \mathbb{R}^{n_s \times n_x}$ with total $n_s = n_t n_p$ snapshots, the SVD is $F = V\Sigma W$, which enables approximating field f by the K -dimension POD basis for any input model parameters.

2.2 Bayesian POD (BayPOD)

BayPOD focuses on simultaneously deriving low-dimensional projection and mapping from input parameters of the full-order model to latent projection coefficients. To learn the map $\alpha : \mathcal{P} \times \mathcal{T} \rightarrow \mathcal{A}$, we consider neural network mappings. To account for physics constraints, BayPOD reformulates the linear representation of each snapshot’s approximation, $\tilde{\mathbf{f}}$, constituting of the POD approximation and a *particular solution* given the corresponding initial and boundary conditions $\tilde{\mathbf{f}}(t; \mathbf{p}) = \bar{\mathbf{f}} + \sum_{k=1}^K \bar{\mathbf{v}}_k \alpha_k(t; \mathbf{p})$, where $\alpha_k(t; \mathbf{p})$ is the corresponding *learnable* POD expansion coefficient, and $\bar{\mathbf{v}}_k \in \mathbb{R}^{n_x}$ is the k -th left singular vector of F as the POD bases that satisfy homogeneous boundary conditions. Here $\bar{\mathbf{f}}$ is the *particular solution* given the set of potentially inhomogeneous initial and boundary conditions for better embedding physics constraints [Swischuk et al., 2019, Boluki et al., 2024]. The corresponding field value of snapshot s at the spatial point x , \tilde{f}_{sx} , is modeled as a normal random variable:

$$\tilde{f}_{sx} \sim \mathbf{N}(\mathbf{u}_x^\top \boldsymbol{\alpha}_s, \gamma_x^{-1}) \quad (1)$$

with $\mathbf{u}_x \in \mathbb{R}^K$ the K -dimensional POD basis at position x , and γ_x^{-1} the variance at x . Using mean-field variational inference, BayPOD finds the variational posterior of model parameters. Due to its generative nature, it provides uncertainty estimates of predicted system dynamics in different setups, which is the enabler of optimal and adaptive decision making, active learning for sample efficiency in this work.

2.3 BayPOD-AL

We now present our active learning framework, BayPOD-AL, by exploring different uncertainty-based active learning (UAL) strategies, leveraging the inherent UQ capabilities of BayPOD. Consider \mathbf{D}_L and \mathbf{D}_U the iteratively updated ‘labeled’ and ‘unlabeled’ datasets corresponding to collected snapshots from high-fidelity FOMs and the FOM settings without actual simulated snapshots, respectively. BayPOD-AL focuses on reducing the cost of running FOM simulations to train BayPOD by choosing the most informative FOM settings in \mathbf{D}_U to collect new BayPOD training data from FOM based on an acquisition function a . Specifically, BayPOD-AL iteratively queries for snapshots, solutions to corresponding FOM differential equations, with the considered most informative settings, the corresponding input FOM parameters in this work $\mathbf{p}^* = \arg \max_{\mathbf{p}_U \in \mathbf{D}_U} a(q(\cdot | \mathbf{D}_L), \mathbf{p}_U)$. Here the acquisition function $a(q(\cdot | \mathbf{D}_L), \mathbf{p}_U)$ guides the AL procedure considering the potential uncertainty of iteratively updated BayPOD models. At each AL iteration, with the currently learned BayPOD model, BayPOD-AL determines the most informative input FOM parameter in \mathbf{D}_U , i.e. \mathbf{p}^* , as the output to query additional training snapshots from the FOM. Note that BayPOD directly provides the model posterior given the labeled data or simulated snapshots, $q(\cdot | \mathbf{D}_L)$. Until the trained BayPOD model reaches a satisfactory performance, BayPOD-AL continues to add new simulated FOM snapshots to \mathbf{D}_L .

In our BayPOD-AL framework, due to solving a homogeneous problem that frees us from worrying about boundary/initial constraints, we only search for the most ‘informative’ input FOM parameters $\mathbf{p}^* \in \mathbf{D}_U \subset \mathcal{P}$. Assuming that \mathbf{D}_U contains n_p inputs, and for each input $\mathbf{p}_U \in \mathbf{D}_U$, we want to query FOM solutions for n_t^p **fixed** time points. At each AL iteration, we compute the acquisition function for a batch of n_t^p snapshots for **each** input. Finally, the most informative snapshots from FOMs will be appended to \mathbf{D}_L for the next AL and model update iteration. This process continues

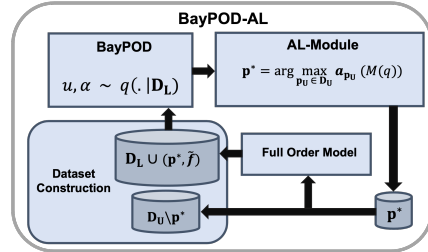


Figure 1: A schematic illustration of the proposed BayPOD-AL framework.

until the model reaches the desired performance. Having access to the BayPOD’s variational posterior $q(\cdot|\mathbf{D}_L)$, we define a measure function $M^{(t)}(q(\cdot|\mathbf{D}_L, \mathbf{p}_U, x)) : \mathcal{U} \times \mathcal{A} \rightarrow \mathbb{R}$ estimating the ‘informativeness’ of snapshots at input \mathbf{p}_U , time t , and position x . By taking the average over all snapshots given \mathbf{p}_U , we define the following acquisition function evaluating such ‘informativeness’ based on the adopted measure function:

$$\mathbf{a}_{\mathbf{p}_U}^{(i)} = \frac{1}{n_t^{\mathbf{p}}} \sum_t \frac{1}{n_x} \sum_x M^{(i,t)}(q(\cdot|\mathbf{D}_L, \mathbf{p}_U, x)), \quad i \in \{0, \dots, n_{\mathbf{p}}\} \quad (2)$$

BayPOD-AL then queries the corresponding FOM for snapshots by the most ‘informative’ input:

$$\mathbf{p}^* = \mathbf{p}_U^{(i)} = \arg \max_i \mathbf{a}_{\mathbf{p}_U}^{(i)}. \quad (3)$$

Figure 1 provides a schematic illustration of BayPOD-AL. As discussed in Hino [2020], Zhao et al. [2021], Rahmati et al. [2024], the acquisition function plays a critical role in achieving desired AL efficiency. To derive efficient active learning for BayPOD, we evaluate two different choices by considering: 1) *BayPOD-UAL*: the predictive model uncertainty based on the estimated posterior predictive variance from BayPOD; and 2) *BayPOD-EAL*: the estimated approximation error, directly targeting the ROM learning objective, for which we estimate the upper bound of the approximation error and utilize that as the measure function in (2). Details of each strategy are provided in Appendix B and C.

3 Experiments

Using the same example as in Boluki et al. [2024], we implement our BayPOD-AL on predicting the evolution of temperature fields, f , with heat diffusivity parameter, κ , over a rod of length L with time-dependent boundary conditions. In the following, we report the performance statistics of five runs of experiments for each of AL algorithms. Each run is different only in the initial \mathbf{D}_L . We compare the performance of BayPOD-UAL and BayPOD-EAL with the random sampling strategy which selects the next κ for FOM simulations randomly at each iteration.

We train BayPOD on the temporally low-resolution training snapshots and report the AL performances on both temporally low- ($n_t^{\mathbf{p}} = 50$) and high-resolution ($n_t^{\mathbf{p}} = 200$) test datasets. Figure 2 compares the performance of BayPOD-EAL, BayPOD-UAL, and random sampling strategies. Table 1 summarizes the performance statistics of our results after 5 AL iterations. It is clear after 5 AL iterations (250 new snapshots), BayPOD-EAL leads to a model with the best empirical performance. Over the first 10 AL iterations, on average it has $4.7\times$ and $9\times$ lower MSE than BayPOD-UAL, and $6\times$ and $6.5\times$ lower MSE than random sampling when evaluated on low- and high-resolution datasets respectively. This further demonstrates the robustness of BayPOD-EAL, in leading the model to optimal performance even when the training data resolution differs from the test data, owing to it being objective-driven. As mentioned in Section 1, due to the inherent model mismatch, uncertainty alone cannot efficiently reduce the labeling cost for learning ROMs. On both datasets, BayPOD-UAL grants an initial performance lead compared to random sampling, but it soon slows down with comparable performance for the low-resolution dataset and worse performance for the high-resolution dataset. After 15 to 20 AL iterations, both BayPOD-EAL and random sampling provide comparable performance, meaning that BayPOD-AL reduces the computational cost related to training data by a factor of 3 to 4, demonstrating its cost-effectiveness in learning ROMs.

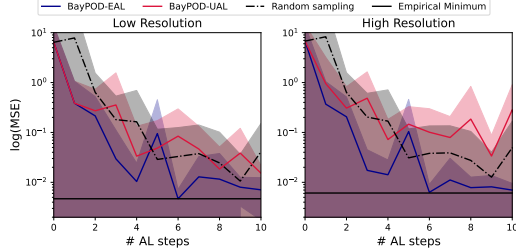


Figure 2: Performance comparison of BayPOD-EAL, BayPOD-UAL, and random sampling strategies on ‘low’ (left) and ‘high’ (right) temporal resolution test datasets.

4 Conclusion

To model the complex systems dynamics, ML surrogate models have shown promising performance. However, the prohibitive cost of acquiring the solutions of the high-fidelity FOMs representing them

Table 1: Mean and standard deviation (std) of approximation error at the 6th AL iterations from five random runs of experiments. Best results are in bold.

| Method | Low Resolution | | High Resolution | |
|-----------------|---|--|--|---|
| | mean | std | mean | std |
| BayPOD-EAL | 4.68×10^{-3} | 1.3×10^{-3} | 6.3×10^{-3} | 1.79×10^{-3} |
| BayPOD-UAL | 8.42×10^{-2} | 1.08×10^{-1} | 1.01×10^{-1} | 1.01×10^{-1} |
| Random Sampling | 3.29×10^{-2} | 4.7×10^{-2} | 3.8×10^{-2} | 4.95×10^{-2} |

hinders the applicability of these models in real-world scenarios. Inspired by recent advances in AL, we present the BayPOD-AL framework, which iteratively suggests the most informative data that can boost the surrogate model’s performance. The Bayesian framework explicitly updates the model posterior, with which different objective-driven acquisition functions targeting ROM learning can be adopted to achieve efficient AL. Our experiments demonstrate the robustness of the proposed framework as well as its efficacy in reducing the cost of learning ROMs, thereby improving their applicability in real-world problems.

Acknowledgements This work was supported in part by the U.S. National Science Foundation (NSF) grants IIS-2212419; and by the U.S. Department of Energy (DOE) Office of Science, Advanced Scientific Computing Research (ASCR) M2DT Mathematical Multifaceted Integrated Capability Center (MMICC) under Award B&R# KJ0401010/FWP# CC130, program manager W. Spatz, and Award DE-SC0012704, program manager Margaret Lentz. Portions of this research were conducted with the advanced computing resources provided by Texas A&M High Performance Research Computing.

References

- J. T. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds, 2020. URL <https://arxiv.org/abs/1906.03671>.
- P. Benner, S. Gugercin, and K. Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531, 2015. doi: 10.1137/130932715. URL <https://doi.org/10.1137/130932715>.
- B. Besselink, U. Tabak, A. Lutowska, N. van de Wouw, H. Nijmeijer, D. Rixen, M. Hochstenbach, and W. Schilders. A comparison of model reduction techniques from structural dynamics, numerical mathematics and systems and control. *Journal of Sound and Vibration*, 332(19):4403–4422, 2013. ISSN 0022-460X. doi: <https://doi.org/10.1016/j.jsv.2013.03.025>. URL <https://www.sciencedirect.com/science/article/pii/S0022460X1300285X>.
- S. Boluki, X. Qian, and E. R. Dougherty. Experimental design via generalized mean objective cost of uncertainty. *IEEE Access*, 7:2223–2230, 2019. doi: 10.1109/ACCESS.2018.2886576.
- S. Boluki, S. Z. Dadaneh, E. R. Dougherty, and X. Qian. Bayesian proper orthogonal decomposition for learnable reduced-order models with uncertainty quantification. *IEEE Transactions on Artificial Intelligence*, 5(3):1162–1173, 2024. doi: 10.1109/TAI.2023.3268609.
- A. M. DeGennaro, N. M. Urban, B. T. Nadiga, and T. Haut. Model structural inference using local dynamic operators. *International Journal for Uncertainty Quantification*, 9(1):59–83, 2019. ISSN 2152-5080. doi: 10.1615/int.j.uncertaintyquantification.2019025828. URL <http://dx.doi.org/10.1615/Int.J.UncertaintyQuantification.2019025828>.
- M. Guo, S. A. McQuarrie, and K. E. Willcox. Bayesian operator inference for data-driven reduced-order modeling. *Computer Methods in Applied Mechanics and Engineering*, 402:115336, 2022. ISSN 0045-7825. doi: <https://doi.org/10.1016/j.cma.2022.115336>. URL <https://www.sciencedirect.com/science/article/pii/S0045782522004273>. A Special Issue in Honor of the Lifetime Achievements of J. Tinsley Oden.
- G. Hachohen, A. Dekel, and D. Weinshall. Active learning on a budget: Opposite strategies suit high and low budgets, 2022. URL <https://arxiv.org/abs/2202.02794>.

- J. Hesthaven and S. Ubbiali. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *Journal of Computational Physics*, 363:55–78, 2018. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2018.02.037>. URL <https://www.sciencedirect.com/science/article/pii/S0021999118301190>.
- H. Hino. Active learning: Problem settings and recent developments, 2020. URL <https://arxiv.org/abs/2012.04225>.
- S. M. Hirsh, D. A. Barajas-Solano, and J. N. Kutz. Sparsifying priors for bayesian uncertainty quantification in model discovery, 2021. URL <https://arxiv.org/abs/2107.02107>.
- N. Houlsby, F. Huszár, Z. Ghahramani, and M. Lengyel. Bayesian active learning for classification and preference learning, 2011. URL <https://arxiv.org/abs/1112.5745>.
- I. Lagaris, A. Likas, and D. Fotiadis. Artificial neural networks for solving ordinary and partial differential equations. *IEEE Transactions on Neural Networks*, 9(5):987–1000, 1998. ISSN 1045-9227. doi: 10.1109/72.712178. URL <http://dx.doi.org/10.1109/72.712178>.
- P. Munjal, N. Hayat, M. Hayat, J. Sourati, and S. Khan. Towards robust and reproducible active learning using neural networks, 2022. URL <https://arxiv.org/abs/2002.09564>.
- P. Pant, R. Doshi, P. Bahl, and A. Barati Farimani. Deep learning for reduced order modelling and efficient temporal evolution of fluid simulations. *Physics of Fluids*, 33(10), Oct. 2021. ISSN 1089-7666. doi: 10.1063/5.0062546. URL <http://dx.doi.org/10.1063/5.0062546>.
- T. Penzl. Algorithms for model reduction of large dynamical systems. *Linear Algebra and its Applications*, 415(2):322–343, 2006. ISSN 0024-3795. doi: <https://doi.org/10.1016/j.laa.2006.01.007>. URL <https://www.sciencedirect.com/science/article/pii/S0024379506000371>. Special Issue on Order Reduction of Large-Scale Systems.
- A. H. Rahmati, M. Fan, R. Zhou, N. M. Urban, B.-J. Yoon, and X. Qian. Understanding uncertainty-based active learning under model mismatch, 2024. URL <https://arxiv.org/abs/2408.13690>.
- M. Raissi. Deep hidden physics models: Deep learning of nonlinear partial differential equations, 2018. URL <https://arxiv.org/abs/1801.06637>.
- M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations, 2017. URL <https://arxiv.org/abs/1711.10561>.
- M. Raissi, P. Perdikaris, and G. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2018.10.045>. URL <https://www.sciencedirect.com/science/article/pii/S0021999118307125>.
- P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, B. B. Gupta, X. Chen, and X. Wang. A survey of deep active learning, 2021.
- S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz. Data-driven discovery of partial differential equations, 2016. URL <https://arxiv.org/abs/1609.06401>.
- S. Saifullah, S. Agne, A. Dengel, and S. Ahmed. Analyzing the potential of active learning for document image classification, 11 2022.
- R. Savvides, H. Luu, and K. Puolamäki. Error bounds for any regression model using Gaussian processes with gradient information. *Proceedings of Machine Learning Research*, 238:397–405, 2024. ISSN 2640-3498. URL <http://aistats.org/aistats2024/>. International Conference on Artificial Intelligence and Statistics, AISTATS ; Conference date: 02-05-2024 Through 04-05-2024.
- B. Settles. Active learning literature survey. 2009.

- R. Swischuk, L. Mainini, B. Peherstorfer, and K. Willcox. Projection-based model reduction: Formulations for physics-based machine learning. *Computers Fluids*, 179:704–717, 2019. ISSN 0045-7930. doi: <https://doi.org/10.1016/j.compfluid.2018.07.021>. URL <https://www.sciencedirect.com/science/article/pii/S0045793018304250>.
- S. Verma. A survey on machine learning applied to dynamic physical systems, 2020. URL <https://arxiv.org/abs/2009.09719>.
- D. Wang and Y. Shang. A new active labeling method for deep learning. In *2014 International Joint Conference on Neural Networks (IJCNN)*, pages 112–119, 2014. doi: 10.1109/IJCNN.2014.6889457.
- J. Wu, J. Chen, and D. Huang. Entropy-based active learning for object detection with progressive diversity constraint, 2022. URL <https://arxiv.org/abs/2204.07965>.
- B.-J. Yoon, X. Qian, and E. R. Dougherty. Quantifying the objective cost of uncertainty in complex dynamical systems. *IEEE Transactions on Signal Processing*, 61(9):2256–2266, 2013. doi: 10.1109/TSP.2013.2251336.
- B.-J. Yoon, X. Qian, and E. R. Dougherty. Quantifying the multi-objective cost of uncertainty. *IEEE Access*, 9:80351–80359, 2021. doi: 10.1109/ACCESS.2021.3085486.
- G. Zhao, E. Dougherty, B.-J. Yoon, F. J. Alexander, and X. Qian. Bayesian active learning by soft mean objective cost of uncertainty. In A. Banerjee and K. Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 3970–3978. PMLR, 13–15 Apr 2021. URL <https://proceedings.mlr.press/v130/zhao21c.html>.
- Y. Zhu and N. Zabarbas. Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. *Journal of Computational Physics*, 366:415–447, Aug. 2018. ISSN 0021-9991. doi: 10.1016/j.jcp.2018.04.018. URL <http://dx.doi.org/10.1016/j.jcp.2018.04.018>.

A BayPOD: Posterior Inference

As mentioned in Section 2.2 the response is modeled as a normally-distributed random variable, $\tilde{f}_{sx} \sim \mathbf{N}(\mathbf{u}_x^\top \boldsymbol{\alpha}_s, \gamma_x^{-1})$, with $\boldsymbol{\alpha}_s, \mathbf{u}_x \in \mathbb{R}^K$ the K -dimensional POD basis vector at x , and K POD coefficients for snapshot s . Considering zero-mean normal priors on POD basis and coefficients, i.e., $\mathbf{u}_x \sim \mathbf{N}(0, I)$ and $\boldsymbol{\alpha}_s \sim \mathbf{N}(0, \gamma_\alpha^{-1}I)$, as well as conjugate gamma distributions over the precision parameters, $\gamma_\alpha, \gamma_x \sim \text{Gamma}(1, 1)$, the model is completed.

Using variational inference, the posteriors over parameters are inferred. More specifically, variational distributions $q(\cdot)$ is put over model parameters with the independence assumption, that is $q(\mathbf{u}, \boldsymbol{\alpha}, \gamma) = q(\mathbf{u})q(\boldsymbol{\alpha})q(\gamma)$. Since we consider using a neural network (NN) for coefficient mapping, the variational distribution over $\boldsymbol{\alpha}_s$ can be defined as $q(\boldsymbol{\alpha}_s) = \mathbf{N}(\boldsymbol{\alpha}_s; \mu_w(\mathbf{p}), \Sigma_w(\mathbf{p}))$, with μ_w and Σ_w the mean and covariance matrix of NN form with weights w . Similar as in Boluki et al. [2024], we employ the same NN architecture with two hidden layers, 50 nodes per layer, and using rectified linear unit (ReLU) activation functions. To benefit from conjugate priors, the variational posteriors for POD basis and precision parameters are set to be normal and gamma distributions. Finally, by minimizing the Kullback-Leibler (KL) divergence between the variational posteriors and the true posteriors, the optimal parameters are obtained.

B BayPOD-UAL: Uncertainty-guided AL

The acquisition function can be utilized by adopting any measure function in BayPOD-AL. More specifically, due to the availability of parameter posterior distributions in BayPOD, there is flexibility in defining uncertainty-based measure functions. For BayPOD-UAL, we define a measure function dependent on each snapshot’s predictive posterior variance. Specifically, using Monte Carlo (MC) sampling we estimate $\hat{V}^{(t)}(\mathbf{p}_U, x)$, the sample variance of each input \mathbf{p}_U at the specific time point t and position x . Finally, by setting $M = \hat{V}$ in (2), \mathbf{p}^* can be found by (3).

C BayPOD-EAL: Error-guided AL

Due to the potential mismatch between ROMs and their corresponding high-fidelity model solution, BayPOD-UAL may lead to degraded AL performance as shown in other ML problems [Munjal et al., 2022, Saifullah et al., 2022, Rahmati et al., 2024]. Consequently, we develop BayPOD-EAL, which benefits from an error-dependent measure function. In this approach, to define the measure function we aim to estimate the ROM approximation error by its bounds. Savvides et al. [2024] approximates the error upper bounds following the assumption that the underlying ground-truth function can be modeled based on a Gaussian Process (GP). We define a similar error-guided measure function and name the corresponding approach BayPOD-EAL. Considering $\tilde{f}_{\mathbf{p}_U x}^{(t)}$ as the model’s prediction for unlabeled input \mathbf{p}_U , at time t and position x , the primary objective is to estimate $(f_{\mathbf{p}_U x}^{(t)} - \tilde{f}_{\mathbf{p}_U x}^{(t)})^2$, where $f_{\mathbf{p}_U x}^{(t)}$ is the ground truth or FOM solutions. To estimate the error upper bound, Savvides et al. [2024] first considers that the ground-truth function f can be written as a GP with a symmetric, positive definite kernel. Based on this reformulation of f , conditioned on \mathbf{D}_L , the posterior distribution, \mathcal{F}' , over f , can be utilized to find $U^{(t)}(\mathbf{p}_U, x)$, the upper-bound of the expected posterior loss $L^{(t)}(\mathbf{p}_U, x) = E_{f \sim \mathcal{F}'}[(f_{\mathbf{p}_U x}^{(t)} - \tilde{f}_{\mathbf{p}_U x}^{(t)})^2]$ by further assuming that kernels are continuously twice differentiable and translation invariant and the governing function’s variance is bounded. Utilizing the model’s posterior, $q(\cdot | \mathbf{D}_L)$, and by representing the model’s prediction with its estimated mean via MC sampling, we set $M = U$ in (2) which is then used in (3) to find \mathbf{p}^* .

D Rod Temperature Evolution

Depending on the heat diffusivity parameter, κ , the heat diffusion over a rod is governed by:

$$\frac{\partial f}{\partial t} = \kappa \frac{\partial^2 f}{\partial x^2}, \quad (4)$$

for which the initial condition and Dirichlet boundary conditions in our experiments are set as $f(x = 0, t) = 3 \sin(2t)$, $f(x = L, t) = 3$, and $f(x, t = 0) = 0$, respectively. These constraints, are incorporated through the particular solution, \mathbf{f} . \mathbf{f} can be derived by solving the problem with boundary conditions $f(x = 0, t) = 0$ and $f(x = L, t) = 1$, and the problem with the boundary condition $f(x = 0, t) = 1$ and $f(x = L, t) = 0$, which we denote as the steady-state solution $\bar{\mathbf{f}}_L(x)$ and $\bar{\mathbf{f}}_0(x)$, respectively. Finally, the corresponding *particular solution* can be written as $\mathbf{f} = 3 \sin(2t) \bar{\mathbf{f}}_0(x) + 3 \bar{\mathbf{f}}_L(x)$ [Boluki et al., 2024]. This allows embedding physics when training BayPOD by learning based on the modified snapshots with homogeneous boundary conditions that are acquired by subtracting snapshots’ particular solutions from them. To accurately model dynamics with inhomogeneous boundary conditions, the final approximation is constructed by adding back the *particular solution*.

E Experiment Settings

In all our experiments, each snapshot is a n_x -dimensional vector, with $n_x = 200$, for a temperature field over the rod with diffusivity κ at specific time t . We consider evaluating BayPOD-AL with 90 equidistant values as diffusivity parameters in $[0.1, 0.9]$, K. For both AL algorithms, BayPOD-UAL and BayPOD-EAL, we prioritize corresponding diffusivity parameter values in K based on the experimental settings detailed below. During the AL process, by considering $n_t^P = 50$ fixed time points that are randomly chosen from 628 equidistant temporal points in $[0, 2\pi]$, which mimics the case that coarse-grained FOM snapshots are used for ROM learning to further improve computational efficiency. At each step of AL, the 50 new snapshots corresponding to previously chosen fixed time points with the selected informative diffusivity parameter (κ^*) are added to \mathbf{D}_L . The performance comparison experiments are based on random runs starting from a randomly chosen diffusive parameter value in K, with an initial set \mathbf{D}_L that contains 50 snapshots.

We focus on an extrapolation setting for performance evaluation with respect to input diffusive parameters, where the snapshots corresponding to the last 20 parameters in K are chosen as the test dataset. To further investigate the robustness of BayPOD-AL methodologies, we report the performance on two test datasets with different number of time points: 1) $n_t^P = 50$ (the same number of time points as in training, 1000 snapshots in total), and 2) $n_t^P = 200$ (4000 snapshots in total) randomly chosen from T , for which we refer to as ‘low’ and ‘high’ temporal resolution datasets

respectively. To account for the objective, the acquisition function (2) is calculated by considering the low-resolution setting ($n_t^P = 50$ per unlabeled input) for experiments on the low-resolution test dataset, and the high-resolution setting ($n_t^P = 200$ per unlabeled input) for experiments on the high-resolution test dataset. In this work, the dimension of POD bases is set to 7.

All the experiments are performed using 10 GB memory on a 40 GB A100 GPU with each experiment taking less than 7 hours.