Perspectives of Physics-Based Machine Learning Strategies for Geoscientific Applications Governed by Partial Differential Equations

Denise Degen[1], Daniel Caviedes Voullième[2,3], Susanne Buiter[4,5], Harrie-Jan Hendricks Franssen[3,6], Harry Vereecken[3,6], Ana González-Nicolás[2], and Florian Wellmann[1,7]

[1]RWTH Aachen University, Computational Geoscience, Geothermics and Reservoir Geophysics (CG3), Mathieustraße 30, 52074 Aachen, Germany
[2]Forschungszentrum Jülich GmbH, Jülich Supercomputing Centre (JSC), Simulation and Data Lab. Terrestrial Systems, Wilhelm-Johnen-Straße, 52425 Jülich, Germany
[3]Forschungszentrum Jülich GmbH, Agrosphere, IBG-3, Wilhelm-Johnen-Straße, 52425 Jülich, Germany
[4]RWTH Aachen University, Tectonics and Geodyamics (TAG), Lochnerstraße 4-20, 52064 Aachen, Germany
[5]Helmholtz Centre Potsdam – GFZ German Research Centre for Geosciences, Telegrafenberg, 14473 Potsdam, Germany
[6]Centre for High Performance Computing Terrestrial Systems, Geoverbund ABC/J, Wilhelm-Johnen-Straße, 52425 Jülich, Germany
[7]Fraunhofer Research Institution for Energy Infrastructures and Geothermal Systems (IEG), Am Hochschulcampus 1, 44801 Bochum, Germany

Correspondence: Denise Degen (denise.degen@cgre.rwth-aachen.de)

Abstract. An accurate assessment of the physical states of the Earth system is an essential component of many scientific, societal and economical considerations. These assessments are becoming an increasingly challenging computational task since we aim to resolve models with high resolutions in space and time, to consider complex coupled partial differential equations, and to estimate uncertainties, which often requires many realizations. Machine learning methods are becoming a very popular method for the construction of surrogate models to address these computational issues. However, they also face major challenges in producing explainable, scalable, interpretable and robust models. In this manuscript, we evaluate the perspectives of geoscience applications of physics-based machine learning, which combines physics-based and data-driven methods to overcome the limitations of each approach taken alone. Through three designated examples (from the fields of geothermal energy, geodynamics, and hydrology), we show that the non-intrusive reduced basis method as a physics-based machine learning approach is able to produce highly precise surrogate models that are explainable, scalable, interpretable, and robust.

1 Introduction

Applications in Geosciences dealing with land surface and subsurface environments cover a broad range of disciplines and Earth system compartments comprising lithosphere (including deep reservoirs), groundwater, soil and the land surface. Subsurface environment applications typically deal with the management of natural resources such as subsurface water stored in soils and aquifers (e.g., water supply, contamination), energy (e.g., geothermal energy, hydrocarbons) that require an in-depth understanding of the governing physical, chemical, and biological principles. In addition, subsurface environments are

heterogeneous across a variety of spatial scales and processes in these environments are typically complex, highly nonlinear, and time-dependent. Spatial scales may range from the nano-scale to the basin- and continental-scale. Despite the enormous variation in applications in terms of scale and processes, there are underlying common principles. Many of these processes are governed by partial differential equations (PDEs) and are embedded in continuum physics approaches (Bauer et al., 2021; Bergen et al., 2019). In this study, we focus on subsurface applications from the fields of geodynamics, hydrology, and geothermal energy to showcase various methods that are applied in applications which are characterized by data sparsity and difficult accessibility for sampling and measuring states and fluxes as well as for mapping the spatial organization of key subsurface properties.

We focus on continuum PDEs and examine how to accelerate the numerical solution of PDEs that arise from geoscientific applications combining methods from the field of projection-based model order reduction (Benner et al., 2015) and machine learning. Our decision to focus on the aspect of accelerating the solution of PDEs stems from the fact that this is essential in geoscientific applications that aim at improving the understanding of the Earth's systems with its associated physical processes. An improved understanding requires extensive parameter estimation studies, sensitivity analyses, and uncertainty quantification, which are computationally expensive for many geoscientific PDE-based models (Degen et al., 2022c). This problem is further increased by the circumstance that the governing PDEs are becoming more and more complex as our understanding of the physical processes and their coupling increases. As an example, geothermal studies moved from a purely thermal system to a coupled thermo-hydro-mechanical and depending on the scale also chemical system (e.g., Gelet et al., 2012; Jacquey and Cacace, 2020b; Kohl et al., 1995; Kolditz et al., 2012; O'Sullivan et al., 2001; Stefansson et al., 2020; Taron and Elsworth, 2009). Recent studies include more advanced considerations of the mechanical components, yielding nonlinear hyperbolic PDEs (e.g., Gelet et al., 2012; Jacquey and Cacace, 2017, 2020b, a; Poulet et al., 2014).

The more complex PDEs allow a better characterization of the physical processes in the subsurface. However, they also lead to several challenges:

Challenge 1: In geoscientific applications, one primary interest is to determine which parameters (e.g., rock and fluid properties) influence the state distribution (e.g., temperature, pressure, stress) to what extent. This question can be addressed through sensitivity analyses, which are categorized into local and global sensitivity analysis (SA) (Wainwright et al., 2014). Local SAs are computationally fast and require only a few simulations. However, they neither return information on the parameter correlation nor do they allow the exploration of the entirety of the parameter space (Wainwright et al., 2014; Sobol, 2001). Global SAs overcome these issues but are computationally challenging because they demand numerous simulations (Degen et al., 2021b; Degen and Cacace, 2021).

Challenge 2: Geoscientific and geophysical investigations are often subject to various sources of uncertainty. To assess these uncertainties, stochastic descriptions of the PDEs or probabilistic inverse methods such as Markov chain Monte Carlo are commonly employed. However, these methods have in common that they require solving many forward evaluations, yielding an even more computationally demanding problem (Rogelj and Knutti, 2016; Stewart and Lewis, 2017).

Challenge 3: Often, PDE-based systems are required for predictions and real-time applications (Bauer et al., 2021; Sabeena and Reddy, 2017). Hence, there is the need to solve PDEs fast. This imposes further difficulties since the simulations for fully

coupled nonlinear hyperbolic PDEs are too demanding to allow such fast applications. Therefore, they currently have only limited applicability in many geoscientific research areas (Bauer et al., 2021; Bergen et al., 2019; Willcox et al., 2021).

These challenges have one important aspect in common: they require multiple or fast simulations. If a single simulation already requires a long simulation time, then addressing these challenges becomes difficult to prohibit.

In this work, we focus on the utility of surrogate models as a way to speed-up PDE-based simulations of the physical state in geoscientific problems. In particular, we determine how surrogate models (sometimes also referred to as meta models) can be employed to obtain precise predictions at a reduced computational cost. With the increasing interest in machine learning and the availability of various physics-driven approaches, it becomes apparent that we need to clarify which requirements we have for our surrogate models. Machine learning has become very popular and powerful in application fields where a vast amount of data is available (Baker et al., 2019; Willcox et al., 2021). However, Baker et al. (2019) also stated in their report to the United States Department of Energy that machine learning models need to be scalable, interpretable, generalizable, and robust to be attractive for scientific applications, which is also stated in Willcox et al. (2021). This is not only important for scientific machine learning models but for surrogate models in general.

Requirement 1.1 Physical Scalability: The system resolutions are constantly increasing since one wants to resolve models with higher and higher resolutions in space and time (Bauer et al., 2021; van Zelst et al., 2022). This increase in numerical system size, which is, for instance, noticeable by the increase in degrees of freedom, yields a computationally demanding problem. As an example in geodynamic simulations, one is interested both in large and small-scale deformations simultaneously. For the time component, variations in the order of milliseconds (seismic cycle) and variations in the order of millions of years (basin and plate-scale geodynamics) are of interest. In hydrological problems, capturing the spatial small-scale heterogeneity of the surface and subsurface, the short-term nature of meteorological forcing (hours to days), and the long-term nature of subsurface fluxes (tens of years and longer) pose similar challenges (Blöschl and Sivapalan, 1995; Condon et al., 2021; Fan et al., 2019; Paniconi and Putti, 2015). This results in computational expensive problems (van Zelst et al., 2022), which often require high-performance computing (HPC) infrastructures even for solving a single forward problem.

Requirement 1.2 Performance Scalability: Porting applications to HPC infrastructures is another major requirement, which needs the rewrite of major parts of the software packages (Alexander et al., 2020; Artigues et al., 2019; Baig et al., 2020; Bauer et al., 2021; Bertagna et al., 2019; Lawrence et al., 2018; Gan et al., 2020; Grete et al., 2021; Hokkanen et al., 2021). Therefore, one important aspect is to evaluate how physics-based machine learning methods can help in the transition to modern HPC infrastructures.

Requirement 2 Interpretability: Another main requirement is to produce surrogate models that maintain the characteristic of PDE solutions and the original structure of the governing principles (i.e., that they map from model parameters to state information). Related to this is the consideration of nonlinear problems (Grepl, 2012; Hesthaven and Ubbiali, 2018; Reichstein et al., 2019; van Zelst et al., 2022), which, in the field of geosciences, is often related to the consideration of mechanical or complex flow effects. Such nonlinear problems result in extensive computation times because of extra iterations in the numerical solve (van Zelst et al., 2022).

Requirement 3 Generalizability: Furthermore, it is important that the approaches apply to various disciplines and are not restricted to single applications only. To enable a transfer between disciplines it is important to provide accessible solutions. Often software packages are developed in-house and are not available to the entire community.

Requirement 4 Robustness: The last requirement is robustness. It is critical that the solutions produced by the surrogate models are consistent for different levels of accuracy. As an example, it is not desired to have a surrogate model that predicts for a lower accuracy a diffusion-dominated solution and for a slightly increased accuracy an advection-dominated solution.

The paper is structured as follows: We present the PDEs considered in this study first (see Sec. 2). Afterwards, we present the concept of data-driven, physics-based approaches, and physics-based machine learning in the same section. In section 3, we present three designated case studies, from the field of geothermal energy, geodynamics, and hydrology, to illustrate the potential of physics-based machine learning. This is further emphasized in section 4, where we evaluate how physics-based machine learning can address various challenges commonly occurring in the field of geosciences. We conclude the paper in section 5.

2   Surrogate Modeling Methods

In the following, we introduce different surrogate model techniques to address the challenges presented in Section 4. Thereby, a focus lays on inverse applications, uncertainty quantification methods, global sensitivity analyses, and other methods concerned with parameter estimation.

Surrogate models can be constructed with physics- and data-driven approaches (Fig. 3.5a,b). Physics-driven approaches preserve the governing equations, meaning that they, for instance, conserve mass, momentum, and energy in the same way as the original discretized version does. They also maintain the characteristic that for a given set of model parameters (e.g., material properties), they produce information about the state variables (e.g., temperature, pressure) at, for instance, every node of the discretized model. But are limited in the complexity that they can capture (Fig. 3.5a). We here focus on hyperbolic and/or non-linear PDEs as these pose computational challenges in physics-based methods, as outlined above. We will not cover linear elliptic and linear parabolic PDEs for which we advice using pure physics-based approaches. These can provide error bounds in addition to the surrogate model allowing to obtain objective guarantees of the approximation quality. In Tab. 1, we list the PDEs used throughout the manuscript and their respective classes. Within the physics-driven approaches, we

Table 1. List of partial differential equations and their respective classes used throughout the manuscript.

PDE	Class
Heat Transfer Equation	nonlinear, hyperbolic
Heat Transfer Equation with shear heating	nonlinear, hyperbolic
time-dependent Richard's equation	nonlinear, elliptic-parabolic

focus on techniques that allow a retrieval of the entire state. Projection-based model order reduction techniques, such as the Proper Orthogonal Decomposition (POD), balanced truncation, and the reduced basis method, belong to this class (Benner

et al., 2015). We discuss physics-based approaches in detail in Sect. 2.1. Machine learning techniques are a common example of data-driven approaches (Baker et al., 2019; Bauer et al., 2021; Willcox et al., 2021). They have the advantage of capturing

115   the full complexity of the model at hand but loose the information about the physical system (Fig. 3.5b). Examples of other data-driven techniques are Kriging (Gaussian Processes), the polynomial chaos expansion, and response surfaces (e.g., Miao et al., 2019; Mo et al., 2019; Navarro et al., 2018). Data-driven methods are described in Sect. 2.2. The last category is physics-based machine learning, which combines aspects of both physics-based and data-driven techniques to overcome the limitations of the individual approaches (Fig. 3.5c). A detailed overview of this technique is provided in Sect. 2.3.
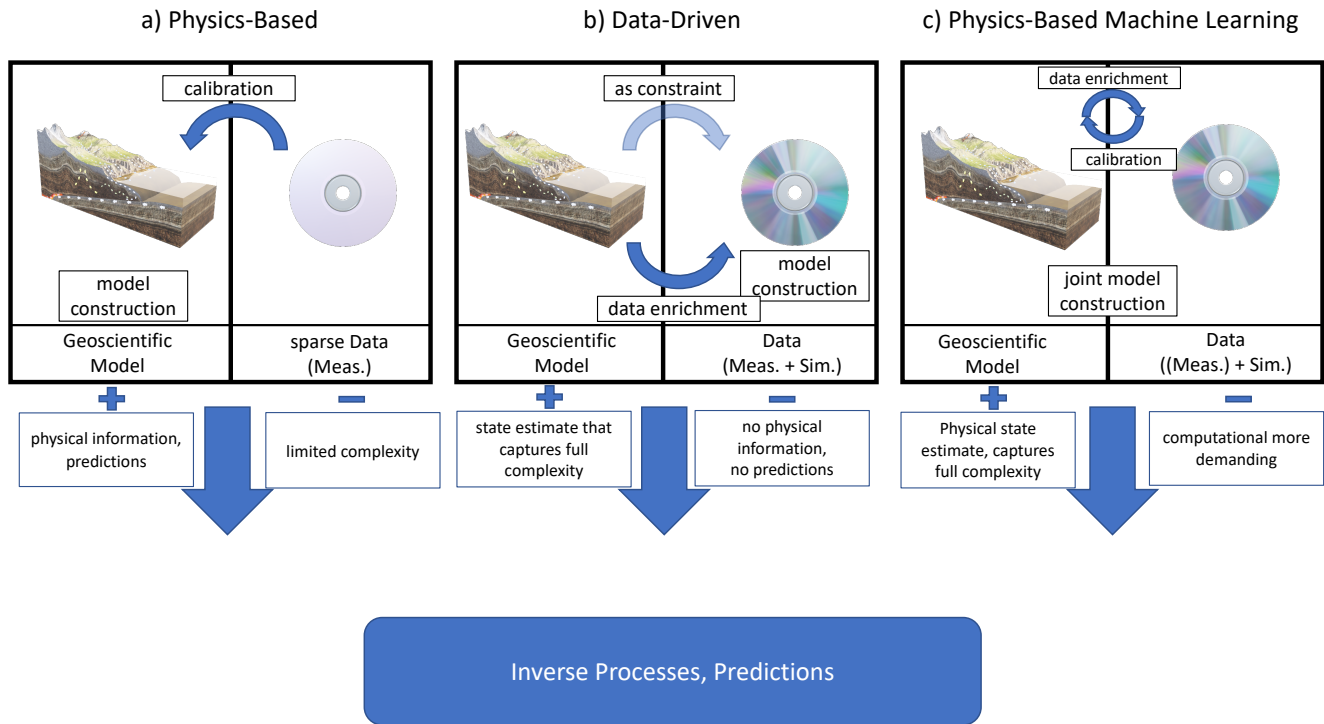
Figure 1. Schematic comparison of physics-based, data-driven, and physics-based machine learning methods for the construction of surrogate models. Meas. = measured (e.g., well data, surface observations) and Sim. = simulated. The boxes with a + list advantages, the boxes with a - list disadvantages of the method above.

120   ## 2.1   Physics-Based Surrogate Modeling / Model Order Reduction

Model order reduction (also called reduced order modeling) has been extensively used in various application fields, such as groundwater (Ghasemi and Gildin, 2016; Gosses et al., 2018; Rousset et al., 2014) and thermal studies (Rizzo et al., 2017). The idea is to find a low-dimensional representation of the original high-dimensional problem, while maintaining the input-output relationship. This low-dimensional representation allows a fast computation, enabling, in turn, probabilistic inverse methods

and other forward-intensive algorithms. Commonly known methods of projection-based model order reduction techniques are the Proper Orthogonal Decomposition (POD) (Benner et al., 2015; Hesthaven et al., 2016; Volkwein, 2011), the reduced basis (RB) method (Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015), and balanced truncation (Antoulas, 2005). In the following, we will briefly introduce the POD and RB methods since they are relevant for the remaining case studies. For a more detailed overview of projection-based model order reduction, we refer to Benner et al. (2015).

POD is the probably most widely used projection-based model order reduction technique. For the construction of the reduced model, we first calculate snapshots. Snapshots are state distributions for different parameters (e.g., rock properties) and/or different time steps. Afterwards, we perform a singular value decomposition. The reduced basis is obtained by keeping the singular vectors corresponding to the largest singular values. The resulting orthonormal POD basis is turned into the reduced model through a Galerkin projection (Benner et al., 2015; Hesthaven et al., 2016; Volkwein, 2011). We categorize the POD as a physics-based method since through the projection the PDE is translated from the space of the spatial coordinates to the material properties. In the case that all POD modes would be used this leads to an equivalent representation. The Galerkin projection needs access to the stiffness matrix and load vector and hence direct access to the discretized version of the PDE. Note that a POD can also be directly applied to, for instance, image data that has not been constructed by PDEs. However, in this case, no Galerkin projection would be applied. In the Geoscience community, the POD method has been used in, for instance, thermal applications (Rousset et al., 2014) and groundwater studies (e.g., Ghasemi and Gildin, 2016; Gosses et al., 2018; Rizzo et al., 2017; Vermeulen et al., 2004; Vermeulen and Heemink, 2006).

Another physics-based method is the RB method. The idea of RB is similar to the principle idea of POD. The main difference is the construction of the reduced basis. In RB the selection of the snapshots is often performed via a greedy algorithm as the sampling strategy. The greedy algorithm chooses at each iteration, the solution that adds the highest information content (Bertsimas and Tsitsiklis, 1997; Veroy et al., 2003). Afterwards, the orthonormalized snapshots are added to the reduced basis and the reduced model is again obtained via a Galerkin projection (Benner et al., 2015; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015). Since this version of the RB method does not calculate the snapshots in advance but calculates them "on-the-fly" it performs slightly better for higher-dimensional parameter spaces (Jung, 2012). However, for the efficient construction of the reduced basis, an error bound or estimator is required. Currently, no efficient error bounds for hyperbolic PDEs exist, limiting the applicability of the method. The RB method is a physics-based method since, as described for the POD above, it is a projection method, and the Galerkin method requires the discretized version of the PDE.

A note of caution: the combination of the POD and the Galerkin projection step is sometimes referred as POD (Swischuk et al., 2019) only and sometimes as POD-Galerkin (Busto et al., 2020).

Both the POD and the RB method have limitations when considering complex geophysical problems. Both methods aim at an affine decomposable problem, meaning that our problem is decomposable into a parameter-dependent and -independent part. This requirement is naturally given for linear but not for nonlinear problems. Methods such as the empirical interpolation method (Barrault et al., 2004) exist to extend the methods for the nonlinear case, but their performance for high-order nonlinearities is limited.

Our choice to mostly rely on physics-based methods stems from the reason that understanding the physical processes is a central component in Geosciences. Physics-driven methods are ideal for predictions and risk assessments but they also have the major disadvantage of being computationally too demanding for complex coupled subsurface processes. In literature, this can often be observed indirectly. For example, in geothermal applications basin-scale applications usually consider a hydrothermal system, whereas reservoir-scale applications also incorporate the mechanical and/or the chemical processes (e.g., Freymark et al., 2019; Gelet et al., 2012; Jacquey and Cacace, 2017, 2020b, a; Kohl et al., 1995; Kolditz et al., 2012; Koltzer et al., 2019; O'Sullivan et al., 2001; Poulet et al., 2014; Stefansson et al., 2020; Taron and Elsworth, 2009; Taron et al., 2009; Vallier et al., 2018; Wellmann and Reid, 2014). Another example is magnetotellurics (MT). The inversion of MT data is very costly, hence; most inversions are performed in 1D or 2D (e.g., Chen et al., 2012; Conway et al., 2018; Jones et al., 2017; Rosas-Carbajal et al., 2014). 3D MT inversion is even with HPC infrastructures challenging (Rochlitz et al., 2019; Schaa et al., 2016). How to accelerate the solve of the Maxwell equations through the reduced-basis method and simultaneously achieve efficient error estimates or bounds have been shown fundamentally in e.g., Chen et al. (2010), Hammerschmidt et al. (2015), Hess and Benner (2013a), Hess and Benner (2013b), and Hess et al. (2015). The potential of the reduced-basis method is well illustrated in Manassero et al. (2020), where the authors perform an adaptive Markov chain Monte Carlo run for an MT application using the reduced-basis method.

2.2 Data-Driven Techniques

In applications where we have a huge amount of data (e.g., earthquake and seismic applications) and only limited knowledge of the physical processes, data-driven approaches such as machine learning are superior to purely physics-based methods (Bergen et al., 2019; Raissi et al., 2019). Machine learning approaches are powerful in discovering low-dimensional patterns and in reducing the dimensionality of the model. Hence, they would be ideal for surrogate models. A review of the state and potential of machine learning techniques for Solid Earth is provided in Bergen et al. (2019) focusing on data-rich applications such as the analyses of earthquakes. The applied methods in this field compromise, for instance, (deep) neural networks, random forest, decision trees, and support vector machines. However, open questions about the reliability, the explainability of the models, the robustness, and the rigorosity of how much the underlying assumptions have been tested and validated remain (Baker et al., 2019; Willcox et al., 2021). This is critical in applications such as risk assessment, where we need to provide guarantees and confidence intervals for our predictions. Another issue is that data-driven approaches such as machine learning are problematic in terms of robustness and have no proven convergence if applied to the so-called "small data" regime (Raissi et al., 2019). Raissi et al. (2019) define the small data regime, as a regime where the collection of data becomes too expensive and thus conclusion have to be drawn on incomplete data sets.

This is especially critical for the application field of Solid Earth, where we classically are concerned with a lack of data. Hence, this field does not allow straightforward applicability of data-driven approaches. Furthermore, in this field, we need to perform predictions and risk assessments. Therefore, black-box approaches, such as machine learning, pose a major challenge in how to provide guarantees and how to obtain interpretable models (Baker et al., 2019; Willcox et al., 2021).

Some data-driven methods assume that we are interested in a quantity of interest and not in the entire state distribution. Possible quantities of interest are the value of the state variable at a certain location in space or the amount of substance that is accumulated over time in a certain region. Note that this class of methods allows a violation of the governing conservation laws since it commonly uses statistical methods for the construction of the surrogate model (Bezerra et al., 2008; Cressie, 1990; Khuri and Mukhopadhyay, 2010). Typical methods used in the field of Geosciences are Kriging, polynomial chaos expansion, and response surfaces (Baş and Boyacı, 2007; Bezerra et al., 2008; Frangos et al., 2010; Khuri and Mukhopadhyay, 2010; Miao et al., 2019; Mo et al., 2019; Myers et al., 2016; Navarro et al., 2018). We will not further discuss this class of methods since in this paper we focus on applications that are not only interested in a quantity of interest but also in the entire state distribution. For an overview of data-driven machine learning techniques, we refer to Jordan and Mitchell (2015); Kotsiantis et al. (2007); Mahesh (2020).

2.3 Physics-Based Machine Learning

Data-driven methods are a non-ideal choice in the Earth Sciences because they do not provide an understanding of the underlying subsurface processes (Reichstein et al., 2019). Furthermore, they require more data than physics-based methods (Willcox et al., 2021), which is relevant because some applications fields such as geodynamics and geothermal applications are characterized by data sparsity. However, physics-based methods are also not applicable for complex coupled processes since they do not capture the full complexity of the problem.

This is the reason why we introduce the concept of physics-based machine learning. Note that we discuss the potential of these methods for applications where primarily physical knowledge is available (in form of PDEs) and very sparse to no data. The discussion would be different in a data-dominated application. Physics-based machine learning uses a combination of data-driven and physics-based techniques, as schematically shown in Fig. 3.5. For solving PDEs Willard et al. (2020) distinguish various techniques, comprising the categories of physics-guided loss functions, physics-guided architectures, and hybrid models. These various approaches are already compared in several papers (e.g., Faroughi et al., 2022; Swischuk et al., 2019; Willard et al., 2020). However, they focus on applications were substantially more data is available than in most subsurface applications. Therefore, we shift the focus to application with very sparse data sets. This impacts the potential of physics-based machine learning methods differently, depending on the paradigm used to combine physics- and data-based methods. So, instead of presenting numerous of these techniques in detail, as already done in the afore mentioned papers, we want to discuss the different paradigms that exist. Here, we identify two end-member cases: i) physics-guided loss functions, and ii) hybrid models.

In physics-guided loss function methods, the idea is to use a data-driven approach and incorporate the physics in the loss function to force the system to fulfill the physical principles. They have a better performance for the small data regime since the physical constraints regularize the system and they constrain the size of admissible solutions (Raissi et al., 2019). Within the field of physics-guided loss functions various different approaches are available (e.g., Bar-Sinai et al., 2019; De Bézenac et al., 2019; Dwivedi and Srinivasan, 2020; Geneva and Zabaras, 2020; Karumuri et al., 2020; Meng and Karniadakis, 2020; Meng et al., 2020; Pang et al., 2019, 2020; Peng et al., 2020; Raissi et al., 2019; Shah et al., 2019; Sharma et al., 2018; Wu

et al., 2020; Yang et al., 2018; Yang and Perdikaris, 2018; Yang et al., 2020; Zhu et al., 2019). For the following, we as an example focus on the method of physics-informed neural networks (PINNs).

The second class of physics-based machine learning techniques that we present in more detail are hybrid models. In this technique, the physics is not added as a constraint in the loss function but instead, we perform both a physics-based and a data-driven approach. Often the physics-based method is executed first and the output serves as an input to the machine learning approach (Willard et al., 2020; Hesthaven and Ubbiali, 2018; Swischuk et al., 2019). Also, for this class of methods, various methods are available (Hesthaven and Ubbiali, 2018; Malek and Beidokhti, 2006; Swischuk et al., 2019; Wang et al., 2019), here we focus on the non-intrusive RB method.

Note that both methods act as examples to better illustrate the different concepts used for combining physics- and data-based approaches. Numerous other methods such as the Physics-Encoded Neural Network (PeNN) (Chen et al., 2018a; Li et al., 2020; Rao et al., 2021), Physics-Encoded Recurrent-Convolutional Neural Network (PeRCNN) (Rao et al., 2021), the Fourier Neural Operator (FNO) (Li et al., 2020), and DeepONets (Lu et al., 2021). We will not discuss in detail but shortly explain their relations to the two paradigms, represented by PINNs and the non-intrusive RB method.

2.3.1 Physics-guided loss functions – Physics-informed neural networks

PINNs overcome the shortcomings of machine learning techniques by including physics into the system. The principle idea is that in applications where we lack data we often have other sources of information, including domain expert knowledge and physical laws (e.g., conservation and empirical laws). By including these additional sources into machine learning the problem is regularized and the set of admissible solutions greatly reduced, yielding significantly improved performance of the algorithm (Raissi et al., 2019). The idea of including physical knowledge is well known for Gaussian Processes. However, Gaussian Processes have only limited applicability for nonlinear problems (Raissi et al., 2017; Raissi and Karniadakis, 2018). Therefore, PINNs rely on neural networks which are known for their good performance in the estimation of nonlinear functions (Raissi et al., 2019).

In Figure 2, we show the setup of a PINN schematically. The first step is to create a (deep) neural network with space and time as inputs. The network generates a model for the state variable u. Through automatic differentiation, we obtain the spatial and temporal derivatives, which, in turn, are used in the calculation of the PDE. We learn the shared parameters of the network and the PDE by minimizing a joint loss function as presented in Equation 1 (Raissi et al., 2019):

$$MSE = MSE_u + MSE_f,$$

$$\text{where} \quad MSE_u = \frac{1}{N_u} \sum_{i=1}^{N_u} |u\left(t_u^i, x_u^i\right) - u^i|^2,$$

$$\text{and} \quad MSE_f = \frac{1}{N_f} \sum_{i=1}^{N_f} |f\left(t_f^i, x_f^i\right)|^2. \tag{1}$$

Here, MSE stands for mean squared error, $u(t,x)$ denote the variable with the training data $\{t_u^i, x_u^i, u^i\}_{i=1}^{N_u}$. $\{t_f^i, x_f^i,\}_{i=1}^{N_f}$ are the collocation points for our PDE $f(x,t)$. The subscript $_u$ refers to the constrains for the boundary and initial conditions,
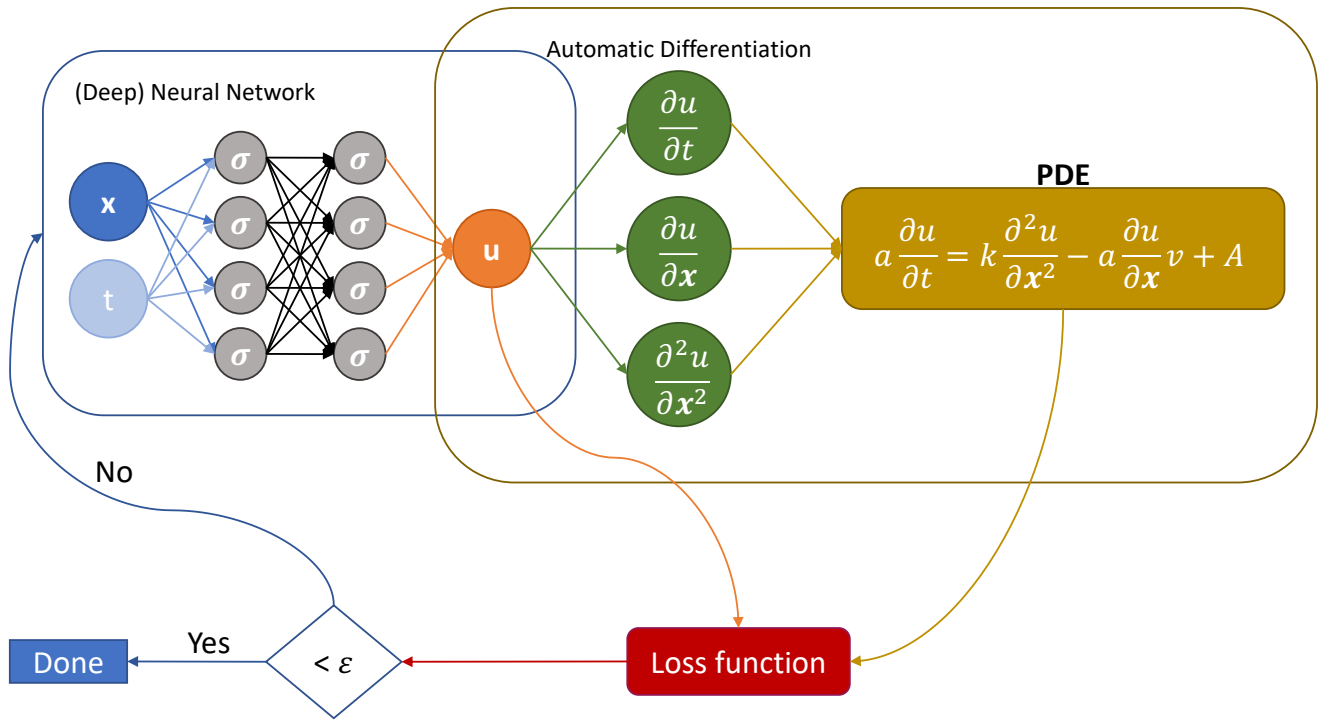
Figure 2. Schematic representation of a physics-informed neural network.

and the subscript $_f$ to the PDE constrains. Consequently, the MSE_u minimizes the difference between the data u and the simulated data at the collocation points (for the boundary and initial conditions). The MSE_f enforces the structure of the PDE at selected collocation points. Through the terms $\frac{1}{N_u}$ and $\frac{1}{N_f}$, we can weight the components of the loss function depending on, for instance, the amount of available data and physical knowledge.

PINNs are frequently used due to their high flexibility and straightforward implementation. They are used as a state estimation technique since they can recover the entire state from a given set of observations. PINNs face problems in high-frequency and multi-scale feature scenarios (Wang et al., 2022). Further disadvantages of PINNs are that the PDE is used as a constrain (among other constrains) and is enforced at a limited number of selected points only. Chuang and Barba (2022) point out that PINNs have also disadvantages concerning their computational efficiency. Due to the usage of automatic differentiation many residuals need to be evaluated leading to high computational costs. Additionally, in their tested computational fluid dynamics applications, losses close to zero needed to be minimized, yielding performance and convergences issues. This problem is also expected for a wide range of geoscientific applications. Lastly, Chuang and Barba (2022) reported that PINNs did not reproduce the desired physical solution in all of their benchmark examples. Therefore, the applicability of PINNs should be investigated prior to their use.

Numerous variations of PINNs are available, for which we provide an overview in the following:

- B-PINNs (provide an build-in uncertainty quantification, (Yang et al., 2020)),

- fPINNs (for fractional PDEs, (Pang et al., 2019)),

- nPINNs (for a nonlocal universal Laplace operator, (Pang et al., 2020)),

- PPINNs (offer a time parallelization, (Meng et al., 2020)),

- sPINNs (for stochastic PDEs, (Chen et al., 2019)),

- XPINNs (offer an efficient space-time decomposition, (Jagtap and Karniadakis, 2020)).

PINNs directly provide an estimate of the state and do not preserve the input-output relationship (e.g., from model parameters to state information) of our original physical problem. Hence, we no longer have a model that takes rock and fluid properties as input and produces estimates of, for instance, the temperature or pressure as our output. Additionally, PINNs assume that observation data might be sparse but are available. In many geoscientific applications, we have an extremely sparse data set and sometimes no direct observation data. Just as an example, if we want to investigate the formation of a sedimentary basin, we have only direct measurements for the present state. For the past, we have at best indirect measurements at hand. Consequently, we do not have enough data to utilize the PINN. In order to compensate, an enrichment of the data set by, for instance, numerical simulations is needed. However, these numerical simulations are the results of PDEs. So, the natural question would be why we do not use a method that builds upon the principles of PDEs rather than using the PDEs as a constraint. Therefore, we do not see benefits in employing PINNs for applications that have nearly no data, which is the case for the geothermal and geodynamic applications we consider in this paper. For subsurface hydrological data the situation is better since exhaustive data for the land surface is available. However, the data is commonly of a low quality (e.g., coarse, with systematic biases, measurement gaps). This yields similar problems for the PINNs as in the "no data" case. Hence, we focus in the remaining part of the paper on another physics-based machine learning technique, namely the non-intrusive RB method, which we present in the next section.

2.3.2 Hybrid Methods – Non-Intrusive Reduced Basis Method

The non-intrusive RB method originates from the model order reduction community. Model order reduction and machine learning have huge similarities. For example, the Proper Orthogonal Decomposition is very closely connected to the principal component analysis (Swischuk et al., 2019). Swischuk et al. (2019) pointed out that the difference between the methods has historical reasons. Model order reduction methods have been developed by the scientific computing community to reduce the high-dimensional character of typical scientific applications. In contrast, machine learning originates from the computer science community aiming at generating low-dimensional representations through black-box approaches (Swischuk et al., 2019).

The first method that we presented, physics-informed neural networks, can be seen as a machine-learning method, where we use physics to constrain our system. On the other hand, the non-intrusive RB approach can be seen as a modification of the

model order reduction techniques of POD and RB using data-driven approaches. The major difference between the methods is in the question of what we learn from the algorithm. For PINNs, we are learning the operators themselves. In contrast, the non-intrusive RB method derives a mapping between the input parameters (e.g., rock properties) and the output (e.g., temperature and pressure distribution) (Swischuk et al., 2019).

The idea behind model order reduction is that the reduced solution $u_L(\boldsymbol{x}; \boldsymbol{\mu})$ can be expressed as the sum of the product of the basis functions $\psi(\boldsymbol{x})$ and the reduced coefficients $u_{\mathrm{rb}}(\boldsymbol{\mu})$. Here, $\boldsymbol{x}$ defines the spatial coordinates and $\boldsymbol{\mu}$ the parameters. The procedure is described in Eq. 2 (Hesthaven and Ubbiali, 2018):

$$u_L(\boldsymbol{x}; \boldsymbol{\mu}) = \sum_{l=1}^{L} u_{\mathrm{rb}}^{(l)}(\boldsymbol{\mu}) \psi_l(\boldsymbol{x}) \tag{2}$$

So, for the construction of the surrogate model, two steps are required. First the basis functions are determined, and afterwards, the reduced coefficients conventionally by a Galerkin projection. In Section 2.1, we presented two projection-based model order reduction techniques. The disadvantage of the presented methods is that they have only limited applicability to nonlinear PDEs (Hesthaven and Ubbiali, 2018; Wang et al., 2019). Furthermore, they require access to the decomposed stiffness matrix and load vector. However, this retrieval is sometimes prevented by the forward solver (i.e., in commercial software packages) (Peherstorfer and Willcox, 2016a). Both the limitation regarding the nonlinearity and the software are arising from this intrusive Galerkin projection.

To overcome the problem with the projection recent advances replace the intrusive projection with a non-intrusive approximation (Hesthaven and Ubbiali, 2018; Wang et al., 2019). Hesthaven and Ubbiali (2018); Wang et al. (2019) propose a (deep) neural network for learning the reduced coefficient. Other approaches including interpolation and machine learning methods have been explored for this purpose (Swischuk et al., 2019). Swischuk et al. (2019) compare four machine learning methods, namely neural networks, multivariate polynomial regression, k-nearest neighbors, and decision trees, for the determination of the reduced coefficients in aerodynamic and structural applications. Audouze et al. (2009, 2013); Wirtz et al. (2015) use Gaussian process regression to determine the reduced coefficients. Another strategy is proposed in Mainini and Willcox (2015), where the reduced coefficients are learned by an adaptive mixture of self-organizing maps and response surfaces.

Due to the different methods for obtaining the reduced coefficients, the commonly used greedy sampling strategy of the intrusive RB method (Hesthaven et al., 2016; Veroy et al., 2003) has to be modified. For the selection of basis functions, we can either use a POD (Hesthaven and Ubbiali, 2018; Wang et al., 2019) or a modified greedy sampling strategy (Chen et al., 2018b). In the case of the POD, which is used in this manuscript, the snapshots (i.e., simulation results for different model parameters) serve as an input and the basis functions as an output. For the second stage, the projection is via a machine learning method. The labels are the different model parameters (e.g., permeability, thermal conductivity), the training data the matrix product of the snapshots and the basis functions, and the outputs are the reduced coefficients. Taking the matrix product of snapshots and basis functions means that the dimension of spatial degrees of freedom does not enter into the training of the machine learning model, which has great implications regarding the cost of training these models, as we will entail in Section 3.5.
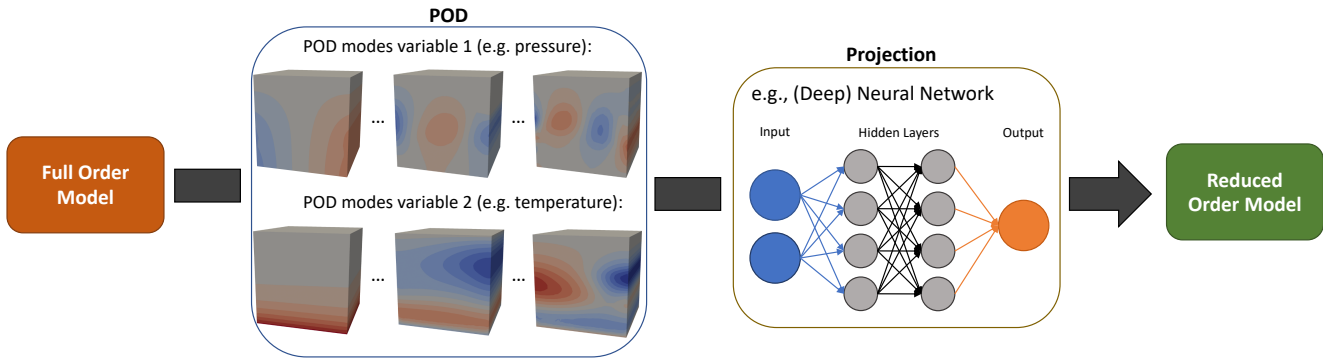
Figure 3. Schematic representation of the non-intrusive reduced basis method with a Proper Orthogonal Decomposition sampling strategy.

Note that non-intrusive model order reduction is not restricted to the inference of the output states via the input parameters. As an example, Peherstorfer and Willcox (2016a) developed a data-driven operator inference non-intrusive method. However, this method is restricted to low-dimensional nonlinearities, due to computational reasons (Peherstorfer and Willcox, 2016a). As we aim to demonstrate the potential of physics-based machine learning, we chose as an example two techniques: one closer to the community of computer science and the other closer to the community of scientific computing. We do not aim to provide an overview of all possible methods.

At the end, again a note of caution with regard to the terminology. The non-intrusive RB method consist of a sampling and projection step. If we use, for instance, t he POD as the sampling method and the NN as the projection method, than this is often referred to as POD-NN (Hesthaven and Ubbiali, 2018) instead of non-intrusive RB. In this manuscript, we use various machine learning techniques for the projection step. Since this has no general impact on the applicability of physics-based machine learning we use the more general term of non-intrusive RB to avoid unnecessary confusion.

2.3.3 Other Physics-Based Machine Learning Methods

As discussed before, many physics-based machine learning methods are available, all with their own advantages and disadvantages. Nonetheless, we can identify generally different approaches on how to integrate physics into data-driven approaches, see also Willcox et al. (2021). These diverse approaches yield different potential for addressing current challenges faced in subsurface applications, as we will present in Section 4. So, in the following, we briefly present other physics-based machine learning methods and how they relate to the two in detail presented paradigms.

The idea behind PeNNs is to hard encode the prior physical knowledge in the architecture of neural networks. This can be achieved through various methods (Chen et al., 2018a; Li et al., 2020; Rao et al., 2021). One possibility is the PeRCNN, where an optional physics convolution layer is implemented. This means the method is in between PINNs and the non-intrusive RB method. It is able to overcome some common problems encountered with PINNs since it does not implement physics as one of possibly many constraints. However, the method is still a statistical interpolation method and therefore cannot guarantee a

preservation of the physics. Regarding the categories that are used in Willard et al. (2020), this method would fall under the category of physics-guided architectures.

Another method that is associated with PeNNs are FNOs (Faroughi et al., 2022; Li et al., 2020). This method uses Fourier layers instead of the typical layers used in a neural network. Conceptually this is close to the non-intrusive RB method. In the Fourier layers the data is decomposed, similar to the POD step of the non-intrusive RB method. However, the main difference is that FNO is limited to sine and cosine functions. Neural operators are known to require large training data sets (Lu et al., 2022), which is critical in the applications presented here since the generation of the data is often very costly. Furthermore, they are limited when it comes to complex 3D problems (Lu et al., 2022). Another difference to the non-intrusive RB method is that again the approach is embedded into a data-driven scheme. Something similar is the case for DeepONets (Lu et al., 2021), which overcome the requirement of structured data that FNOs have, but show conceptually similar limitations as FNOs.

2.3.4 Differences in Physics-Based Machine Learning

We here provide examples from the field of hydrology and water resources management since machine learning is currently actively studied in these research fields (Shen et al., 2021; Sit et al., 2020). This is fueled by the promise of ML to unravel information and provide hydrological insights (Nearing et al., 2021) in what can be considered increasingly rich(er) datasets, especially for surface hydrology (in comparison to subsurface hydrology and other subsurface geoscientific applications, such as geodynamics). Typical applications of ML include runoff and flood forecasting, drought forecasting, water quality, subsurface flows, among many others (see Ardabili et al., 2020; Bentivoglio et al., 2021; Mohammadi, 2021; Sit et al., 2020; Zounemat-Kermnai et al., 2021, for extended reviews on the topic). Many of these applications are based on artificial neural networks (ANN) and deep learning toolsets which have become accessible to hydrologists thanks to the arrival of well-established, and GPU-enabled libraries. It is currently clear that ML offers a wide range of possibilities in hydrology, such as faster-than-real-time forecasting, extracting information from citizen data, parameter inversion, and uncertainty quantification. The idea of ML surrogates is increasingly exploited in hydrological research (e.g Bermúdez et al., 2018; Liang et al., 2019; Tran et al., 2021; Zahura et al., 2020; Zanchetta and Coulibaly, 2022).

Physics-based ML is still rare and incipient in hydrology, with efforts largely motivated by overcoming the black-box nature of traditional ML techniques (Jiang et al., 2020; Young et al., 2017) and the need to include physical constraints and theory (Nearing et al., 2021). Both the techniques to inform ML of the physics, as well as how much physics is embedded into the problem depend on the application and the underlying deterministic model, ranging from ordinary partial differential equations (ODE) (Jiang et al., 2020), classical lumped cascade hydrological models (Young et al., 2017), subsurface hydraulic parameter estimation based on nonlinear diffusion equations (Tartakovsky et al., 2020), Darcy equation (He et al., 2020) and Richards equation (Bandai and Ghezzehei, 2021), and overland flow (Maxwell et al., 2021). It is therefore clear that there is a growing momentum and clear potential gains in exploring Physics-based ML in hydrological sciences, motivated by methodological, theoretical, and computational reasons.

The term physics-based machine learning is used in an increasing number of scientific works however, the methodologies employed vastly differ. Therefore, we would like to briefly elaborate on the differences between these techniques. Note that this

explanation is purely meant to illustrate the usage of the term physics-based machine learning throughout the remaining part of this paper. To our knowledge, there is no unique definition of that term. Furthermore, we will only illustrate the differences between the methodologies presented here. We do not aim to provide a complete overview of all physics-based ML techniques but rather illustrate the various concept associated with the technique in general.

What might be sometimes defined as "physics-informed/based" approaches, are applications, where the surrogate model is constructed with classical data-driven ML techniques but the data originates from a physical model instead of observation data. In this paper, the definition of physics-based ML is purely based on the methodology. Since the former application is methodology-wise the same as a data-driven approach this would be defined here as a data-driven and not a physics-based ML technique.

The two physics-based ML techniques presented here, are PINNs and the non-intrusive RB method. Both methods have similar aims but are vastly different in the way where and how ML techniques and physics are utilized. PINNs use the physics as an additional constrain in the loss function and does not preserve the original structure of the problem. In contrast, the non-intrusive RB method extracts the main physical characteristics and weights them using a ML technique, while maintaining the input-output relationship.

3 Benchmark Studies

Combining both data- and physics-driven techniques allows for robust and reliable (surrogate) models that enable probabilistic inverse processes, which yield a significant improvement in the understanding of the system Earth. For an illustration of the potential that physics-based machine learning poses for the community, we present in the following three benchmark studies from the field of geothermal, geodynamical, and hydrological applications. Afterwards, we use these studies to illustrate how physics-based machine learning can help to overcome the challenges introduced at the beginning of the paper.

3.1 General Workflow

Before we present three case studies in the next section, we present a general workflow (Fig. 4) and point out some common pitfalls for the data-driven part of the methodology. Although the procedure is developed with the projection part of the non-intrusive RB method in mind, it applies to a wide range of data-driven approaches. For the following workflow, we assume that a geometrical and physical model is already available. Hence, we assume that the model is already tested for single forward simulations.

1. At the beginning of the procedure, we need to define the training and validation set. For its generation, several aspects have to be taken into account.

    (a) The sampling strategy is crucial to ensure both a small and reliable training set. We recommend using the Latin hypercube sampling method. Regular sampling schemes with equal spacings should be used with caution. They bear the danger of having always the same ratio between the various input parameters. In this case, the predictions

 will only be accurate if that ratio is maintained. To ensure that the sampling method is not negatively impacting the prediction, we strongly recommend using a different sampling strategy for the validation and training data set. In our studies, we use randomly chosen parameter combinations for the validation data set.

(b) Another important aspect is the size of the training and validation set. The size of the training data set cannot be determined in general since it is dependent on the underlying complexity of the parameter space. However, as a control check, the training data set needs to be significantly larger than the number of basis functions obtained via the POD method. The size of the validation data set should be at least 10 % of the training data set.

2. After generating the training and validation data set and before applying the physics-based machine learning method, it is important to preprocess the data. The preprocessing involves the following steps:

(a) The scaling of the input parameters. Common scaling methods are the normal and the min-max transform. In this study, we use normal transformed parameters. This means that the mean of each of the input parameters is zero and that they have a standard deviation of 1.

(b) Also crucial is the preprocessing of the actual training/validation data. This consists of two steps:

i. The first step is the removal of static effects such as the initial conditions.As an example, we often have an initial temperature distribution that is a magnitude larger, than the changes induced by the varying rock properties. In order to capture the changes induced by the rock properties it is important to remove this "masking" effect.

ii. The other important step is the scaling of the data. Again, common scaling methods are the normal and the min-max transform. In this paper, we employed, where necessary, the min-max transform.

3. After preparing the data, we can perform the physics-based machine learning method. When using methods, such as neural networks, the search for optimal hyperparameters (parameters that determine the architecture of the machine learning model) can be extremely time-consuming. This is a cost that should not be underestimated since it can prolong the construction time of the surrogate model significantly, which majorly impacts the efficiency of the concept. To considerably speed up the hyperparameter search, we advise using a Bayesian optimization scheme with hyperbands (BOHB) (Falkner et al., 2018). The idea behind BOHB is to accelerate the convergence rate of the Bayesian optimization for the tuning of the hyperparameters with the hyperbands. Furthermore, the methodology yields a scalable approach suited for parallel computing. For further details, we refer to Falkner et al. (2018).

3.2 Geothermal Example

For the geothermal benchmark study, we consider a three-dimensional model with three equally spaced horizontal layers (Fig. 5). The problem is non-dimensional, with an extent of 1 in x-, y-, and z-direction. The spatial mesh is a tetrahedral mesh with 11,269 nodes.
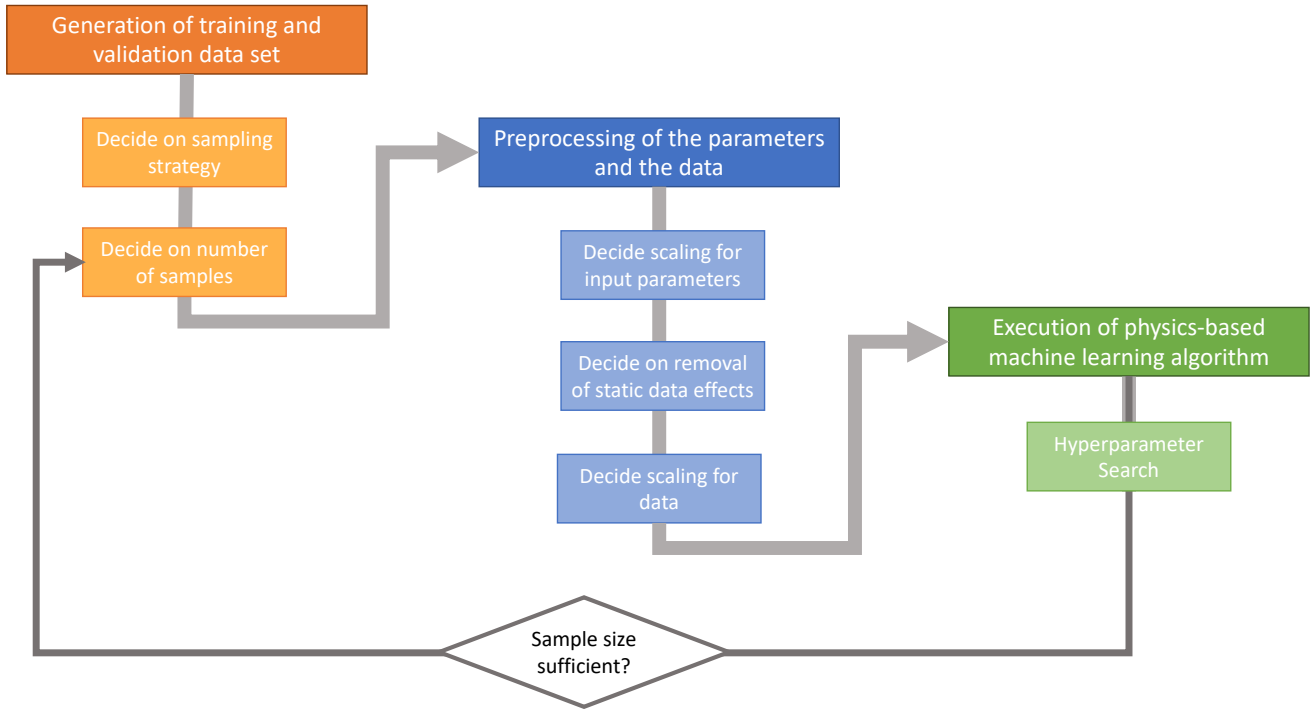
Figure 4. Schematic representation of the workflow to efficiently generate surrogate models by using physics-based machine learning techniques.

As the physical model, we take a fully coupled hydrothermal simulation into account, for which we consider the heat transfer equation (Nield and Bejan, 2017):

$$
\begin{aligned}
\frac{\partial\left((\rho c)_m T\right)}{\partial t} &= \nabla \cdot \left(\lambda \nabla T - (\rho c T)_f \, v\right) + H, \\
\text{with} \quad (\rho c)_m &= (1-\phi)\,\rho_s c_s + \phi \rho_f c_f, \\
\text{and} \quad \lambda_m &= (1-\phi)\,\lambda_s + \phi \lambda_f
\end{aligned}
\tag{3}
$$

Here, ρ is the density, c the specific heat capacity, T the temperature, t the time, λ the thermal conductivity, H the radiogenic heat production, and ϕ the porosity. The subscript m, s, and f are acronyms for the porous medium, the solid, and the fluid properties, respectively. For the fluid discharge v, we consider Darcy's law (Nield and Bejan, 2017):

$$
v = -\frac{k}{\mu}\left(\nabla p + \rho_f g \nabla z\right),
\tag{4}
$$

where k denotes the permeability, μ the viscosity, p the pressure, g the gravity acceleration, and z the vertical coordinate component. In this section, we consider natural convection resulting in a temperature dependent density of the following form
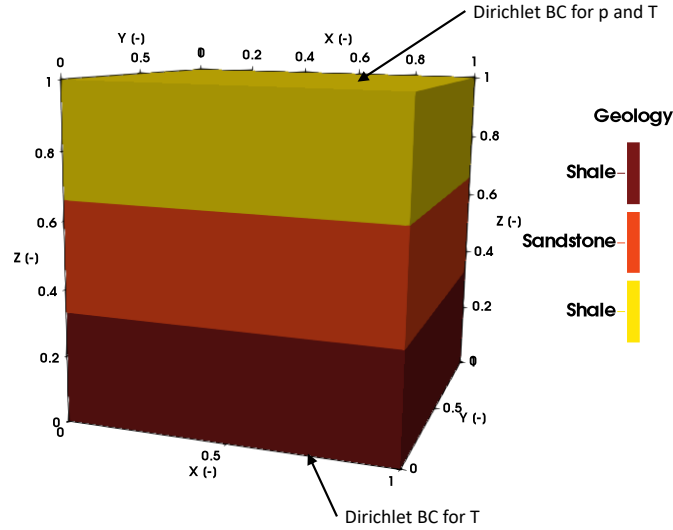
Figure 5. Representation of the geological model used for the geothermal benchmark study. In addition to the geological layers also the boundary conditions are indicated.

(Nield and Bejan, 2017):

$$\rho_f = \rho_{ref}\left(1 - \alpha\left(T - T_{ref}\right)\right), \tag{5}$$

where, the subscript ref denotes the reference properties, and α the thermal expansion coefficient. Furthermore, we consider the Boussinesq assumption (Nield and Bejan, 2017):

$$B = \alpha \Delta T \tag{6}$$

Note that we consider the nondimensional representation to investigate the relative importance of the material properties and to improve the efficiency. This means all properties and variables presented in the following are presented in their dimensionless form. Since the nondimensionalization is not the focus of this paper, we only present the nondimensional equations themselves and not their derivations. For a detailed derivation of the nondimensional equations refer to Degen (2020); Huang (2018). The nondimensional heat and fluid flow can be expressed as:

$$
\begin{aligned}
\nabla \cdot q^* &= 0, \\
q^* &= -\nabla^* p^* + \sqrt{Ra}\, T^* \hat{j}, \\
\frac{\partial T^*}{\partial t} + \sqrt{Ra}\, q^* \cdot \nabla^* T^* - \nabla^2 T^* &= 0, \\
\text{with } Ra &= \frac{\rho_{ref}\, g \alpha k l_{\text{ref}} \Delta T}{\mu \kappa}, \\
\text{and } \kappa &= \frac{\lambda_m}{\rho_{ref}\, c_f}.
\end{aligned}
\tag{7}
$$

Here, the star asterisk denotes the nondimensional variables, $\hat{j}$ the unit vector, and l_{ref} the reference length,.

For the numerical modeling, we are using the finite element software DwarfElephant (Degen et al., 2020a, b), which is based on the MOOSE framework (Permann et al., 2020).

The pressure distribution has a Dirichlet boundary condition of zero at the top and the remaining boundary conditions are no-flow boundaries. For the temperature, we apply Dirichlet boundary conditions at the top and the base of the model, where the top boundary condition has a value of zero and the base one has a value of one. The remaining boundaries have zero Neumann boundary conditions. For the time stepping, we consider an equal spacing between 0 and 0.8 resulting in 15 time steps. Note that the time is dimensionless. The initial condition is zero everywhere in the model domain.

During the study, we assume that the top and base layers are low permeable layers with a variation range of 1 to 4 for Ra and the middle layer is a high permeable layer with a variation range of 36 to 64 for Ra. For these parameter ranges, we construct two training data sets: one with 50 samples and one with 100 samples. Throughout the entire paper, we use a Latin Hypercube sampling strategy for the training data sets and a random sampling strategy for the validation data set. In the case of the geothermal example, we have a validation data set consisting of 20 samples.

For this benchmark example, we investigate how to construct reliable surrogate models for the pressure and temperature distribution resulting from the nonlinear and hyperbolic PDE (Eq. 3). We employ the non-intrusive RB method with a Gaussian Process Regression (GPR) as the machine learning method during the projection step since the here tested variations are linear. We use the GPR method during the projection step since it is known to perform well for linear variations and has fewer hyperparameters that need to be determined, yielding a more efficient construction of the surrogate model.

We start the discussion with the surrogate model for the pressure distribution (top three panels of Fig. 6). The pressure distribution for the full finite element model is shown in Fig. 6a. As mentioned in the description of the non-intrusive RB method, the method consists of two steps: the POD step (where we determine the basis functions), and the projection step (here the weights are determined). Overall, we desire to achieve an accuracy of 10^{-3}, this accuracy is chosen with respect to typical errors of pressure measurements. To reach this accuracy, we require five basis functions, which are partly shown in Fig. 3. The error distribution for one randomly chosen sample from the validation data set is shown in Fig. 6b. Here, we can see that the error distributions follow the pattern of the higher-order modes obtained by the POD method. This is in accordance with our expectations since we truncated the basis after reaching the desired accuracy. Consequently, we lose information about the higher-order modes. In Fig. 6c, we compare the accuracies for the training data sets with 100 and 50 samples, and we reach our desired accuracy for 50 simulations already.

Similar results are obtained for the temperature distribution (Fig. 6d), here we require seven basis functions to obtain an accuracy of 10^{-3}. Note that in this example, we chose the same tolerances for the pressure and temperature model. However, they do not need to be the same. Hence, the tolerances can be adjusted to the varying measurement accuracies of, for instance, pressure and temperature measurements. Also, for the temperature model we observe that the errors (Fig. 6e) follow the distribution of the higher-order POD modes and that 50 simulations in the training set are sufficient to reach the desired tolerance (Fig. 6f).
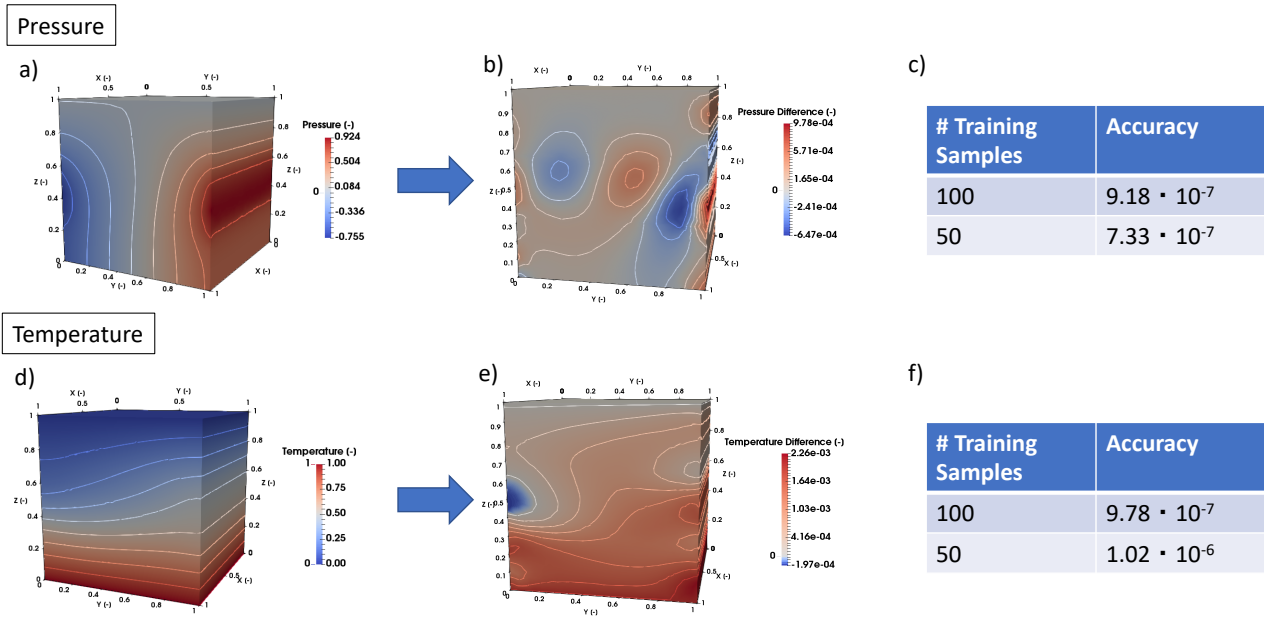
Figure 6. Representation of the accuracy for the pressure and temperature distribution of the geothermal benchmark study. The three top panels show the model for the pressure response, where a) contains the original pressure distribution of the finite element model, b) shows the error between the reduced and full model, and c) lists the accuracy of the reduced pressure response model for different training sample sizes. Analogously, the three bottom panels are displaying the temperature response, where d) shows the temperature response of the finite element model, e) illustrates the error between the reduced and full model, and f) displays the accuracy of the reduced temperature response model for varying training sample sizes.

The benefits of the method become apparent when we have a closer look at the computation times. The finite element model requires 85 s to solve for the pressure and temperature for all timesteps. On the other hand, the reduced model requires 1 ms for solving either the pressure or the temperature for a single time step. Note that in the case of the reduced model, we can solve for individual time steps and state variables and do not need to solve for the entire time period if we are, for instance, only interested in the final time step. But even if we want to solve for both variables for all time steps, we are nearly three orders of magnitude faster with the reduced model. From previous studies using the intrusive RB method, this speed-up is expected to be higher for real case models because the complexity of these models increases slower than the degrees of freedom (Degen et al., 2020a; Degen and Cacace, 2021; Degen et al., 2021b).

3.3 Geodynamic Example

For the numerical modeling of the geodynamic benchmark, we use the thermo-mechanical finite element code SULEC v. 6.1
515 (Buiter and Ellis, 2012; Ellis et al., 2011; Naliboff et al., 2017). The software solves the conservation of momentum:

$$\nabla \cdot \sigma' - \nabla p + \rho g = 0, \tag{8}$$

and the incompressible continuity equation:

$$\nabla \cdot v = 0, \tag{9}$$

where σ' is the deviatoric stress tensor, p is the pressure, ρ is the density, g is the gravitational acceleration, and v is the
520 velocity. The density variations are described analogous to the geothermal example (eq. 5). Also the heat transfer is similar to
the geothermal description (eq. 3), with the main difference that we obtain as an additional source term the shear heating:

$$\rho c \left(\frac{\partial T}{\partial t} + v \cdot \nabla T \right) = \nabla \cdot \lambda \nabla T + H + 2\sigma'_2 \cdot \dot{\epsilon}'_2. \tag{10}$$

Here, σ'_2 and $\dot{\epsilon}'_2$ are the second invariant of the deviatoric stress and strain tensors, respectively.

For this study, we consider the development of a salt dome (Fig. 7). The linear viscous salt rising buoyantly upward through
525 linear viscous sediment layers. The rise is initiated by a sinuisoidal perturbance of the salt-sediment interface with a 40 m am-
plitude. Note that all Neumann boundary conditions (Fig. 7) are no-flow boundary conditions. All velocity Dirichlet boundary
conditions have a value of 0 m s^{-1}. Furthermore, the temperature at the top of the model has a fixed value of 0 °C and at the
base a value of 140 °C. The discharge temperature Dirichlet boundary condition at the lateral sides has a value of 0 m^3 s^{-1}. The
model extends 8 km in the x-direction, and 4 km in the y-direction. For the spatial discretization of the 2D benchmark study,
530 we use quadratic cells with a resolution of 40 m x 40 m, resulting in a structured mesh with 20,301 nodes. SULEC solves for
velocities on the nodes, whereas pressure is constant over the cells and obtained in Uzawa iterations (Pelletier et al., 1989).

For the time stepping, we use fixed time steps of 5 kyrs and simulated from 0 Ma to 6.25 Ma. In contrast to the previous
example, we consider here the dimensional case. Nondimensional forward simulations are more efficient for the physics-based
machine learning methods. However, many simulations in the field of geosciences are often conducted in dimensional form.
535 Hence, we want to illustrate how these simulations need to be processed to ensure an efficient construction of the surrogate
models. Note that we describe the general workflow in Section 3.1, we will here only briefly highlight the most important steps
for the geodynamic case study.

For the given example, we allow a variation of the thermal conductivity of the sediments between 2.5 W m^{-1} K^{-1} and 3.0 W
m^{-1} K^{-1}, and for the salt between 5.0 W m^{-1} K^{-1} and 8.0 W m^{-1} K^{-1}. This means we assume that the thermal conductivity of the
540 salt has an higher uncertainty than the one of the sediments. We construct a training set of 50 samples with a latin hypercube
sampling strategy and the validation data set consists of 10 randomly chosen samples. The remaining model parameters stay
constant throughout all simulations and are displayed in Tab. 2. We consider linear viscous materials that are tracked with
tracers. For the benchmark example, we have in total 182,709 tracers at the start, yielding a minimum of 16 and a maximum
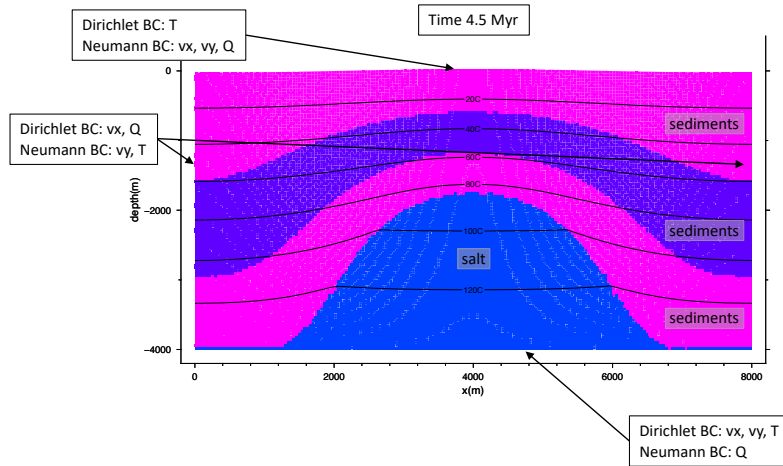
Figure 7. Geological model of the salt diapir at 4.5 Myr. The temperature isolines are plotted for a realization from the training data set.

Table 2. Model parameters of the geodynamic benchmark example.

Parameter	Value	
	Sediments	**Salt**
Density	2300 kg m^3	2160 kg m^3
Viscosity	3.00·10^{19} Pa s	2.67·10^{18} Pa s
Gravitational acceleration	-9.81 m s^{-2} (in y-direction only)	
Specific heat capacity	833 J kg^{-1} K^{-1}	
Radiogenic heat production	0 W m^{-3}	
Thermal expansion coefficient	0 K^{-1}	

of 20 tracers per cell (which are maintained by a tracer population control). The tracers are moved using a second order
545 Runge-Kutta scheme.

In Fig. 8, we present the result of the finite element forward model and the accuracy of the surrogate model. Fig. 8a shows the temperature distribution of the full model ranging from 0 to 140 degrees. We observe that the changes induced by the variations of the thermal conductivities (at the interface between the salt dome and the sediments) are much smaller than the initial temperature distribution. Furthermore, the initial temperature distribution is independent of any variations in the thermal
550 conductivity. Hence it is the same for all forward simulations. Therefore, we train the model only on the temperature variations, which we in addition scale between 0 and 1 by calculating the minimum and maximum temperature values of the entire training data set. The scaled temperature distribution used within the algorithm is shown in Fig. 8b. If visualizations of the absolute temperature distribution are desired they are obtained by adding the initial condition after calculating the reduced forward model.

555 Fig. 8c displays the difference between the full and reduced model, where the highest errors are in the order of 0.1 °C. This means that the errors induced by the approximation are lower than those of typical temperature measurements. We observe the highest errors at the flanks, in the lower part of the model. This matches our expectations since we have the highest changes at the interface between the salt diapir and the surrounding sediments. Furthermore, the temperature distribution at the flanks is more diffusive in contrast to the very sharp changes at the top of the salt dome, which adds to the higher errors of the reduced

560 model at these locations. In total, we required 18 basis functions to reach the given accuracy. Note that the number of basis functions is relatively high because of the rather abrupt changes that we need to capture.
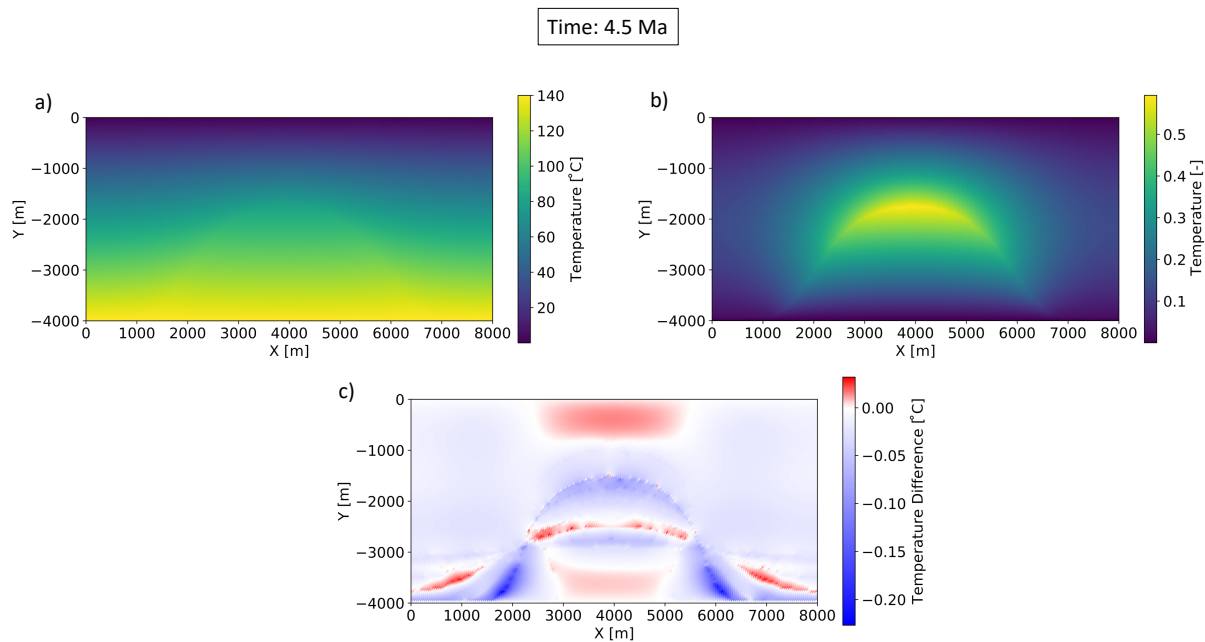
Figure 8. Comparison of the full finite element and the reduced model. Shown are in a) the temperature distribution of the finite element solution, b) the scaled finite element solution as utilized in the training of the reduced model, c) the difference between the reduced and full solution.

Analogously to the geothermal example, we evaluate the gain in computation time. The average computation time over 50 finite element simulations is 1.05 h (including input-output processes) on two Intel Xeon Platinum 8160 CPUs (24 cores, 2.1 GHz, 192 GB of RAM) using the PARDISO solver (Schenk and Gärtner, 2004). In contrast, the average computation time over

565 50 reduced simulations is about 2 ms for a single time step. Note that with the reduced model, we can perform predictions for individual time steps. That this is of general interest can be seen directly in this case study. Although we calculate the solution for all 1,250 time steps, we output 14 only. Note that only these 14 timesteps have been used to construct the surrogate model. However, the model is trained for spatial and temporal variations. Hence, we can retrieve also intermediate time steps. The obtained speed-ups range between five orders of magnitude (interested in all 14 display timesteps) and six orders of magnitude

570 (interested in a single time step).

Table 3. Hyperparamters of the neural network for the geodynamic benchmark example. Note that hl denotes hidden layers

Hyperparameter	Value
Number of hidden layers	3
Number of neurons per hidden layer	427 (hl 1), 383 (hl 2), 313 (hl 3)
Number of epochs	22,438
Learning rate	$4.68 \cdot 10^{-4}$
Batch size	216
Loss Function	sigmoid
Optimizer	Adam

To construct the surrogate model, we require in total 18 basis functions to reach our POD tolerance of 10^{-3}, as displayed in Fig. 9. Having a closer look at the basis functions, we observe that the "low frequency" information is presented in the first basis functions and the "high frequency" information in the last basis functions. This nicely shows the analogy between the non-intrusive RB method and the Fourier decomposition.

575   For the determination of the hyperparameters, we use bayesian optimization with hyperbands (as described in Sec. 3.1). To save memory, the training was performed using only every second node. All results and accuracies presented here, are calculated using all nodes in space. The hyperparameters obtained after optimization are presented in Tab. 3. With these hyperparameters, we obtain an error of $2.21 \cdot 10^{-6}$ on the scaled training data set and an error of $1.99 \cdot 10^{-6}$ on the scaled validation data set.
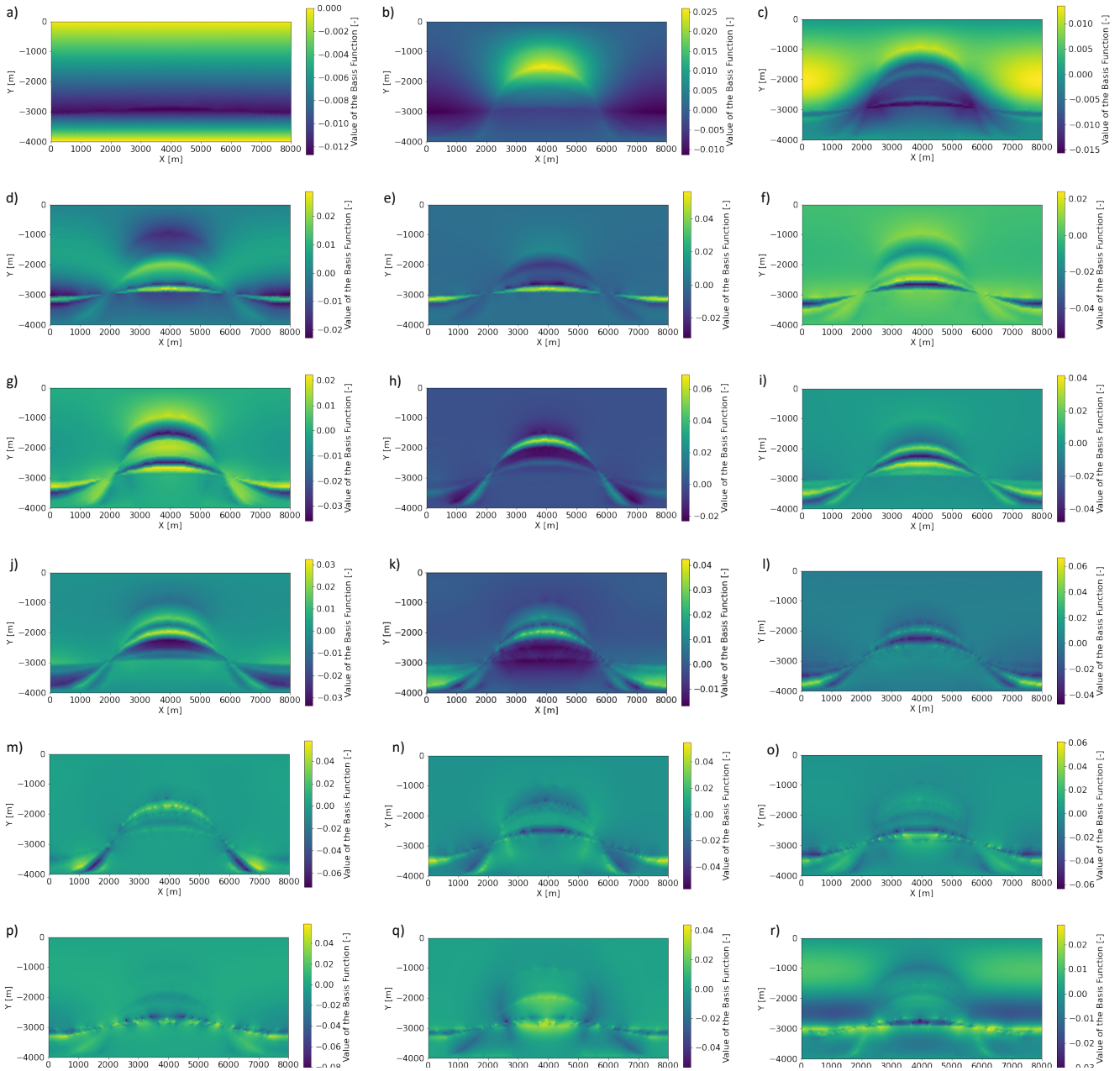
Figure 9. Illustration of the 18 basis functions of the surrogate model, where a) shows the first basis function and r) the last basis function.

3.4 Hydrological Example

In this section, we present a proof-of-concept hydrological application inspired by a well-known two-dimensional infiltration problem on a domain with soil heterogeneity, originally proposed by Forsyth et al. (1995). The setup is illustrated in Figure 10. The domain has impervious lateral and bottom boundaries, and four zones with different soils, represented through different permeabilities. A uniform initial pressure of $h = -100$ cm was set and an infiltration flux q is prescribed on the top-left corner
585   of the domain.

Although this is evidently a benchmark problem, it offers many of the complexities which arise in subsurface flow simulation, such as heterogeneity and sharp infiltration fronts, and therefore nicely serves as a proof-of-concept.
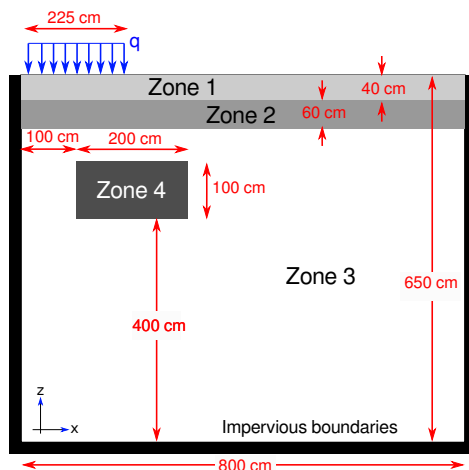
Figure 10. Sketch of the infiltration problem for the proof-of-concept hydrological application. Adapted from Forsyth et al. (1995).

The physical model for variably saturated flow in a porous medium is the Richards equation (Richards, 1931)

$$\frac{\partial \theta}{\partial t} = \nabla \left[\mathbf{K}(h) \nabla (h + z) \right], \tag{11}$$

590   where h is pressure, θ is the volumetric soil water content, K is hydraulic conductivity, t is the time, and z is the vertical coordinate. A soil model is necessary to provide closure relationships, for which, for this case, we use the Mualem model for the water retention curve $\theta(h)$ and the van Genuchten model for $K(\theta(h))$ (van Genuchten, 1980). The resulting mathematical model is a highly nonlinear diffusion equation, formally an elliptic-parabolic PDE, which also shows very sharp front propagation akin to that of advection problems (Caviedes-Voullième et al., 2013).

595   To numerically solve Equation 11 we rely on the well-established hydrological model ParFlow (Kuffour et al., 2020). ParFlow solves the Richards equation via a backward Euler finite-difference scheme and uses multigrid-preconditioned Newton-Krylov methods to solve the resulting system. It is massively parallelized and in fact, recently ported to GPUs (Hokkanen et al., 2021). For the infiltration problem here, we do not leverage on ParFlow's HPC capabilities, since the time-to-solution of this

Table 4. Hyperparameters of the neural network for the hydrological benchmark example. Note that hl denotes hidden layers

Hyperparameter	Value
Number of hidden layers	6
Number of neurons per hidden layer	88
Number of epochs	26,750
Learning rate	$3.08 \cdot 10^{-4}$
Batch size	152
Loss Function	sigmoid
Optimizer	Adam

problem is of only a few seconds on a single CPU core. In contrast, we ran all realizations of both training sets in parallel in an
600   HPC node in the JUWELS system at the Jülich Supercomputing Centre.

In this proof-of-concept exercise, we first construct a training set for the non-intrusive RB method. 100 different combinations of permeabilities for the four materials in the domain are considered, as well as various inflow rates q. A Latin Hypercube strategy was used for sampling the 5-dimensional parameter space. A validation set of 20 samples (with a random sampling strategy) was also generated. Importantly, permeability values follow a log-normal distribution, whereas the infiltration flux
605   follows a uniform distribution. The original permeabilities proposed by Forsyth et al. (1995) were selected as the mean value for the log-normal distributions with a standard deviation of 1. The uniform distribution for the infiltration flux was centered around $4 \, \mathrm{cmd}^{-1}$ and a standard deviation of 0.5.

Similar to the previous case studies, we use the non-intrusive RB method to construct a surrogate model that predicts the hydraulic head for every parameter combination in the above-defined ranges at every time step. For the construction of the
610   surrogate model, we require 124 basis functions. This number is significantly larger than the ones in the previous examples. This is related to the more pronounced nonlinearity of the solutions, which differs for the various time periods. Nonetheless, we obtain solutions in less than 3 ms (calculated over 100 iterations) yielding a speed-up of three orders of magnitude (if we want to predict the solution for a single time step). With the hyperparameters presented in Tab. 4, we achieve an accuracy of $8.71 \cdot 10^{-7}$ for the scaled training set and $2.89 \cdot 10^{-5}$ for the validation data set.

615   The errors for the training and validation data set are global errors. In the next step, we look at the spatial distribution of the approximation errors. Therefore, we pick an arbitrary solution from the validation data set and compare the solution for the last time step (Figure 11). Fig. 11a and b look visually the same, which is the reason we focus the presentation on the difference plot displayed in Fig. 11c. Here, we observe that no part of the model is underrepresented and that the errors are the highest in areas where there is strong spatial heterogeneity of the permeability fields and of the infiltration flux (most notably around
620   Zone 4), both of which contribute to strong non-linear gradients in hydraulic head. The magnitude of the differences is below 2 cm which is for all practical purposes an excellent agreement.

To ensure that the behavior is the same over the entire time period, we pick four random time steps and repeat the procedure (Figure 12). Note that for an easier comparison between the various temporal responses, we plot the solution over the number
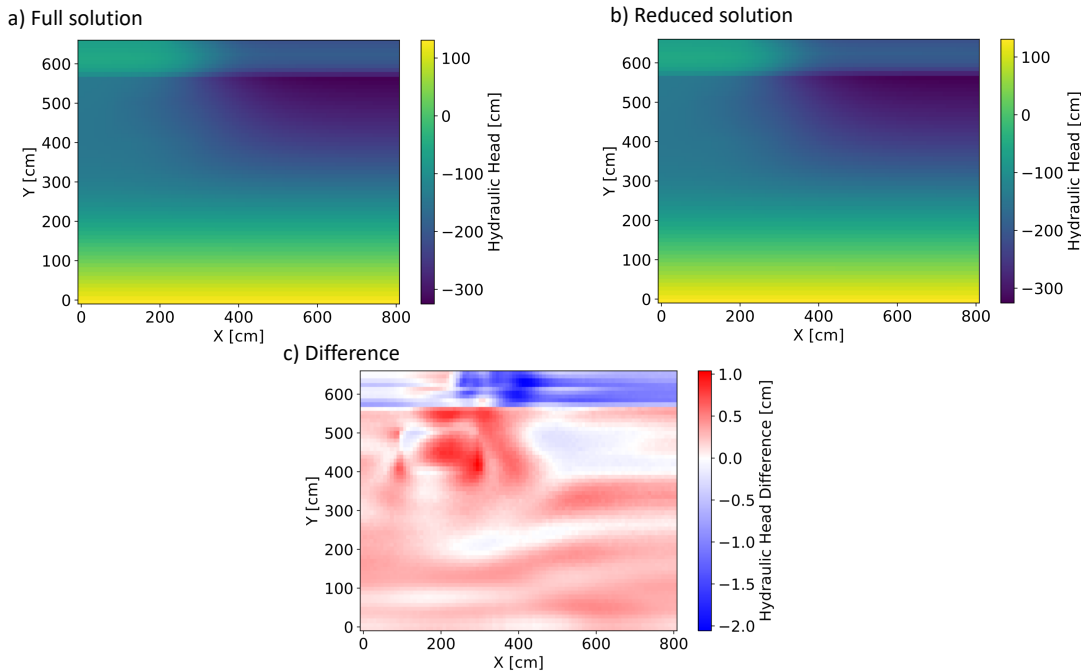
Figure 11. Comparison of the prediction accuracy of the surrogate model for subsurface hydrology. We show in a) the full solution of the Richards equation, in b) the prediction from the surrogate model and in c) the difference between the full and reduced solutions for the time step 30 of an arbitrary chosen solution from the validation data set.

of nodes and no longer in the 2-D representation from above. We observe that the approximation quality is good over the entire
625   time period and that the errors tend to decrease with progressing time steps, which is again related to the nonlinearity.

3.5   Physics-Based Machine Learning versus Data-Driven Machine Learning

In the previous section, we demonstrated how the non-intrusive RB method can be used to construct reliable and efficient surrogate models through three designated benchmark examples. The focus of this manuscript is to illustrate the perspective of physics-based machine learning for subsurface geoscientific applications and how this perspective might differ for various
630   approaches existing for physics-based machine learning. Therefore, we extend the previous sections by constructing surrogate models through neural networks for all three examples. We chose neural networks for this purpose because this enables a comparison between a physics-based and a data-driven machine learning approach, while at the same time showing also the differences between the two paradigms presented for physics-based machine learning, as we detail later on.

One advantage associated with physics-based machine learning vs. data-driven approaches is the reduction in the amount
635   of training data required by reducing the number of admissible solutions through physical knowledge (Faroughi et al., 2022; Raissi et al., 2019). Due to the simplicity of the presented examples this could not be observed, and we obtained similar global
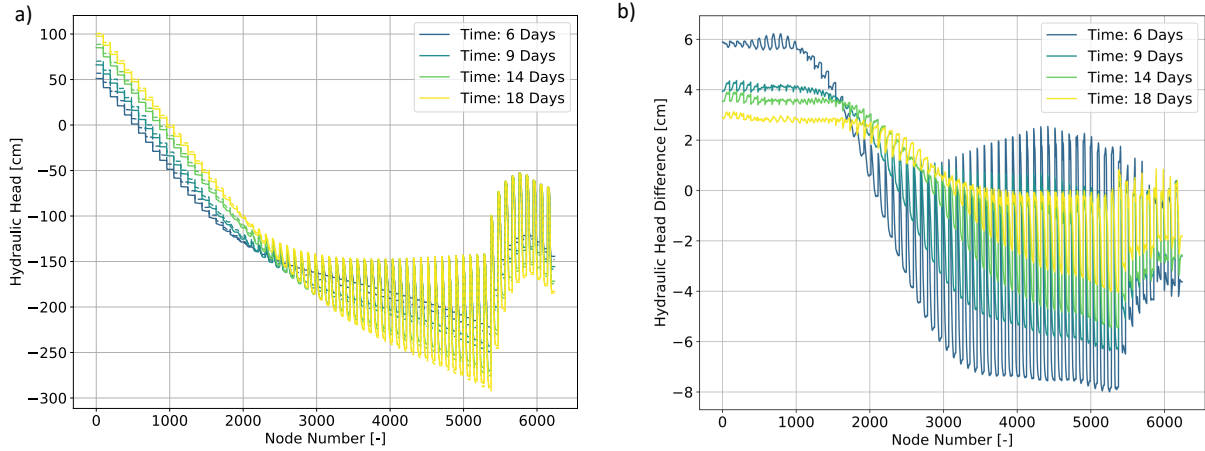
Figure 12. Comparison of the prediction accuracy of the surrogate model for various time steps. We show in a) the response, and in b) the differences between the full and reduced solutions. In a) the full solutions are denoted with solid lines and the reduced solutions with dashed lines.

errors for both the non-intrusive RB and the NN surrogate models. Nonetheless, it has been shown in, for instance, Santoso et al. (2022) that for more complex applications significantly less data is required for physics-based approaches.

Another important aspect becomes apparent by focusing on the local error distributions presented in Fig. 13. Generally, the error distributions show a noisy behavior for the data-driven approach, whereas they exhibit a smooth behavior for the non-intrusive RB method. Furthermore, we can observe in the areas marked with A and B, that boundary conditions are not preserved. Not only are they not preserved but show also one of the highest errors in the entire surrogate model, which is related to the general challenge of optimizing for loss function values close to zero (Chuang and Barba, 2022). Furthermore, we observe in area C a sharp line, where adjacent to each other regions of higher and lower error values occur, which can not be explained through physical processes. Similar observations of error distributions not corresponding to physical effects are observed for areas D and E. In contrast, the error behavior of the non-intrusive RB method is clearly related to physical processes and changes in material properties, as detailed in the previous sections. This observation is not only relevant for the comparison of data-driven and physics-based machine learning approaches but also for distinguishing between the various physics-based machine learning methods. Methods that use physics-guided loss functions (e.g., PINNs) will exhibit the same error behavior as the physics enters only in the loss functions, and also physics-guided architecture will suffer from this as long as the outer layer still operates as a neural network.

Finally, in Tab.5, we compare the construction times for the surrogate models (both for the non-intrusive RB method, and the NN), the evaluation times for a single surrogate model run, and the original computation times for the full-dimensional simulations. The first observation is that the construction time of the surrogate models is considerably smaller for the non-intrusive RB method than for the NN. This is related to the dimension of the training data set entering the machine learning part. For the non-intrusive RB method, we first determine the basis functions and determine only the coefficient over the

Table 5. Comparison of the computational cost for the three benchmark examples considering both physics-based and data driven machine learning methods. Note that the construction time excludes the time required for the generation of the training data. The hyperparameters of the NN are identical to the ones described for the non-intrusive RB method. For the geothermal example, we use the hyperparameters provided for the geodynamic example since the non-intrusive RB method used a Gaussian Process Regression, and we perform the construction only for the temperature.

	Geothermal		Geodynamic		Hydrologica	
	NI-RB	NN	NI-RB	NN	NI-RB	NN
Surrogate Construction Time [min]	0.58	84	7	125	20	115
Surrogate Evaluation Time [ms]	1	3	2	3	3	2
Full Model Solve [min]	1.4		63		0.05	

machine learning technique. This means, the training data has the dimension $N_s \times N_{bfs}$, where N_s is the number of snapshots and N_{bfs} is the number of basis functions. In contrast, the dimension of the training data set for the NN is $N_s \times N_{dofs}$, where N_{dofs} denotes the number of spatial degrees of freedom (e.g., nodes, elements). Typically the number of basis functions is orders of magnitude smaller than the degrees of freedom yielding the reduction in the construction time. The evaluation cost of the surrogate models is comparable for both the non-intrusive RB method and the NN. Furthermore, the evaluation times of the surrogate model are between three to six orders of magnitude lower than the full-order evaluations. This demonstrates the benefits of surrogate models for applications where results are either required in real-time or numerous evaluations are necessary. Note that again the discussed aspect of the increased construction time applies not only to NNs but also to methods such as PINNs.

To conclude, even in applications where we do not have the added benefit of the reduction in the amount of simulation required, the non-intrusive RB method performs better than data-driven alternatives, while yielding a lower computational cost.

Another point to note, in this manuscript we looked at parameterized partial differential equations, assuming the parameter of interest is a material parameter. This assumption is purely exemplary. PINNs sampling the input from the state itself can incorporate changing initial and boundary conditions through the sampling. Similar considerations are valid for the non-intrusive RB method. Here, additional parameters are included to take variations in the initial and boundary conditions into account. Examples are provided in, for instance, Degen (2020); Degen and Cacace (2021); Degen et al. (2022b); Grepl (2005).

4   Challenges and Requirements

We introduced three challenges and four requirements common to many geoscientific applications in the introduction. In the following, we elaborate on how physics-based machine learning, in particular, the non-intrusive RB method can help to address these challenges and fulfill the requirements.
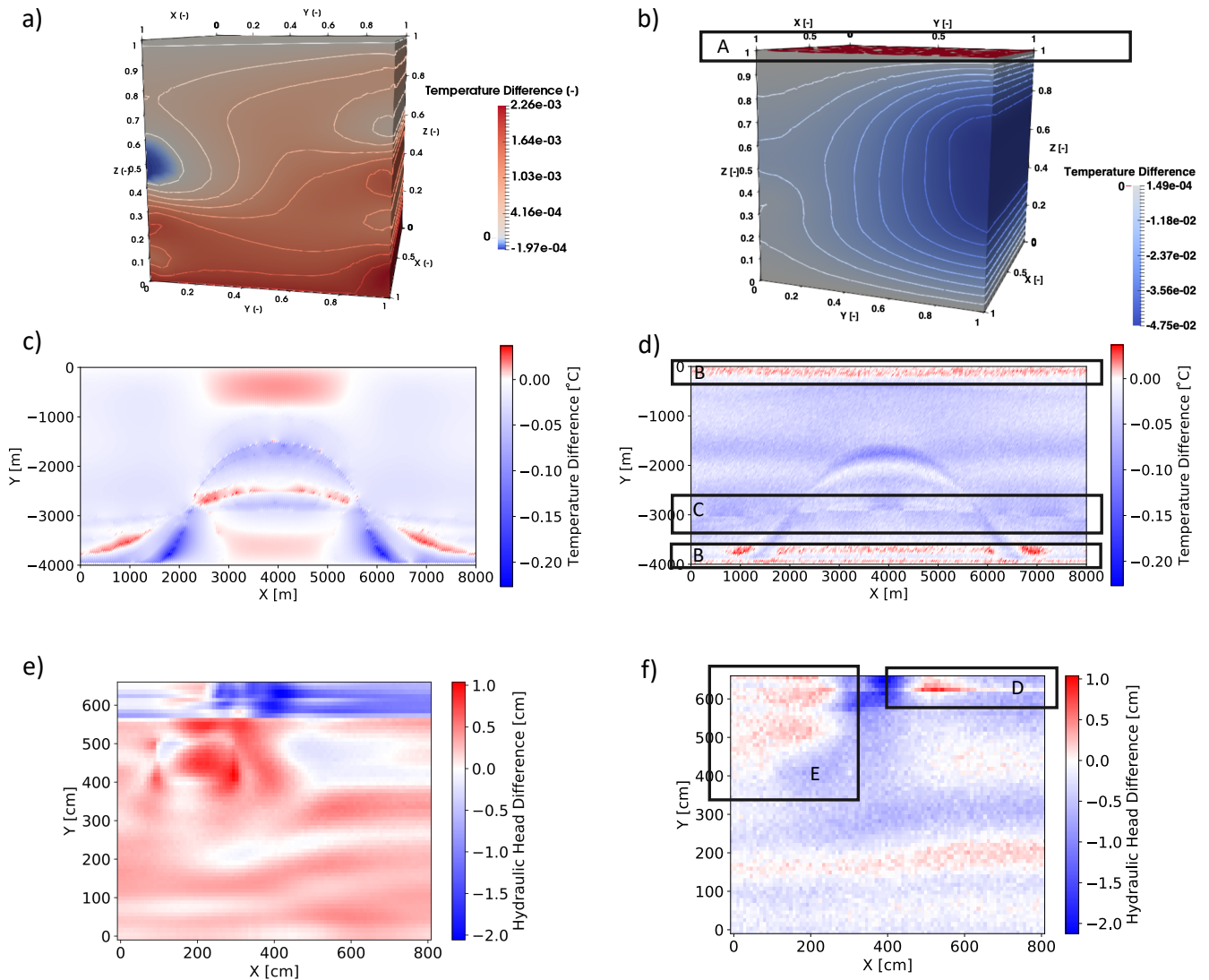
Figure 13. Comparison of the local error distribution for the surrogate models of the geothermal example (a) non-intrusive RB method, b) NN), the geodynamic example (c) non-intrusive RB method, d) NN), and the hydrological example (e) non-intrusive RB method, f) NN).

4.1   Challenge 1: Sensitivity Analysis

Global sensitivity analyses determine the relevant physical parameters for the respective processes and their correlation (Sobol, 680   2001). Therefore, they enhance our understanding of the subsurface but they are computationally demanding, making them pro-

hibitive for large-scale models. The non-intrusive RB method addresses this computational challenge through the construction of surrogate models. As pointed out by Razavi et al. (2012) and Song et al. (2015) the usage of surrogate models in sensitivity analyses is critical. It is critical because if the surrogate model does not represent the high-dimensional model appropriately, we face the high risk of producing biased sensitivity analyses and deriving the wrong conclusions. Although, we agree with this statement, we think it is useful to distinguish between different surrogate modeling techniques highlighting the importance of a surrogate model fulfilling the listed four requirements. As shown for its intrusive counterpart, the results of a sensitivity analysis do not change significantly with the surrogate model accuracy (Degen and Cacace, 2021).

The non-intrusive RB method maintains the input-output relationship of the PDEs, in contrast to, for instance, PINNs. This means that in the classical formulation PINNs sample from spaces in time and space, and produce as output state information at any arbitrary location of the model. These considerations are for a fixed set of model parameters (e.g., material properties) making the method unsuitable for sensitivity analysis, where the model parameters need to be changed. This is not only a disadvantage of PINNs but also of all other methods that follow the idea of going from state information to state information, while maintaining the same material properties. Note that recent developments (Zhang et al., 2022) allow a variation of the material properties. However, the disadvantage is that the material properties embedded in the loss function change throughout the training process, meaning that an unphysical correlation between hyperparameters and physical parameters is introduced.

In contrast, the non-intrusive RB method yields surrogates that do not change the general characteristics of the original problem. Consequently, we can address the challenge of performing computationally expensive global sensitivity analysis by using surrogate models constructed by the non-intrusive RB method. This reduces the computational cost by several orders of magnitude making it a suitable technique to ensure the feasibility of global sensitivity studies also for large-scale models (Degen et al., 2021b; Degen and Cacace, 2021; Degen et al., 2021a, 2022b).

4.2   Challenge 2: Uncertainty

Another application field where the non-intrusive RB method offers great potential is the field of uncertainty quantification. Here, the method can be used to generate fast surrogate models as a replacement for the computationally expensive high-dimensional problem (Degen et al., 2021b, 2022c, a). Uncertainty quantification (Iglesias and Stuart, 2014) determines the uncertainties of the input parameters and has great potential in reducing the risk associated with installations. Uncertainties can be further reduced by finding the optimal position of new measurement locations by employing optimal experimental design methods (Alexanderian, 2021). Also, for this technique, the non-intrusive RB method is suitable to construct surrogate models as shown in Degen et al. (2022a). An important consideration in the usage of surrogate models for uncertainty quantification is the reliability of the surrogate. The surrogate needs to be representative of the original high-dimensional model because otherwise the uncertainty quantification is biased, analog to the discussion in the previous section. Therefore, we require at least error estimators for the surrogate model and in the ideal case error bounds, i.e.: guarantees of the model accuracy. In contrast to other surrogate modeling techniques, the non-intrusive RB method further has the advantage of preserving the input-output relationship and therefore maintains the structure of the original model.

Interesting are also adaptive MCMC approaches, where a smaller reduced basis is constructed which is adapted during the MCMC analysis. The idea of constructing an adaptable reduced basis, is presented in Qian et al. (2017), where the authors create a reduced basis that is valid for a trust-region. An adaptive MCMC algorithm for MT applications has been presented in Manassero et al. (2020), which follows up-on the ideas of Ortega-Gelabert et al. (2020).

To conclude, as for the sensitivity analyses the computational challenge is addressed by using surrogate models. Note that as before methods that map from model parameters to state information are better suitable within an uncertainty quantification framework.

4.3 Challenge 3: Real-Time

In many geoscientific applications, we need to obtain results in real-time. Examples are forecasting applications often employed in the field of hydrology and climate (Bauer et al., 2021; Sabeena and Reddy, 2017), and monitoring and/or risk analyses (Asante-Duah, 2021; Kumar et al., 2021; Yu et al., 2018). The latter analysis is often important for fields such as geothermal energy and natural hazards (Kumar et al., 2021; Liu and Ramirez, 2017; Yu et al., 2018). In the field of geothermal energy that concerns the topic of induced seismicity, and the need to have fast and reliable predictions to employ necessary security measures. However, with increasingly complex models that require more and more computation time, this leads to major challenges. In order to address these challenges, a possibility might be to think of employing physically simplified models, which could result in additional error sources that might bias further analyses. Another possibility is to employ data-driven approaches, however, the question of how the quality of these models can be guaranteed is an open research question (Baker et al., 2019; Willcox et al., 2021). Therefore, physics-based machine learning techniques, such as the non-intrusive RB method that fulfill the listed four requirements have great potential since the produced surrogate models are computationally fast to compute and at the same time maintain the general characteristics of the original problem.

Consequently, the challenge of producing real-time predictions is addressable by the use of surrogate models. We generally see greater potential in methods that follow the conceptual ideas of the non-intrusive RB method since they allow for both parameter and state estimation. Nonetheless, in the context of state estimation and in situations where some physical knowledge is available, methods that map from state-to-state information (such as PINNs) are advantageous since they allow also a direct incorporation of the measurement data. However, these are not the target applications of this manuscript.

Looking at all three presented challenges, we see great potential in the non-intrusive RB method, which combines physics and data-driven approaches. This avoids simplifications on the physical level and at the same time yields explainable, generalizable, scalable, and robust models (Willcox et al., 2021), as we will detail in the next sections.

4.4 Requirement 1.1: Physical Scalability

To improve our understanding of the subsurface, we often want to construct models with an increasingly higher resolution in space and time, and also concerning the parameter representation. To give a few examples, in the field of geodynamics we typically aim to investigate different scales in both time and space. On the one hand, we are interested in large-scale spatial

deformations related to plate tectonics, on the other hand, we want to investigate how microstructures form. For the time component, we face the same issues of being interested in seismic cycles (in the order of milliseconds) and basin dynamics (order of millions of years) (Bauer et al., 2021; van Zelst et al., 2022). Having a single model that crosses over all scales is computationally infeasible, even with adaptive time stepping and spatial discretization schemes. Furthermore, we need to additionally consider the complexity of the parameter space. On a larger scale, local heterogeneities are often negligible and we can use homogenization approaches (Regenauer-Lieb et al., 2013) and often simpler physical formulations. On a smaller scale, these concepts are not applicable and we need to consider the full complexity.

One way to address this problem is a multilevel approach, a concept that works independently of physics-based machine learning. However, physics-based machine learning and other surrogate modeling techniques can considerably accelerate the procedure, as we show later on. The advantages of multilevel approaches to improve convergence have been pointed out by, for instance, Bauer et al. (2021). We can, for example, combine models with different spatial and temporal resolutions, which yields faster convergences. This faster convergence is achieved by starting the solve on the coarsest level first. Once we obtain results for this level, we move to the next level which has a higher resolution. However, since we already have a good starting guess, we need significantly fewer iterations on this level in comparison to the number of iterations required if we would directly solve on the finer level. Multilevel or multi-fidelity approaches imply not only different spatial and temporal resolutions but also allow for the employment of different physical descriptions (adjusted to the resolution) and different methodologies (Peherstorfer et al., 2018).

Physics-based machine learning can improve the computational time on several levels. First of all, we can provide surrogate models for the coarse levels, through methods such as the non-intrusive RB method. The non-intrusive RB method performs very well in the reduction of the spatial and temporal degrees of freedom but is inefficient if the dimension of the parameter space becomes too large. However, other methods might be less efficient in the reduction of space and time but more efficient in dealing with high-dimensional parameter spaces (Benner et al., 2015). Multi-level approaches allow combining these techniques. Hence, we can employ different methodologies on each of these levels tailored to the different requirements. On the finest level, we can even employ the original full-dimensional problem.

4.5   Requirement 1.2: Performance Scalability

In the following, we elaborate on how physics-based machine learning approaches impact the field of HPC. Therefore, we first present the current issues we phase and explain afterwards how the methods presented here help to address these aspects.

Previously, we described that the complexity of the considered physical processes and their associated coupling processes are constantly increasing. This has a high impact on the computational resources since more complex scenarios yield a higher computational demand; hence, often requiring HPC infrastructures. At the same time, we observe the end of the scalability laws such as the Dennard scaling (Frank et al., 2001) and Moore's law (Bondyopadhyay, 1998).

A paradigm shift is underway as a consequence of this. The hardware infrastructure has been moving more and more to graphics processing units (GPUs) and is becoming more heterogeneous than before. This poses major challenges for most areas of Geosciences (Bauer et al., 2021), as we can no longer rely on hardware developments to cover the required higher

computational demand (end of the free-lunch era) but we need to invest in adapting the algorithms and codes themselves (Bauer et al., 2021). This is a non-trivial and potentially time-consuming process that often requires rather specialized knowledge. Moreover, the specific problems can vary greatly for the different geoscientific communities. For example, the climate community relies on very large, well-established community models. Although efforts are underway to port them to GPUs (Bauer et al., 2021), the size and complexity of the codes, and potentially even their inherent structure make this an ambitious and costly task (Lawrence et al., 2018). Often, as well, more than one intensive kernel may exist and the low-hanging computational hotspot to port is not obvious (Gan et al., 2020). A criticism raised by Bauer et al. (2021) is that the software developments are primarily driven by scientific relevance neglecting computer sciences aspects in terms of efficiency, which further complicates the transition to future HPC infrastructures. The situation is different for other geoscientific communities since they have smaller –and possibly more manageable– codes which can therefore be more flexible adapted, or more recent codes, which may even already leverage novel technologies. In either case, porting tasks are still not trivial, but with the help of current developments to leverage GPUs, such as CUDA and OpenACC, as well as domain-specific languages (DSLs) –e.g., Kokkos (Edwards et al., 2014; Trott et al., 2021), RAJA (Beckingsale et al., 2019), Alpaka (Zenker et al., 2016), GridTools (Afanasyev et al., 2021), ClawDSL (Clement et al., 2018)–, the resources and expertise required can be reduced, or at least compartmentalized and better distributed in time. We can also provide cross-domain platforms such as MOOSE and Firedrake to synchronize the developments in computer and domain sciences. Moreover, it is becoming increasingly relevant to consider co-design approaches (Germann, 2021), in which domain-science applications (and developers) engage with library, back-end, and even hardware developers in pointing where significant (progressive and disruptive) advances can be made without constant adaptation from scientific applications.

Other aspects that we would like to address here are the discretization of time (Bauer et al., 2021) and how to adapt beyond the current HPC infrastructures. This includes future-proofing against architectural shifts, vendors, and languages. To name a few examples, we require strategies to adapt to GPUs from various manufacturers (e.g., upcoming AMD and Intel GPU accelerated systems). Solutions such as CUDA are optimized for Nvidia hardware and comparing the performance of CUDA against OpenACC (or CUDA through portability layers such as Kokkos) yields indications that this transition still poses challenges (Artigues et al., 2019; Baig et al., 2020). These considerations do not even cover the problems that might arise from tensor processing units (TPUs). In all, the heterogeneous computing infrastructures of the future and asynchronous processes pose major and partly unpredictable challenges emphasizing the need for flexible, open-source software projects.

There are various possibilities to address these issues. The first is naturally the intensive training of researchers in the field of HPC, including, for instance, advanced programming training sessions, which is a long learning process. However, here we will focus on the aspect of how physics-based machine learning and other methodologies can help to fulfil these requirements.

Therefore, let us briefly recap the physical problem. In fields such as climate, groundwater, geodynamical, and geothermal applications we often have problems that are governed by partial differential equations (PDEs). The porting of PDE solvers is an ongoing research question (Alexander et al., 2020; Bertagna et al., 2019; Grete et al., 2021; Hokkanen et al., 2021) and no general solution is available yet. We continue to face an immediate need to adjust to new infrastructures consisting of both central processing units (CPUs) and GPUs, and many codes remain still not ported. An alternative approach to harness HPC

resources is the usage of physics-based machine learning methods to leverage heterogeneous computations. The need to use GPUs has also been recognized in climate science (Bauer et al., 2021), where the use of machine learning techniques wherever it may be possible without a loss of quality is seen as a way forward. Similar arguments are present in other related communities where porting and machine learning are seen as ways to address the computational demands in fluid mechanics (Dauxois et al., 2021), and in hydrodynamics where they are perceived as possible surrogates for forecasting (Morales-Hernández et al., 2020).

In the following, we explain how physics-based machine learning methods help to optimally use HPC infrastructures, starting with PINNs. PINNs are directly portable to GPUs since they consist of a deep neural network. The calculation of the PDE inside the loss function offers also portability since the derivatives are calculated via automatic differentiation (Raissi et al., 2019). Additionally, the non-intrusive RB method is partly portable to GPUs insofar as the (deep) neural network that replaces the classical Galerkin projection can be directly ported. The calculation of the PDEs and hence the generation of the training data is not easily portable. However, note that this task is embarrassingly parallel since all evaluations are independent of each other. Hence, we can take full advantage of a given HPC system. This is not the case for the inverse processes for which we use the surrogate models. Methods such as Markov chain Monte Carlo are only partly parallelizable since they are dependent on previous results, emphasizing further the usefulness of the methodologies presented here. The presented schemes propose an intermediate solution to solve the paradigm of the new architectural constraints. It can also easily incorporate any new advances for porting PDEs to GPUs. Of course, limitations can still arise when the size and cost of forward simulations for the training set are too large, which reverts the issue to porting PDE solvers. However, once that is achieved, physics-based machine learning workflows directly benefit.

The advantage of such workflows is that GPU support for most of the algorithms required currently exists in environments such as TensorFlow and PyTorch (as described in Section 4.7) and DSLs. With the here presented software packages and environments ranging from machine learning libraries over cross-domain finite element solvers to DSLs, we aim to provide a future proof concept that can also adapt to new GPU systems (including vendors other than Nvidia) and new architectures such as TPUs. In the ideal case, software built upon these frameworks requires only minimal changes, if at all, to adapt to the architectural changes.

The high portability of these workflows is in line with the general proposition in Bauer et al. (2021), where the authors emphasize the need to use utmost flexible software for adaptability with short-living hardware infrastructures. We strongly highlight that the lifespan of scientific software is usually substantially longer than that of hardware. Consequently, software sustainability must be at the center of geoscientific model development. In the context of a changing hardware landscape, it is therefore essential to future-proof codes, so that their applicability to new hardware (and software) paradigms is ensured, contributing to their sustainability. Ideally, it will not be necessary to repeat substantial porting efforts in the future. This future-proofing is non-trivial and intertwined with ongoing porting efforts, and to achieve these porting efforts should strive to achieve performance portability, productivity, and maintainability. For this, a good separation-of-concerns (between geoscientific code, algorithms, and hardware-specific code) is necessary, for which again solutions such as DSLs are interesting, and evermore popular approaches (Artigues et al., 2019; Evans et al., 2021; Lawrence et al., 2018). Although there are several paths to this,

we argue that leveraging on well-established accelerated environments to build physics-based machine learning models acting as PDE-solver surrogates, is one approach.

From the numerics and algorithmic point of view, there are also opportunities to improve scalability and performance, some of which may also be better suited to upcoming hardware, but for which new implementations in specific codes are inescapable. These include ideas such as mesh adaptivity (e.g. Chen et al., 2021; Kevlahan, 2021; Gerhard et al., 2015; Özgen-Xian et al., 2020; Piggott et al., 2005), higher-order local schemes (e.g. Abdi et al., 2017; Käser and Dumbser, 2006; Gassner and Winters, 2021; Kärnä et al., 2018), local time-stepping (e.g. Dazzi et al., 2018) and parallelization in time (Lions et al., 2001; Maday, 2008). These different strategies can also be combined into even more sophisticated algorithms exploiting several features (e.g. Caviedes-Voullième et al., 2020; Dawson et al., 2013; Rannabauer et al., 2018). Mesh adaptivity seeks to reduce computational load, compress data and reduce communications while keeping accuracy, all of which contribute to scalability. Higher-order schemes which preserve locality (e.g., Discontinuous Galerkin) increase the load on arithmetic where accelerators excel and allow to reduce the number of elements (which potentially reduces memory and communication costs) for a target accuracy in comparison to lower-order schemes. Local time-stepping explicit schemes allow for asynchronous updates, which enhances parallelization and may allow reducing costly reduction operations.

Similarly, parallelization in time can strongly relax the fundamental constrain of one time step depending on the previous one (Bauer et al., 2021; Carraro et al., 2015; Fisher and Gürol, 2017; Hamon et al., 2020). Interestingly, some physics-based machine learning techniques provide parallel-in-time schemes (Meng et al., 2020). Here, we highlight the parareal algorithm, first developed by Lions et al. (2001). The current stage of the algorithm is presented in Maday (2008). The algorithm follows domain decomposition approaches that divide the domain into smaller subdomains. Domain decomposition methods then solve the problem iteratively over the subdomains using different processors. The parareal algorithm divides the global time evolution into a set of independent smaller time intervals. For the iterative algorithm, the method relies on a predictor-corrector approach that yields generally fast convergences (Maday, 2008). A parareal implementation for PINNs, called PPINNs has been already developed in Meng et al. (2020). Similarly, the Parallel Full Approximation Scheme in Space and Time (PFASST) (Emmett and Minion, 2012, 2014; Hamon et al., 2020) falls into the same class as parareal algorithm.

4.6   Requirement 2: Interpretability

The next requirement that needs to be addressed is interpretability. For PDE-based applications, we often face the need to provide predictions. However, guaranteeing the correctness of these predictions, with surrogate models is challenging (Willcox et al., 2021), especially for nonlinear applications. Many surrogate model techniques that work well for linear examples do not work in the nonlinear setting (Grepl, 2012). Some physics-based methods, such as the reduced basis method, which yield interpretable models, are extendable to the nonlinear case but are significantly impacted in their efficiency when encountering high degrees of nonlinearity (Hesthaven and Ubbiali, 2018). Neural networks are known for approximating well nonlinear problems (Raissi et al., 2019), but require a high amount of data, do not preserve physical principles, and are in general not explainable.

Both of these aspects are critical since applications in the field of, for instance, geothermal energy and geodynamics, do not have a sufficient amount of data. Furthermore, we need to conduct predictions and risk assessments, which are critical with data-driven schemes since their quality is related to the quality and amount of training data. These aspects are the reason, why the mentioned application fields heavily rely on physical models. We see here greater potential for methods such as the non-intrusive RB method, that incorporate the physics into the surrogate model construction instead of PINNs since the non-intrusive RB method results in explainable surrogate models.

To clarify PINNs are a complete black-box approach. That means we provide our input data (e.g., sample points in space and time) and obtain predictions for the entire state. So, the method can reconstruct the state for non sampled points in space and time. In contrast to the non-intrusive RB method, we provide our input data (e.g., rock and fluid properties) and obtain the corresponding state output. This state output is consisting of two parts, the basis functions (determined by the sampling method, e.g. the POD method) and the weights of the basis functions (determined by the machine learning method). This has several advantages:

1. The POD modes correspond to the different physical processes (as demonstrated in the case studies in this paper)

2. The decomposition of our original problem into basis functions times weights is mathematically identical (Hesthaven and Ubbiali, 2018)

3. The machine learning method only investigates the weights

Especially, the last point is important for nonlinear examples. Although a PDE might be nonlinear that does not necessarily mean that the variations of the state are highly nonlinear because of different model parameters. The RB method focuses on the variations of the state caused by changes in the model parameters. So, it depends on this complexity and not on the spatial and temporal distributions.

4.7 Requirement 3: Generalizability

In the following, we want to emphasize the importance of generalizability. Approaches to construct suitable surrogate models need to be applicable across different disciplines and not restricted to single applications only. In this paper, we focus on PDE-based systems and demonstrated the suitability of the non-intrusive RB method for three geoscientific applications with partly significantly varying conditions and requirements. The non-intrusive RB methods works efficiently for all three examples, which cover a wide range of the general challenges encountered in geoscience applications and many other PDE-dominated domains.

Connected to the aspect of generalizability is also the question of accessibility, meaning the open-access/open-source availability of research and software. Accessibility is a topic that has great importance in general, especially in the context of interdisciplinarity. The importance of accessibility is already recognized in Bergen et al. (2019), where Bergen et al. (2019) point out that the recent rapid growth of geoscientific applications using machine learning, is associated with the availability of easy to use machine learning toolboxes.

Machine learning methods are mainly developed by the computer science community, whereas model order reduction methods (the second set of methods presented here) are originating from the scientific computing and applied mathematical community (Swischuk et al., 2019). Geosciences belong to neither of these communities. However, it heavily relies on the methodological developments within these communities to further advance. This opens the path to several subsequent challenges since the terminology and the purposes of the various communities are vastly different.

Consequently, if new algorithms are solely presented in form of papers expert knowledge in several fields is required. This is a major challenge since it makes the exploration of different methodologies extremely time-consuming, especially for changing applications. Therefore, it is of great importance to have flexible access to a broad range of methods. Note that with the term flexible, we mean the implementation of algorithms that are not tight to a specific forward solver. Examples for such accessible software framework are available for many scientific fields. For inverse methods, packages as Dakota (Adams et al., 2020), Raven (Rabiti et al., 2020), and various designated Python libraries such as SciPy (Virtanen et al., 2020), SALib (Herman and Usher, 2017), and pyMC (Patil et al., 2010) are available. Naturally, this does not mean that no prior knowledge about the methods is required. A general understanding of the methods is necessary to ensure a reliable operation and performance of the respective software packages. The step of accessibility should solely simplify the applicability and testing of the different methods for scientific purposes.

In the following, we want to clarify how we understand the term accessibility and which impacts it has on the development of research software. Accessible software means in the first term open-source software. However, accessibility means more than open-source. Following the classification of black-box models in Peherstorfer and Willcox (2016b), we also define software packages as non-accessible if the complexity of the code prevents an understanding of the methodology and if there is no detailed documentation on the package. Specifying this last point, we want to make the difference between educational and scientific software packages. Educational software packages are often written in scripting/parsed languages, such as Python, and use simple structures for the implementation (e.g., Ballarin et al., 2017). This eases the learning of the procedures but comes at the cost of lower efficiency. In contrast, scientific software packages are often written in compiled language packages, such as C++, and/or using highly object-orientated structures (e.g., Adams et al., 2020; Rabiti et al., 2020). This complicates the learning process but allows the efficient execution of the methods for real-case applications. In no way do we want to state that only educational software packages are accessible and should be further developed. The point that we want to make here is that scientific software packages in addition require extensive documentation and tutorials to compensate for the complex nature of the code, as presented in, for example, Adams et al. (2020); Herman and Usher (2017); Patil et al. (2010); Rabiti et al. (2020); Virtanen et al. (2020). However, scientific software packages that lack these additional resources fall in our opinion under the category of non-accessible software packages.

With this last aspect, we already covered the main impacts of accessibility on software development. For the point of understanding, scripting/parsed languages are preferential. However, from the efficiency point of view, compiled languages would be preferential. Having access to the methods in various compute languages is naturally the ideal case. Here, the parsed implementation serves as an entry point to improve the understanding, whereas the compiled implementation is used later on for the actual studies. However, developing the code in several languages or structures is time-consuming and often neither

possible nor feasible. Therefore, the above-mentioned compensation by documentation seems in our point of view a suitable

950   compromise. Also, note that accessibility is not a responsibility of the developing communities (such as computer sciences) alone but a joint responsibility of all communities.

For machine learning methods various high performance Python packages exist, such as: Keras (Chollet et al., 2015), MXNet (Chen et al., 2015), PyTorch (Paszke et al., 2019), TensorFlow (Abadi et al., 2015), Theano (Team et al., 2016), DeepXDE (implementation of PINNs and various modifications, (Lu et al., 2019)), HeAT (Götz et al., 2020), and SciANN (implementation

955   of PINNs based on Keras/TensorFlow, (Haghighat and Juanes, 2021)).

The variety of available packages shows the potential for easy usability of machine learning in Geosciences but presents also one major danger. The software packages are rapidly developing and for some, the developments stopped (Team et al., 2016). Hence, software that relies on these packages should be developed with the possibility to interchange these libraries. Clearly, this is a non-trivial task but nevertheless important to consider.

960   Regarding open-source finite element solver several packages are available (in various compute languages): e.g., FEniCS (Alnæs et al., 2015), FireDrake (Rathgeber et al., 2016), and MOOSE framework (Permann et al., 2020). Note that we need these solvers to generate the input data for the physics-based machine learning methods.

4.8   Requirement 4: Robustness

The last requirement, we address here is the aspect of robustness. Hence, the question of how to ensure that the model does

965   not change if, for instance, the accuracy or the amount of data used for the surrogate model construction is increased. For the application of data-driven methods to PDE-based systems that can be rather challenging. If we, for instance, generate training sets from numerical simulations and use only spatial and temporal sample points of the solutions to construct our surrogates (as done for PINNs), we need to be careful that we include all physical effects. To give an example, we need to ensure that for a geothermal study we do not extract samples only in the conductive but also advective regime. This is an aspect that does

970   not occur for the non-intrusive RB method since it constructs the surrogate model directly on the solution space. For the non-intrusive RB method (geothermal example in Section 3.2) we construct two surrogate models with different training data set sizes. Although we see a difference in the overall accuracy of the model, both models return the same general characteristics of the physical problem, demonstrating the robustness. This can be further highlighted by looking at the intrusive counterpart of the method. Degen and Cacace (2021) show, that the results of the sensitivity analysis are not impacted by the model accuracy,

975   which demonstrates that the general characteristic is the same for all models.

5   Conclusions

We present the great potential of physics-based machine learning for geoscientific systems governed by partial differential equations through three designated examples from the fields of geothermal energy, geodynamical systems, and hydrology. These three applications fields were chosen to demonstrate the potential across different geoscientific applications. We here

980   address three challenges: enabling efficient i) global sensitivity analyses, ii) uncertainty quantification, and iii) real-time appli-

cations for geoscientific problems dominated by PDEs. Furthermore, it is essential that the techniques used for enabling these efficient analyses fulfill the requirements of being scalable, interpretable, generalizable, and robust.

Bergen et al. (2019) pointed out the importance to provide an optimal predictive understanding of the processes for different geosystems. Predictability is a great challenge in ML techniques if based purely on data. In contrast, methods that incorporate our physical knowledge have great advantages in the geosciences. We see especially potential for the non-intrusive RB method since it maintains the structure of the original physical problem, a viewpoint that is also supported by Reichstein et al. (2019) and Willcox et al. (2021).

The presented methods allow for real-time or faster than real-time updates of models. This makes them interesting for applications and concepts such as digital twins. Digital twins are digital copies of the orginal process, having real-time capabilities to assist in, for instance, decision- and policy-making (Bauer et al., 2021). However, we are aware that the geoscience community is rather skeptical concerning black-box approaches such as ML (Bauer et al., 2021). For this reason, it is important to ensure transparent qualitative and quantitative measures (Baker et al., 2019), which are a big challenge for ML in general. In this regard, we like to point out again the non-intrusive RB method, which is in contrast to the other methodologies based on a mathematical rigorous proven method. The intrusive version has a limited applicability and efficiency for nonlinear (Hesthaven and Ubbiali, 2018) and hyperbolic PDEs (e.g., Benner et al., 2015; Hesthaven et al., 2016). In the original formulation of the RB method, a Galerkin projection is used instead of the here presented ML techniques. In consequence, this means that by performing simplifications to the physical problem, we are always able to fall back on a proven scheme for quality insurance purposes for elliptic and parabolic PDEs.

Current research on the mathematical foundations of Physics-based machine learning also includes efficient error estimators for hyperbolic PDEs, as well as the extension to high-dimensional parameter spaces. Further developments in these directions will allow for an ever wider applicability of the methods in the field of geosciences.

Code and data availability. The training and validation data sets, their associated model parameters, as well as the Non-Intrusive RB and Neural Network code for the construction of all surrogate models is published in the following Zenodo repository (https://doi.org/10.5281/zenodo.8369108). For the construction of the geothermal data set the software package DwarfElephant (Degen et al., 2020a, b) has been used. This software is based on the finite element solver MOOSE (Permann et al., 2020), and freely available on Zenodo (https://zenodo.org/badge/latestdoi/117989215). For the generation of the hydrology data set, we used ParFlow v3.10 (available on Zenodo, https://doi.org/10.5281/zenodo.6413322), and for the geodynamic data set SULEC v6.1. The presented geodynamic benchmark example is conceptually simple and reproducible in other open-source software packages such as Aspect v.2.4 (available on Zenodo, https://doi.org/10.5281/zenodo.6903424) with the data provided in the manuscript.

Author contributions. All authors discussed and interpreted the presented work. Denise Degen carried out the simulations and drafted the manuscript. The Hydrological Example and High Performance Computing Section have been jointly drafted by Daniel Caviedes Voullième

and Denise Degen. Ana González-Nicolás configured, ran and postprocessed the hydrological simulations. The salt dome example was set up by Susanne Buiter. Furthermore, all authors read and approved the final manuscript.

Competing interests. The authors declare that they have no conflict of interest.

1015 *Acknowledgements.* The authors gratefully acknowledge the funding provided by Exploratory Research Space Prep Funding through the project "High Performance Computing in the Geosciences" (project number: PF-JARA-SDS009). Furthermore, we thankfully acknowledge the funding provided by the BMBF BioökonomieREVIER funding scheme with its "BioRevierPlus" project (funding ID: 031B1137EX).

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Good-
fellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S.,
Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F.,
Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X.: TensorFlow: Large-Scale Machine Learning on Heterogeneous
Systems, https://www.tensorflow.org/, software available from tensorflow.org, 2015.

Abdi, D. S., Wilcox, L. C., Warburton, T. C., and Giraldo, F. X.: A GPU-accelerated continuous and discontinuous Galerkin
non-hydrostatic atmospheric model, The International Journal of High Performance Computing Applications, 33, 81–109,
https://doi.org/10.1177/1094342017694427, 2017.

Adams, B., Bohnhoff, W., Dalbey, K., Ebeida, M., Eddy, J., Eldred, M., Hooper, R., Hough, P., Hu, K., Jakeman, J., Khalil, M., Maupin,
K., Monschke, J., Ridgway, E., Rushdi, A., Seidl, D., Stephens, J., Swiler, L., and Winokur, J.: DAKOTA, A Multilevel Parallel Object-
Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.12
User's Manual, Sandia National Laboratories, Tech. Rep. SAND2020-12495, 2020.

Afanasyev, A., Bianco, M., Mosimann, L., Osuna, C., Thaler, F., Vogt, H., Fuhrer, O., VandeVondele, J., and Schulthess, T. C.: GridTools: A
framework for portable weather and climate applications, Software X, 15, 100 707, https://doi.org/10.1016/j.softx.2021.100707, 2021.

Alexander, F., Almgren, A., Bell, J., Bhattacharjee, A., Chen, J., Colella, P., Daniel, D., DeSlippe, J., Diachin, L., Draeger, E., Dubey, A.,
Dunning, T., Evans, T., Foster, I., Francois, M., Germann, T., Gordon, M., Habib, S., Halappanavar, M., Hamilton, S., Hart, W., Huang,
Z. H., Hungerford, A., Kasen, D., Kent, P. R. C., Kolev, T., Kothe, D. B., Kronfeld, A., Luo, Y., Mackenzie, P., McCallen, D., Messer,
B., Mniszewski, S., Oehmen, C., Perazzo, A., Perez, D., Richards, D., Rider, W. J., Rieben, R., Roche, K., Siegel, A., Sprague, M.,
Steefel, C., Stevens, R., Syamlal, M., Taylor, M., Turner, J., Vay, J.-L., Voter, A. F., Windus, T. L., and Yelick, K.: Exascale applications:
skin in the game, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 378, 20190 056,
https://doi.org/10.1098/rsta.2019.0056, 2020.

Alexanderian, A.: Optimal experimental design for infinite-dimensional Bayesian inverse problems governed by PDEs: a review, Inverse
Problems, 37, 043 001, 2021.

Alnæs, M., Blechta, J., Hake, J., Johansson, A., Kehlet, B., Logg, A., Richardson, C., Ring, J., Rognes, M. E., and Wells, G. N.: The FEniCS
project version 1.5, Archive of Numerical Software, 3, 2015.

Antoulas, A. C.: Approximation of large-scale dynamical systems, vol. 6, Siam, 2005.

Ardabili, S., Mosavi, A., Dehghani, M., and Várkonyi-Kóczy, A. R.: Deep Learning and Machine Learning in Hydrological Processes
Climate Change and Earth Systems a Systematic Review, in: Lecture Notes in Networks and Systems, pp. 52–62, Springer International
Publishing, https://doi.org/10.1007/978-3-030-36841-8_5, 2020.

Artigues, V., Kormann, K., Rampp, M., and Reuter, K.: Evaluation of performance portability frameworks for the implementation of a
particle-in-cell code, Concurrency and Computation Practice and Experience, 32, https://doi.org/10.1002/cpe.5640, 2019.

Asante-Duah, D. K.: Hazardous waste risk assessment, CRC Press, 2021.

Audouze, C., De Vuyst, F., and Nair, P.: Reduced-order modeling of parameterized PDEs using time–space-parameter principal component
analysis, International journal for numerical methods in engineering, 80, 1025–1057, 2009.

Audouze, C., De Vuyst, F., and Nair, P. B.: Nonintrusive reduced-order modeling of parametrized time-dependent partial differential equa-
tions, Numerical Methods for Partial Differential Equations, 29, 1587–1628, 2013.

1055   Baig, F., Gao, C., Teng, D., Kong, J., and Wang, F.: Accelerating Spatial Cross-Matching on CPU-GPU Hybrid Platform With CUDA and OpenACC, Frontiers in Big Data, 3, https://doi.org/10.3389/fdata.2020.00014, 2020.

Baker, N., Alexander, F., Bremer, T., Hagberg, A., Kevrekidis, Y., Najm, H., Parashar, M., Patra, A., Sethian, J., Wild, S., Willcox, K., and Lee, S.: Workshop Report on Basic Research Needs for Scientific Machine Learning: Core Technologies for Artificial Intelligence, Tech. Rep. 1478744, https://doi.org/10.2172/1478744, 2019.

1060   Ballarin, F., Sartori, A., and Rozza, G.: RBniCS-reduced order modelling in FEniCS, http://mathlab.sissa.it/rbnics, 2017.

Bandai, T. and Ghezzehei, T. A.: Physics-Informed Neural Networks With Monotonicity Constraints for Richardson-Richards Equation: Estimation of Constitutive Relationships and Soil Water Flux Density From Volumetric Water Content Measurements, Water Resources Research, 57, https://doi.org/10.1029/2020wr027642, 2021.

Bar-Sinai, Y., Hoyer, S., Hickey, J., and Brenner, M. P.: Learning data-driven discretizations for partial differential equations, Proceedings of
1065   the National Academy of Sciences, 116, 15 344–15 349, 2019.

Barrault, M., Maday, Y., Nguyen, N. C., and Patera, A. T.: An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations, Comptes Rendus Mathematique, 339, 667–672, 2004.

Baş, D. and Boyacı, I. H.: Modeling and optimization I: Usability of response surface methodology, Journal of food engineering, 78, 836–845, 2007.

1070   Bauer, P., Dueben, P. D., Hoefler, T., Quintino, T., Schulthess, T. C., and Wedi, N. P.: The digital revolution of Earth-system science, Nature Computational Science, 1, 104–113, 2021.

Beckingsale, D. A., Burmark, J., Hornung, R., Jones, H., Killian, W., Kunen, A. J., Pearce, O., Robinson, P., Ryujin, B. S., and Scogland, T. R.: RAJA: Portable Performance for Large-Scale Scientific Applications, in: 2019 IEEE/ACM International Workshop on Performance, Portability and Productivity in HPC (P3HPC), pp. 71–81, https://doi.org/10.1109/p3hpc49587.2019.00012, 2019.

1075   Benner, P., Gugercin, S., and Willcox, K.: A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems, SIAM Review, 57, 483–531, https://doi.org/10.1137/130932715, 2015.

Bentivoglio, R., Isufi, E., Jonkman, S. N., and Taormina, R.: Deep Learning Methods for Flood Mapping: A Review of Existing Applications and Future Research Directions, Hydrology and Earth System Sciences, https://doi.org/10.5194/hess-2021-614, 2021.

Bergen, K. J., Johnson, P. A., Maarten, V., and Beroza, G. C.: Machine learning for data-driven discovery in solid Earth geoscience, Science,
1080   363, 2019.

Bermúdez, M., Ntegeka, V., Wolfs, V., and Willems, P.: Development and Comparison of Two Fast Surrogate Models for Urban Pluvial Flood Simulations, Water Resources Management, 32, 2801–2815, https://doi.org/10.1007/s11269-018-1959-8, 2018.

Bertagna, L., Deakin, M., Guba, O., Sunderland, D., Bradley, A. M., Tezaur, I. K., Taylor, M. A., and Salinger, A. G.: HOMMEXX 1.0: a performance-portable atmospheric dynamical core for the Energy Exascale Earth System Model, Geoscientific Model Development, 12,
1085   1423–1441, https://doi.org/10.5194/gmd-12-1423-2019, 2019.

Bertsimas, D. and Tsitsiklis, J. N.: Introduction to linear optimization, vol. 6, Athena Scientific Belmont, MA, 1997.

Bezerra, M. A., Santelli, R. E., Oliveira, E. P., Villar, L. S., and Escaleira, L. A.: Response surface methodology (RSM) as a tool for optimization in analytical chemistry, Talanta, 76, 965–977, 2008.

Blöschl, G. and Sivapalan, M.: Scale issues in hydrological modelling: A review, Hydrological Processes, 9, 251–290,
1090   https://doi.org/10.1002/hyp.3360090305, 1995.

Bondyopadhyay, P. K.: Moore's law governs the silicon revolution, Proceedings of the IEEE, 86, 78–81, 1998.

Buiter, S. and Ellis, S.: SULEC: Benchmarking a new ALE finite-element code, in: EGU General Assembly Conference Abstracts, p. 7528, 2012.

Busto, S., Stabile, G., Rozza, G., and Vázquez-Cendón, M. E.: POD–Galerkin reduced order methods for combined Navier–Stokes transport equations based on a hybrid FV-FE solver, Computers & Mathematics with Applications, 79, 256–273, 2020.

Carraro, T., Geiger, M., Rorkel, S., and Rannacher, R.: Multiple Shooting and Time Domain Decomposition Methods, Springer, 2015.

Caviedes-Voullième, D., García-Navarro, P., and Murillo, J.: Verification, conservation, stability and efficiency of a finite volume method for the 1D Richards equation, Journal of Hydrology, 480, 69 – 84, 2013.

Caviedes-Voullième, D., Gerhard, N., Sikstel, A., and Müller, S.: Multiwavelet-based mesh adaptivity with Discontinuous Galerkin schemes: Exploring 2D shallow water problems, Advances in Water Resources, 138, 103 559, https://doi.org/10.1016/j.advwatres.2020.103559, 2020.

Chen, J., Hoversten, G. M., Key, K., Nordquist, G., and Cumming, W.: Stochastic inversion of magnetotelluric data using a sharp boundary parameterization and application to a geothermal site, Geophysics, 77, E265–E279, 2012.

Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K.: Neural ordinary differential equations, Advances in neural information processing systems, 31, 2018a.

Chen, T., Li, M., Li, Y., Lin, M., Wang, N., Wang, M., Xiao, T., Xu, B., Zhang, C., and Zhang, Z.: Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems, arXiv preprint arXiv:1512.01274, 2015.

Chen, W., Hesthaven, J. S., Junqiang, B., Qiu, Y., Yang, Z., and Tihao, Y.: Greedy nonintrusive reduced order model for fluid dynamics, AIAA Journal, 56, 4927–4943, 2018b.

Chen, X., Duan, J., and Karniadakis, G. E.: Learning and meta-learning of stochastic advection-diffusion-reaction systems from sparse measurements, arXiv preprint arXiv:1910.09098, 2019.

Chen, Y., Hesthaven, J. S., Maday, Y., and Rodríguez, J.: Certified reduced basis methods and output bounds for the harmonic Maxwell's equations, SIAM Journal on Scientific Computing, 32, 970–996, 2010.

Chen, Y., Simon, K., and Behrens, J.: Extending legacy climate models by adaptive mesh refinement for single-component tracer transport: a case study with ECHAM6-HAMMOZ (ECHAM6.3-HAM2.3-MOZ1.0), Geoscientific Model Development, 14, 2289–2316, https://doi.org/10.5194/gmd-14-2289-2021, 2021.

Chollet, F. et al.: Keras, https://keras.io, 2015.

Chuang, P.-Y. and Barba, L. A.: Experience report of physics-informed neural networks in fluid simulations: pitfalls and frustration, arXiv preprint arXiv:2205.14249, 2022.

Clement, V., Ferrachat, S., Fuhrer, O., Lapillonne, X., Osuna, C. E., Pincus, R., Rood, J., and Sawyer, W.: The CLAW DSL, in: PASC '18: Proceedings of the Platform for Advanced Scientific Computing Conference, ACM, https://doi.org/10.1145/3218176.3218226, 2018.

Condon, L. E., Kollet, S., Bierkens, M. F., Fogg, G. E., Maxwell, R. M., Hill, M. C., Fransen, H.-J. H., Verhoef, A., Van Loon, A. F., Sulis, M., et al.: Global groundwater modeling and monitoring: Opportunities and challenges, Water Resources Research, 57, e2020WR029 500, 2021.

Conway, D., Simpson, J., Didana, Y., Rugari, J., and Heinson, G.: Probabilistic magnetotelluric inversion with adaptive regularisation using the no-U-turns sampler, Pure and Applied Geophysics, 175, 2881–2894, 2018.

Cressie, N.: The origins of kriging, Mathematical geology, 22, 239–252, 1990.

Dauxois, T., Peacock, T., Bauer, P., Caulfield, C. P., Cenedese, C., Gorlé, C., Haller, G., Ivey, G. N., Linden, P. F., Meiburg, E., Pinardi, N., Vriend, N. M., and Woods, A. W.: Confronting Grand Challenges in environmental fluid mechanics, Physical Review Fluids, 6, https://doi.org/10.1103/physrevfluids.6.020501, 2021.

Dawson, C., Trahan, C. J., Kubatko, E. J., and Westerink, J. J.: A parallel local timestepping Runge–Kutta discontinuous Galerkin method with applications to coastal ocean modeling, Computer Methods in Applied Mechanics and Engineering, 259, 154–165, https://doi.org/10.1016/j.cma.2013.03.015, 2013.

Dazzi, S., Vacondio, R., Palù, A. D., and Mignosa, P.: A local time stepping algorithm for GPU-accelerated 2D shallow water models, Advances in Water Resources, 111, 274–288, https://doi.org/10.1016/j.advwatres.2017.11.023, 2018.

De Bézenac, E., Pajot, A., and Gallinari, P.: Deep learning for physical processes: Incorporating prior scientific knowledge, Journal of Statistical Mechanics: Theory and Experiment, 2019, 124 009, 2019.

Degen, D. and Cacace, M.: Effects of transient processes for thermal simulations of the Central European Basin, Geoscientific Model Development, 14, 1699–1719, 2021.

Degen, D., Veroy, K., and Wellmann, F.: Certified reduced basis method in geosciences, Computational Geosciences, 24, 241–259, 2020a.

Degen, D., Veroy, K., and Wellmann, F.: cgre-aachen/DwarfElephant: DwarfElephant 1.0, https://doi.org/10.5281/zenodo.4074777, 2020b.

Degen, D., Spooner, C., Scheck-Wenderoth, M., and Cacace, M.: How biased are our models?–a case study of the alpine region, Geoscientific Model Development, 14, 7133–7153, 2021a.

Degen, D., Veroy, K., Freymark, J., Scheck-Wenderoth, M., Poulet, T., and Wellmann, F.: Global sensitivity analysis to optimize basin-scale conductive model calibration–A case study from the Upper Rhine Graben, Geothermics, 95, 102 143, 2021b.

Degen, D., Cacace, M., and Wellmann, F.: 3D multi-physics uncertainty quantification using physics-based machine learning, Scientific Reports, 12, 17 491, https://doi.org/10.1038/s41598-022-21739-7, 2022a.

Degen, D., Veroy, K., Scheck-Wenderoth, M., and Wellmann, F.: Crustal-scale thermal models: Revisiting the influence of deep boundary conditions, Environmental Earth Sciences, 81, 88, 2022b.

Degen, D., Veroy, K., and Wellmann, F.: Uncertainty quantification for basin-scale geothermal conduction models, Scientific reports, 12, 1–10, 2022c.

Degen, D. M.: Application of the reduced basis method in geophysical simulations: concepts, implementation, advantages, and limitations, Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, Aachen, https://doi.org/10.18154/RWTH-2020-12042, veröffentlicht auf dem Publikationsserver der RWTH Aachen University 2021; Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2020, 2020.

Dwivedi, V. and Srinivasan, B.: Solution of Biharmonic Equation in Complicated Geometries With Physics Informed Extreme Learning Machine, Journal of Computing and Information Science in Engineering, 20, 2020.

Edwards, H. C., Trott, C. R., and Sunderland, D.: Kokkos: Enabling manycore performance portability through polymorphic memory access patterns, Journal of Parallel and Distributed Computing, 74, 3202 – 3216, https://doi.org/https://doi.org/10.1016/j.jpdc.2014.07.003, domain-Specific Languages and High-Level Frameworks for High-Performance Computing, 2014.

Ellis, S., Little, T., Wallace, L., Hacker, B., and Buiter, S.: Feedback between rifting and diapirism can exhume ultrahigh-pressure rocks, Earth and Planetary Science Letters, 311, 427–438, 2011.

Emmett, M. and Minion, M.: Toward an efficient parallel in time method for partial differential equations, Communications in Applied Mathematics and Computational Science, 7, 105–132, 2012.

1165   Emmett, M. and Minion, M. L.: Efficient implementation of a multi-level parallel in time algorithm, in: Domain Decomposition Methods in Science and Engineering XXI, pp. 359–366, Springer, 2014.

Evans, T. M., Siegel, A., Draeger, E. W., Deslippe, J., Francois, M. M., Germann, T. C., Hart, W. E., and Martin, D. F.: A survey of software implementations used by application codes in the Exascale Computing Project, The International Journal of High Performance Computing Applications, p. 109434202110289, https://doi.org/10.1177/10943420211028940, 2021.

1170   Falkner, S., Klein, A., and Hutter, F.: BOHB: Robust and efficient hyperparameter optimization at scale, in: International Conference on Machine Learning, pp. 1437–1446, PMLR, 2018.

Fan, Y., Clark, M., Lawrence, D. M., Swenson, S., Band, L. E., Brantley, S. L., Brooks, P. D., Dietrich, W. E., Flores, A., Grant, G., Kirchner, J. W., Mackay, D. S., McDonnell, J. J., Milly, P. C. D., Sullivan, P. L., Tague, C., Ajami, H., Chaney, N., Hartmann, A., Hazenberg, P., McNamara, J., Pelletier, J., Perket, J., Rouholahnejad-Freund, E., Wagener, T., Zeng, X., Beighley, E., Buzan, J., Huang, M., Livneh, B.,

1175   Mohanty, B. P., Nijssen, B., Safeeq, M., Shen, C., van Verseveld, W., Volk, J., and Yamazaki, D.: Hillslope Hydrology in Global Change Research and Earth System Modeling, Water Resources Research, https://doi.org/10.1029/2018wr023903, 2019.

Faroughi, S. A., Pawar, N., Fernandes, C., Das, S., Kalantari, N. K., and Mahjour, S. K.: Physics-guided, physics-informed, and physics-encoded neural networks in scientific computing, arXiv preprint arXiv:2211.07377, 2022.

Fisher, M. and Gürol, S.: Parallelization in the time dimension of four-dimensional variational data assimilation, Quarterly Journal of the

1180   Royal Meteorological Society, 143, 1136–1147, 2017.

Forsyth, P. A., Wu, Y. S., and Pruess, K.: Robust numerical methods for saturated-unsaturated flow with dry initial conditions in heterogeneous media, Advances in Water Resources, 18, 25 – 38, 1995.

Frangos, M., Marzouk, Y., Willcox, K., and van Bloemen Waanders, B.: Surrogate and reduced-order modeling: a comparison of approaches for large-scale statistical inverse problems [Chapter 7], 2010.

1185   Frank, D. J., Dennard, R. H., Nowak, E., Solomon, P. M., Taur, Y., and Wong, H.-S. P.: Device scaling limits of Si MOSFETs and their application dependencies, Proceedings of the IEEE, 89, 259–288, 2001.

Freymark, J., Bott, J., Cacace, M., Ziegler, M., and Scheck-Wenderoth, M.: Influence of the Main Border Faults on the 3D Hydraulic Field of the Central Upper Rhine Graben, Geofluids, 2019, 2019.

Gan, L., Fu, H., and Yang, G.: Translating novel HPC techniques into efficient geoscience solutions, Journal of Computational Science, p.

1190   101212, https://doi.org/10.1016/j.jocs.2020.101212, 2020.

Gassner, G. J. and Winters, A. R.: A Novel Robust Strategy for Discontinuous Galerkin Methods in Computational Fluid Mechanics: Why? When? What? Where?, Frontiers in Physics, 8, https://doi.org/10.3389/fphy.2020.500690, 2021.

Gelet, R., Loret, B., and Khalili, N.: A thermo-hydro-mechanical coupled model in local thermal non-equilibrium for fractured HDR reservoir with double porosity, Journal of Geophysical Research: Solid Earth, 117, 2012.

1195   Geneva, N. and Zabaras, N.: Modeling the dynamics of PDE systems with physics-constrained deep auto-regressive networks, Journal of Computational Physics, 403, 109 056, 2020.

Gerhard, N., Caviedes-Voullième, D., Müller, S., and Kesserwani, G.: Multiwavelet-Based Grid Adaptation with Discontinuous Galerkin Schemes for Shallow Water Equations, Journal of Computational Physics, 301, 265–288, https://doi.org/http://dx.doi.org/10.1016/j.jcp.2015.08.030, 2015.

1200   Germann, T. C.: Co-design in the Exascale Computing Project, The International Journal of High Performance Computing Applications, 35, 503–507, https://doi.org/10.1177/10943420211059380, 2021.

Ghasemi, M. and Gildin, E.: Model order reduction in porous media flow simulation using quadratic bilinear formulation, Computational Geosciences, 20, 723–735, 2016.

Gosses, M., Nowak, W., and Wöhling, T.: Explicit treatment for Dirichlet, Neumann and Cauchy boundary conditions in POD-based reduction of groundwater models, Advances in water resources, 115, 160–171, 2018.

Götz, M., Debus, C., Coquelin, D., Krajsek, K., Comito, C., Knechtges, P., Hagemeier, B., Tarnawa, M., Hanselmann, S., Siggel, M., Basermann, A., and Streit, A.: HeAT–a Distributed and GPU-accelerated Tensor Framework for Data Analytics, in: Proceedings of the 19th IEEE International Conference on Big Data, pp. 276–288, IEEE, 2020.

Grepl, M.: Reduced-basis Approximation and A Posteriori Error Estimation for Parabolic Partial Differential Equations, Ph.D. thesis, Massachusetts Institute of Technology, 2005.

Grepl, M. A.: Model order reduction of parametrized nonlinear reaction–diffusion systems, Computers & chemical engineering, 43, 33–44, 2012.

Grete, P., Glines, F. W., and O'Shea, B. W.: K-Athena: A Performance Portable Structured Grid Finite Volume Magnetohydrodynamics Code, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, 32, 85–97, https://doi.org/10.1109/tpds.2020.3010016, 2021.

Haghighat, E. and Juanes, R.: SciANN: A Keras/TensorFlow wrapper for scientific computations and physics-informed deep learning using artificial neural networks, Computer Methods in Applied Mechanics and Engineering, 373, 113 552, 2021.

Hammerschmidt, M., Herrmann, S., Pomplun, J., Zschiedrich, L., Burger, S., and Schmidt, F.: Reduced basis method for Maxwell's equations with resonance phenomena, in: Optical Systems Design 2015: Computational Optics, vol. 9630, pp. 138–151, SPIE, 2015.

Hamon, F. P., Schreiber, M., and Minion, M. L.: Parallel-in-time multi-level integration of the shallow-water equations on the rotating sphere, Journal of Computational Physics, 407, 109 210, 2020.

He, Q., Barajas-Solano, D., Tartakovsky, G., and Tartakovsky, A. M.: Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport, Advances in Water Resources, 141, 103 610, https://doi.org/10.1016/j.advwatres.2020.103610, 2020.

Herman, J. and Usher, W.: SALib: An open-source Python library for Sensitivity Analysis, J. Open Source Softw, 2, 97, 2017.

Hess, M. W. and Benner, P.: Fast evaluation of time–harmonic Maxwell's equations using the reduced basis method, IEEE Transactions on Microwave Theory and Techniques, 61, 2265–2274, 2013a.

Hess, M. W. and Benner, P.: Reduced basis generation for maxwell's equations by rigorous error estimation, in: 19th International Conference on the Computation of Electromagnetic Fields (COMPUMAG 2013), pp. PD2–10, 2013b.

Hess, M. W., Grundel, S., and Benner, P.: Estimating the inf-sup constant in reduced basis methods for time-harmonic Maxwell's equations, IEEE Transactions on Microwave Theory and Techniques, 63, 3549–3557, 2015.

Hesthaven, J. and Ubbiali, S.: Non-intrusive reduced order modeling of nonlinear problems using neural networks, Journal of Computational Physics, 363, 55–78, https://doi.org/10.1016/j.jcp.2018.02.037, 2018.

Hesthaven, J. S., Rozza, G., Stamm, B., et al.: Certified Reduced Basis Methods for Parametrized Partial Differential Equations, SpringerBriefs in Mathematics, Springer, 2016.

Hokkanen, J., Kollet, S., Kraus, J., Herten, A., Hrywniak, M., and Pleiter, D.: Leveraging HPC accelerator architectures with modern techniques — hydrologic modeling on GPUs with ParFlow, Computational Geosciences, https://doi.org/10.1007/s10596-021-10051-4, 2021.

Huang, P.: Investigating different formulations for hydrothermal convection in geothermal systems, Master's thesis, RWTH Aachen University, Aachen, 2018.

1240 Iglesias, M. and Stuart, A. M.: Inverse Problems and Uncertainty Quantification, SIAM News, pp. 2–3, 2014.

Jacquey, A. and Cacace, M.: Modelling fully-coupled Thermo-Hydro-Mechanical (THM) processes in fractured reservoirs using GOLEM: a massively parallel open-source simulator, in: EGU General Assembly Conference Abstracts, vol. 19, p. 15721, 2017.

Jacquey, A. B. and Cacace, M.: Multiphysics modeling of a brittle-ductile lithosphere: 2. Semi-brittle, semi-ductile deformation and damage rheology, Journal of Geophysical Research: Solid Earth, 125, e2019JB018 475, 2020a.

1245 Jacquey, A. B. and Cacace, M.: Multiphysics Modeling of a Brittle-Ductile Lithosphere: 1. Explicit Visco-Elasto-Plastic Formulation and Its Numerical Implementation, Journal of Geophysical Research: Solid Earth, 125, e2019JB018 474, 2020b.

Jagtap, A. D. and Karniadakis, G. E.: Extended Physics-Informed Neural Networks (XPINNs): A Generalized Space-Time Domain Decomposition Based Deep Learning Framework for Nonlinear Partial Differential Equations, Communications in Computational Physics, 28, 2002–2041, 2020.

1250 Jiang, S., Zheng, Y., and Solomatine, D.: Improving AI System Awareness of Geoscience Knowledge: Symbiotic Integration of Physical Approaches and Deep Learning, Geophysical Research Letters, 47, https://doi.org/10.1029/2020gl088229, 2020.

Jones, A. G., Afonso, J. C., and Fullea, J.: Geochemical and geophysical constrains on the dynamic topography of the S outhern A frican P lateau, Geochemistry, Geophysics, Geosystems, 18, 3556–3575, 2017.

Jordan, M. I. and Mitchell, T. M.: Machine learning: Trends, perspectives, and prospects, Science, 349, 255–260, 2015.

1255 Jung, N.: Error Estimation for Parametric Model Order Reduction and Its Application, VDI Verlag, 2012.

Karumuri, S., Tripathy, R., Bilionis, I., and Panchal, J.: Simulator-free solution of high-dimensional stochastic elliptic partial differential equations using deep neural networks, Journal of Computational Physics, 404, 109 120, 2020.

Kevlahan, N. K.-R.: Adaptive Wavelet Methods for Earth Systems Modelling, Fluids, 6, 236, https://doi.org/10.3390/fluids6070236, 2021.

Khuri, A. I. and Mukhopadhyay, S.: Response surface methodology, Wiley Interdisciplinary Reviews: Computational Statistics, 2, 128–149, 1260 2010.

Kohl, T., Evansi, K., Hopkirk, R., and Rybach, L.: Coupled hydraulic, thermal and mechanical considerations for the simulation of hot dry rock reservoirs, Geothermics, 24, 345–359, 1995.

Kolditz, O., Görke, U.-J., Shao, H., and Wang, W.: Thermo-hydro-mechanical-chemical processes in porous media: benchmarks and examples, vol. 86, Springer Science & Business Media, 2012.

1265 Koltzer, N., Scheck-Wenderoth, M., Bott, J., Cacace, M., Frick, M., Sass, I., Fritsche, J.-G., and Bär, K.: The Effects of Regional Fluid Flow on Deep Temperatures (Hesse, Germany), Energies, 12, 2081, 2019.

Kotsiantis, S. B., Zaharakis, I., Pintelas, P., et al.: Supervised machine learning: A review of classification techniques, Emerging artificial intelligence applications in computer engineering, 160, 3–24, 2007.

Kärnä, T., Kramer, S. C., Mitchell, L., Ham, D. A., Piggott, M. D., and Baptista, A. M.: Thetis coastal ocean model: discontin-
1270 uous Galerkin discretization for the three-dimensional hydrostatic equations, Geoscientific Model Development, 11, 4359–4382, https://doi.org/10.5194/gmd-11-4359-2018, 2018.

Käser, M. and Dumbser, M.: An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes - I. The two-dimensional isotropic case with external source terms, Geophysical Journal International, 166, 855–877, https://doi.org/10.1111/j.1365-246x.2006.03051.x, 2006.

1275 Kuffour, B. N. O., Engdahl, N. B., Woodward, C. S., Condon, L. E., Kollet, S., and Maxwell, R. M.: Simulating coupled surface–subsurface flows with ParFlow v3.5.0: capabilities, applications, and ongoing development of an open-source, massively parallel, integrated hydrologic model, Geoscientific Model Development, 13, 1373–1397, https://doi.org/10.5194/gmd-13-1373-2020, 2020.

Kumar, P., Debele, S. E., Sahani, J., Rawat, N., Marti-Cardona, B., Alfieri, S. M., Basu, B., Basu, A. S., Bowyer, P., Charizopoulos, N., et al.: An overview of monitoring methods for assessing the performance of nature-based solutions against natural hazards, Earth-Science Reviews, 217, 103 603, 2021.

Lawrence, B. N., Rezny, M., Budich, R., Bauer, P., Behrens, J., Carter, M., Deconinck, W., Ford, R., Maynard, C., Mullerworth, S., Osuna, C., Porter, A., Serradell, K., Valcke, S., Wedi, N., and Wilson, S.: Crossing the chasm: how to develop weather and climate models for next generation computers?, Geoscientific Model Development, 11, 1799–1821, https://doi.org/10.5194/gmd-11-1799-2018, 2018.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A.: Fourier neural operator for parametric partial differential equations, arXiv preprint arXiv:2010.08895, 2020.

Liang, J., Li, W., Bradford, S., and Šimůnek, J.: Physics-Informed Data-Driven Models to Predict Surface Runoff Water Quantity and Quality in Agricultural Fields, Water, 11, 200, https://doi.org/10.3390/w11020200, 2019.

Lions, J.-L., Maday, Y., and Turinici, G.: Résolution d'EDP par un schéma en temps pararéel, Comptes Rendus de l'Académie des Sciences-Series I-Mathematics, 332, 661–668, 2001.

Liu, W. and Ramirez, A.: State of the art review of the environmental assessment and risks of underground geo-energy resources exploitation, Renewable and Sustainable Energy Reviews, 76, 628–644, 2017.

Lu, L., Meng, X., Mao, Z., and Karniadakis, G. E.: DeepXDE: A deep learning library for solving differential equations, arXiv preprint arXiv:1907.04502, 2019.

Lu, L., Jin, P., Pang, G., Zhang, Z., and Karniadakis, G. E.: Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators, Nature machine intelligence, 3, 218–229, 2021.

Lu, L., Meng, X., Cai, S., Mao, Z., Goswami, S., Zhang, Z., and Karniadakis, G. E.: A comprehensive and fair comparison of two neural operators (with practical extensions) based on fair data, Computer Methods in Applied Mechanics and Engineering, 393, 114 778, 2022.

Maday, Y.: The parareal in time algorithm, 2008.

Mahesh, B.: Machine learning algorithms-a review, International Journal of Science and Research (IJSR).[Internet], 9, 381–386, 2020.

Mainini, L. and Willcox, K.: Surrogate modeling approach to support real-time structural assessment and decision making, AIAA Journal, 53, 1612–1626, 2015.

Malek, A. and Beidokhti, R. S.: Numerical solution for high order differential equations using a hybrid neural network—optimization method, Applied Mathematics and Computation, 183, 260–271, 2006.

Manassero, M. C., Afonso, J. C., Zyserman, F., Zlotnik, S., and Fomin, I.: A reduced order approach for probabilistic inversions of 3-D magnetotelluric data I: general formulation, Geophysical Journal International, 223, 1837–1863, 2020.

Maxwell, R. M., Condon, L. E., and Melchior, P.: A Physics-Informed, Machine Learning Emulator of a 2D Surface Water Model: What Temporal Networks and Simulation-Based Inference Can Help Us Learn about Hydrologic Processes, Water, 13, 3633, https://doi.org/10.3390/w13243633, 2021.

Meng, X. and Karniadakis, G. E.: A composite neural network that learns from multi-fidelity data: Application to function approximation and inverse PDE problems, Journal of Computational Physics, 401, 109 020, 2020.

Meng, X., Li, Z., Zhang, D., and Karniadakis, G. E.: PPINN: Parareal physics-informed neural network for time-dependent PDEs, Computer Methods in Applied Mechanics and Engineering, 370, 113 250, 2020.

Miao, T., Lu, W., Lin, J., Guo, J., and Liu, T.: Modeling and uncertainty analysis of seawater intrusion in coastal aquifers using a surrogate model: a case study in Longkou, China, Arabian Journal of Geosciences, 12, 1, 2019.

Mo, S., Shi, X., Lu, D., Ye, M., and Wu, J.: An adaptive Kriging surrogate method for efficient uncertainty quantification with an application to geological carbon sequestration modeling, Computers & Geosciences, 2019.

Mohammadi, B.: A review on the applications of machine learning for runoff modeling, Sustainable Water Resources Management, 7, https://doi.org/10.1007/s40899-021-00584-y, 2021.

Morales-Hernández, M., Sharif, M. B., Gangrade, S., Dullo, T. T., Kao, S.-C., Kalyanapu, A., Ghafoor, S. K., Evans, K. J., Madadi-Kandjani, E., and Hodges, B. R.: High-performance computing in water resources hydrodynamics, Journal of Hydroinformatics, https://doi.org/10.2166/hydro.2020.163, 2020.

Myers, R. H., Montgomery, D. C., and Anderson-Cook, C. M.: Response Surface Methodology: Process and Product Optimization Using Designed Experiments, John Wiley & Sons, 2016.

Naliboff, J. B., Buiter, S. J., Péron-Pinvidic, G., Osmundsen, P. T., and Tetreault, J.: Complex fault interaction controls continental rifting, Nature communications, 8, 1–9, 2017.

Navarro, M., Le Maître, O. P., Hoteit, I., George, D. L., Mandli, K. T., and Knio, O. M.: Surrogate-based parameter inference in debris flow model, Computational Geosciences, 22, 1447–1463, 2018.

Nearing, G. S., Kratzert, F., Sampson, A. K., Pelissier, C. S., Klotz, D., Frame, J. M., Prieto, C., and Gupta, H. V.: What Role Does Hydrological Science Play in the Age of Machine Learning?, Water Resources Research, 57, https://doi.org/10.1029/2020wr028091, 2021.

Nield, D. A. and Bejan, A.: Heat Transfer Through a Porous Medium, in: Convection in Porous Media, pp. 37–55, Springer, 2017.

Ortega-Gelabert, O., Zlotnik, S., Afonso, J. C., and Díez, P.: Fast stokes flow simulations for geophysical-geodynamic inverse problems and sensitivity analyses based on reduced order modeling, Journal of Geophysical Research: Solid Earth, 125, e2019JB018 314, 2020.

O'Sullivan, M. J., Pruess, K., and Lippmann, M. J.: State of the art of geothermal reservoir simulation, Geothermics, 30, 395–429, 2001.

Özgen-Xian, I., Kesserwani, G., Caviedes-Voullième, D., Molins, S., Xu, Z., Dwivedi, D., Moulton, J. D., and Steefel, C. I.: Wavelet-based local mesh refinement for rainfall–runoff simulations, Journal of Hydroinformatics, https://doi.org/10.2166/hydro.2020.198, 2020.

Pang, G., Lu, L., and Karniadakis, G. E.: fPINNs: Fractional physics-informed neural networks, SIAM Journal on Scientific Computing, 41, A2603–A2626, 2019.

Pang, G., D'Elia, M., Parks, M., and Karniadakis, G. E.: nPINNs: nonlocal Physics-Informed Neural Networks for a parametrized nonlocal universal Laplacian operator. Algorithms and Applications, Journal of Computational Physics, 422, 109 760, 2020.

Paniconi, C. and Putti, M.: Physically based modeling in catchment hydrology at 50: Survey and outlook, Water Resources Research, 51, 7090–7129, https://doi.org/10.1002/2015WR017780, 2015.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S.: PyTorch: An Imperative Style, High-Performance Deep Learning Library, in: Advances in Neural Information Processing Systems 32, edited by Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., pp. 8024–8035, Curran Associates, Inc., http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf, 2019.

Patil, A., Huard, D., and Fonnesbeck, C. J.: PyMC: Bayesian Stochastic Modelling in Python, Journal of statistical software, 35, 1, 2010.

Peherstorfer, B. and Willcox, K.: Data-driven operator inference for nonintrusive projection-based model reduction, Computer Methods in Applied Mechanics and Engineering, 306, 196–215, 2016a.

Peherstorfer, B. and Willcox, K.: Data-driven operator inference for nonintrusive projection-based model reduction, Computer Methods in Applied Mechanics and Engineering, 306, 196–215, https://doi.org/10.1016/j.cma.2016.03.025, 2016b.

Peherstorfer, B., Willcox, K., and Gunzburger, M.: Survey of multifidelity methods in uncertainty propagation, inference, and optimization, Siam Review, 60, 550–591, 2018.

1355   Pelletier, D., Fortin, A., and Camarero, R.: Are FEM solutions of incompressible flows really incompressible?(or how simple flows can cause headaches!), International Journal for Numerical Methods in Fluids, 9, 99–112, 1989.

Peng, W., Zhou, W., Zhang, J., and Yao, W.: Accelerating physics-informed neural network training with prior dictionaries, arXiv preprint arXiv:2004.08151, 2020.

Permann, C. J., Gaston, D. R., Andrš, D., Carlsen, R. W., Kong, F., Lindsay, A. D., Miller, J. M., Peterson, J. W., Slaughter,
1360   A. E., Stogner, R. H., and Martineau, R. C.: MOOSE: Enabling massively parallel multiphysics simulation, SoftwareX, 11, 100 430, https://doi.org/https://doi.org/10.1016/j.softx.2020.100430, 2020.

Piggott, M., Pain, C., Gorman, G., Power, P., and Goddard, A.: h, r, and hr adaptivity with applications in numerical ocean modelling, Ocean Modelling, 10, 95–113, https://doi.org/10.1016/j.ocemod.2004.07.007, 2005.

Poulet, T., Veveakis, M., Paesold, M., and Regenauer-Lieb, K.: REDBACK: An open-source highly scalable simulation tool for rock me-
1365   chanics with dissipative feedbacks, in: AGU Fall Meeting Abstracts, 2014.

Prud'homme, C., Rovas, D. V., Veroy, K., Machiels, L., Maday, Y., Patera, A. T., and Turinici, G.: Reliable Real-Time Solution of Parametrized Partial Differential Equations: Reduced-Basis Output Bound Methods, Journal of Fluids Engineering, 124, 70–80, 2002.

Qian, E., Grepl, M., Veroy, K., and Willcox, K.: A certified trust region reduced basis approach to PDE-constrained optimization, SIAM Journal on Scientific Computing, 39, S434–S460, 2017.

1370   Quarteroni, A., Manzoni, A., and Negri, F.: Reduced Basis Methods for Partial Differential Equations: An Introduction, UNITEXT, Springer International Publishing, 2015.

Rabiti, C., Alfonsi, A., Cogliati, J., Mandelli, D., Kinoshita, R., Sen, S., Wang, C., Talbot, P. W., Maljovec, D. P., and Chen, J.: RAVEN user manual, Tech. rep., Idaho National Lab.(INL), Idaho Falls, ID (United States), 2020.

Raissi, M. and Karniadakis, G. E.: Hidden physics models: Machine learning of nonlinear partial differential equations, Journal of Compu-
1375   tational Physics, 357, 125–141, 2018.

Raissi, M., Perdikaris, P., and Karniadakis, G. E.: Numerical Gaussian processes for time-dependent and non-linear partial differential equations, arXiv preprint arXiv:1703.10230, 2017.

Raissi, M., Perdikaris, P., and Karniadakis, G. E.: 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.

1380   Rannabauer, L., Dumbser, M., and Bader, M.: ADER-DG with a-posteriori finite-volume limiting to simulate tsunamis in a parallel adaptive mesh refinement framework, Computers & Fluids, 173, 299–306, https://doi.org/10.1016/j.compfluid.2018.01.031, 2018.

Rao, C., Sun, H., and Liu, Y.: Hard encoding of physics for learning spatiotemporal dynamics, arXiv preprint arXiv:2105.00557, 2021.

Rathgeber, F., Ham, D. A., Mitchell, L., Lange, M., Luporini, F., McRae, A. T., Bercea, G.-T., Markall, G. R., and Kelly, P. H.: Firedrake: automating the finite element method by composing abstractions, ACM Transactions on Mathematical Software (TOMS), 43, 1–27, 2016.

1385   Razavi, S., Tolson, B. A., and Burn, D. H.: Review of surrogate modeling in water resources, Water Resources Research, 48, 2012.

Regenauer-Lieb, K., Veveakis, M., Poulet, T., Wellmann, F., Karrech, A., Liu, J., Hauser, J., Schrank, C., Gaede, O., and Trefry, M.: Multiscale coupling and multiphysics approaches in earth sciences: Theory, Journal of Coupled Systems and Multiscale Dynamics, 1, 49–73, 2013.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., et al.: Deep learning and process understanding for data-driven Earth system science, Nature, 566, 195–204, 2019.

1390   Richards, L. A.: Capillary conduction of liquids through porous mediums, Physics, 1, 318–333, 1931.

Rizzo, C. B., de Barros, F. P., Perotto, S., Oldani, L., and Guadagnini, A.: Adaptive POD model reduction for solute transport in heterogeneous porous media, Computational Geosciences, pp. 1–12, 2017.

Rochlitz, R., Skibbe, N., and Günther, T.: custEM: Customizable finite-element simulation of complex controlled-source electromagnetic data, GEOPHYSICS, 84, F17–F33, https://doi.org/10.1190/geo2018-0208.1, 2019.

Rogelj, J. and Knutti, R.: Geosciences after Paris, Nature Geoscience, 9, 187–189, 2016.

Rosas-Carbajal, M., Linde, N., Kalscheuer, T., and Vrugt, J. A.: Two-dimensional probabilistic inversion of plane-wave electromagnetic data: methodology, model constraints and joint inversion with electrical resistivity data, Geophysical Journal International, 196, 1508–1524, 2014.

Rousset, M. A., Huang, C. K., Klie, H., and Durlofsky, L. J.: Reduced-order modeling for thermal recovery processes, Computational Geosciences, 18, 401–415, 2014.

Sabeena, J. and Reddy, P. V. S.: A Review of Weather forecasting schemes, i-manager's Journal on Pattern Recognition, 4, 27, 2017.

Santoso, R., Degen, D., Cacace, M., and Wellmann, F.: State-of-the-art physics-based machine learning for hydro-mechanical simulation in geothermal applications, European Geothermal Congress 2022, pp. 1–10, 2022.

Schaa, R., Gross, L., and du Plessis, J.: PDE-based geophysical modelling using finite elements: examples from 3D resistivity and 2D magnetotellurics, Journal of Geophysics and Engineering, 13, S59–S73, https://doi.org/10.1088/1742-2132/13/2/S59, 2016.

Schenk, O. and Gärtner, K.: Solving unsymmetric sparse systems of linear equations with PARDISO, Future Generation Computer Systems, 20, 475–487, 2004.

Shah, V., Joshi, A., Ghosal, S., Pokuri, B., Sarkar, S., Ganapathysubramanian, B., and Hegde, C.: Encoding invariances in deep generative models, arXiv preprint arXiv:1906.01626, 2019.

Sharma, R., Farimani, A. B., Gomes, J., Eastman, P., and Pande, V.: Weakly-supervised deep learning of heat transport via physics informed loss, arXiv preprint arXiv:1807.11374, 2018.

Shen, C., Chen, X., and Laloy, E.: Editorial: Broadening the Use of Machine Learning in Hydrology, Frontiers in Water, 3, https://doi.org/10.3389/frwa.2021.681023, 2021.

Sit, M., Demiray, B. Z., Xiang, Z., Ewing, G. J., Sermet, Y., and Demir, I.: A comprehensive review of deep learning applications in hydrology and water resources, Water Science and Technology, 82, 2635–2670, https://doi.org/10.2166/wst.2020.369, 2020.

Sobol, I. M.: Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates, Mathematics and computers in simulation, 55, 271–280, 2001.

Song, X., Zhang, J., Zhan, C., Xuan, Y., Ye, M., and Xu, C.: Global sensitivity analysis in hydrological modeling: Review of concepts, methods, theoretical framework, and applications, Journal of hydrology, 523, 739–757, 2015.

Stefansson, I., Berre, I., and Keilegavlen, E.: A fully coupled numerical model of thermo-hydro-mechanical processes and fracture contact mechanics in porous media, arXiv preprint arXiv:2008.06289, 2020.

Stewart, I. S. and Lewis, D.: Communicating contested geoscience to the public: Moving from 'matters of fact'to 'matters of concern', Earth-Science Reviews, 174, 122–133, 2017.

Swischuk, R., Mainini, L., Peherstorfer, B., and Willcox, K.: Projection-based model reduction: Formulations for physics-based machine learning, Computers & Fluids, 179, 704–717, https://doi.org/10.1016/j.compfluid.2018.07.021, 2019.

Taron, J. and Elsworth, D.: Thermal–hydrologic–mechanical–chemical processes in the evolution of engineered geothermal reservoirs, International Journal of Rock Mechanics and Mining Sciences, 46, 855–864, 2009.

Taron, J., Elsworth, D., and Min, K.-B.: Numerical simulation of thermal-hydrologic-mechanical-chemical processes in deformable, fractured porous media, International Journal of Rock Mechanics and Mining Sciences, 46, 842–854, 2009.

1430 Tartakovsky, A. M., Marrero, C. O., Perdikaris, P., Tartakovsky, G. D., and Barajas-Solano, D.: Physics-Informed Deep Neural Networks for Learning Parameters and Constitutive Relationships in Subsurface Flow Problems, Water Resources Research, 56, https://doi.org/10.1029/2019wr026731, 2020.

Team, T. T. D., Al-Rfou, R., Alain, G., Almahairi, A., Angermueller, C., Bahdanau, D., Ballas, N., Bastien, F., Bayer, J., Belikov, A., et al.: Theano: A Python framework for fast computation of mathematical expressions, arXiv preprint arXiv:1605.02688, 2016.

1435 Tran, H., Leonarduzzi, E., la Fuente, L. D., Hull, R. B., Bansal, V., Chennault, C., Gentine, P., Melchior, P., Condon, L. E., and Maxwell, R. M.: Development of a Deep Learning Emulator for a Distributed Groundwater–Surface Water Model: ParFlow-ML, Water, 13, 3393, https://doi.org/10.3390/w13233393, 2021.

Trott, C., Berger-Vergiat, L., Poliakoff, D., Rajamanickam, S., Lebrun-Grandie, D., Madsen, J., Awar, N. A., Gligoric, M., Shipman, G., and Womeldorff, G.: The Kokkos EcoSystem: Comprehensive Performance Portability for High Performance Computing, Computing in
1440   Science & Engineering, 23, 10–18, https://doi.org/10.1109/mcse.2021.3098509, 2021.

Vallier, B., Magnenet, V., Schmittbuhl, J., and Fond, C.: THM modeling of hydrothermal circulation at Rittershoffen geothermal site, France, Geothermal Energy, 6, 22, 2018.

van Genuchten, M.: A closed-form equation for predicting the hydraulic conductivity of unsaturated soils, Soil Science Society of America Journal, 44, 892–898, 1980.

1445 van Zelst, I., Crameri, F., Pusok, A. E., Glerum, A., Dannberg, J., and Thieulot, C.: 101 geodynamic modelling: how to design, interpret, and communicate numerical studies of the solid Earth, Solid Earth, 13, 583–637, https://doi.org/10.5194/se-13-583-2022, 2022.

Vermeulen, P. and Heemink, A.: Model-reduced variational data assimilation, Monthly weather review, 134, 2888–2899, 2006.

Vermeulen, P., Heemink, A. W., and Te Stroet, C. B.: Reduced models for linear groundwater flow models using empirical orthogonal functions, Advances in water resources, 27, 57–69, 2004.

1450 Veroy, K., Prud'homme, C., Rovas, D. V., and Patera, A. T.: A Posteriori Error Bounds for Reduced-Basis Approximation of Parametrized Noncoercive and Nonlinear Elliptic Partial Differential Equations, in: Proceedings of the 16th AIAA computational fluid dynamics conference, vol. 3847, pp. 23–26, Orlando, FL, 2003.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat,
1455   İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nature Methods, 17, 261–272, https://doi.org/10.1038/s41592-019-0686-2, 2020.

Volkwein, S.: Model reduction using proper orthogonal decomposition, Lecture Notes, Institute of Mathematics and Scientific Computing, University of Graz. see http://www. uni-graz. at/imawww/volkwein/POD. pdf, 2011.

1460 Wainwright, H. M., Finsterle, S., Jung, Y., Zhou, Q., and Birkholzer, J. T.: Making sense of global sensitivity analyses, Computers & Geosciences, 65, 84–94, 2014.

Wang, Q., Hesthaven, J. S., and Ray, D.: Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem, Journal of Computational Physics, 384, 289–307, https://doi.org/10.1016/j.jcp.2019.01.031, 2019.

Wang, S., Yu, X., and Perdikaris, P.: When and why PINNs fail to train: A neural tangent kernel perspective, Journal of Computational
1465   Physics, 449, 110 768, 2022.

Wellmann, J. F. and Reid, L. B.: Basin-scale geothermal model calibration: Experience from the Perth Basin, Australia, Energy Procedia, 59, 382–389, 2014.

Willard, J., Jia, X., Xu, S., Steinbach, M., and Kumar, V.: Integrating Physics-Based Modeling with Machine Learning: A Survey, arXiv:2003.04919 [physics, stat], http://arxiv.org/abs/2003.04919, arXiv: 2003.04919, 2020.

Willcox, K. E., Ghattas, O., and Heimbach, P.: The imperative of physics-based modeling and inverse theory in computational science, Nature Computational Science, 1, 166–168, 2021.

Wirtz, D., Karajan, N., and Haasdonk, B.: Surrogate modeling of multiscale models using kernel methods, International Journal for Numerical Methods in Engineering, 101, 1–28, 2015.

Wu, J.-L., Kashinath, K., Albert, A., Chirila, D., Xiao, H., et al.: Enforcing statistical constraints in generative adversarial networks for modeling chaotic dynamical systems, Journal of Computational Physics, 406, 109 209, 2020.

Yang, L., Zhang, D., and Karniadakis, G. E.: Physics-informed generative adversarial networks for stochastic differential equations, arXiv preprint arXiv:1811.02033, 2018.

Yang, L., Meng, X., and Karniadakis, G. E.: B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data, Journal of Computational Physics, 425, 109 913, 2020.

Yang, Y. and Perdikaris, P.: Physics-informed deep generative models, arXiv preprint arXiv:1812.03511, 2018.

Young, C.-C., Liu, W.-C., and Wu, M.-C.: A physically based and machine learning hybrid approach for accurate rainfall-runoff modeling during extreme typhoon events, Applied Soft Computing, 53, 205–216, https://doi.org/10.1016/j.asoc.2016.12.052, 2017.

Yu, M., Yang, C., and Li, Y.: Big data in natural disaster management: a review, Geosciences, 8, 165, 2018.

Zahura, F. T., Goodall, J. L., Sadler, J. M., Shen, Y., Morsy, M. M., and Behl, M.: Training Machine Learning Surrogate Models From a High-Fidelity Physics-Based Model: Application for Real-Time Street-Scale Flood Prediction in an Urban Coastal Community, Water Resources Research, 56, https://doi.org/10.1029/2019wr027038, 2020.

Zanchetta, A. D. L. and Coulibaly, P.: Hybrid Surrogate Model for Timely Prediction of Flash Flood Inundation Maps Caused by Rapid River Overflow, Forecasting, 4, 126–148, https://doi.org/10.3390/forecast4010007, 2022.

Zenker, E., Worpitz, B., Widera, R., Huebl, A., Juckeland, G., Knüpfer, A., Nagel, W. E., and Bussmann, M.: Alpaka - An Abstraction Library for Parallel Kernel Acceleration, IEEE Computer Society, http://arxiv.org/abs/1602.08477, 2016.

Zhang, E., Dao, M., Karniadakis, G. E., and Suresh, S.: Analyses of internal structures and defects in materials using physics-informed neural networks, Science advances, 8, eabk0644, 2022.

Zhu, Y., Zabaras, N., Koutsourelakis, P.-S., and Perdikaris, P.: Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data, Journal of Computational Physics, 394, 56–81, 2019.

Zounemat-Kermnai, M., Batelaan, O., Fadaee, M., and Hinkelmann, R.: Ensemble Machine Learning Paradigms in Hydrology: A Review, Journal of Hydrology, p. 126266, https://doi.org/10.1016/j.jhydrol.2021.126266, 2021.