
Multi-scale Digital Twin: Developing a fast and physics-informed surrogate model for groundwater contamination with uncertain climate models

Lijing Wang

Dept. of the Geological Sciences
Stanford University
Stanford, CA 94305
lijing52@stanford.edu

Takuya Kurihana

Dept. of Computer Science
University of Chicago
Chicago, IL 60637
tkurihana@uchicago.edu

Aurelien Meray

Dept. of Computer Science
Florida International University
Miami, FL 33133
amera009@fiu.edu

Ilijana Mastilovic

School of Freshwater Sciences
University of Wisconsin - Milwaukee
Milwaukee, WI 53204
ilijana@uwm.edu

Satyarth Praveen

Lawrence Berkeley National Lab
Berkeley, CA
satyarth@lbl.gov

Zexuan Xu

Lawrence Berkeley National Lab
Berkeley, CA
zexuanxu@lbl.gov

Milad Memarzadeh

NASA Ames Research Center
Mountain View, CA
milad.memarzadeh@nasa.gov

Alexander Lavin

Pasteur Labs & ISI
Brooklyn, NY 11205
lavin@simulation.science

Haruko Wainwright

Department of Nuclear Science and Engineering
Massachusetts Institute of Technology
Cambridge, MA
hmwainw@mit.edu

Abstract

Soil and groundwater contamination is a pervasive problem at thousands of locations across the world. Contaminated sites often require decades to remediate or to monitor natural attenuation. Climate change exacerbates the long-term site management problem because extreme precipitation and/or shifts in precipitation/evapotranspiration regimes could re-mobilize contaminants and proliferate affected groundwater. To quickly assess the spatiotemporal variations of groundwater contamination under uncertain climate disturbances, we developed a physics-informed machine learning surrogate model using U-Net enhanced Fourier Neural Operator (U-FNO) to solve Partial Differential Equations (PDEs) of groundwater flow and transport simulations at the site scale. We develop a combined loss function that includes both data-driven factors and physical boundary constraints at multiple spatiotemporal scales. Our U-FNOs can reliably predict the spatiotemporal variations of groundwater flow and contaminant transport properties from 1954

to 2100 with realistic climate projections. In parallel, we develop a convolutional autoencoder combined with online clustering to reduce the dimensionality of the vast historical and projected climate data by quantifying climatic region similarities across the United States. The ML-based unique climate clusters provide climate projections for the surrogate modeling and help return reliable future recharge rate projections immediately without querying large climate datasets. In all, this Multi-scale Digital Twin work can advance the field of environmental remediation under climate change.

1 Introduction

Soil and groundwater contamination occurs across the world with highly variable climate, geology and geographical conditions as well as with various contaminant types from industrial or agricultural activities, to nuclear and battery waste storage. These sites must be carefully managed to prevent hazardous materials from causing harm to humans, wildlife, and ecological systems. Additional challenges arise and exacerbate risks due to global climate change. Uncertain climate scenarios can impact the water balance in dynamic or disruptive the system and remobilize contaminants.

In this paper, we integrate the recent advances in machine learning (ML) and physical simulation to predict the spatiotemporal dynamics of the subsurface contamination under future climate changes for decision-makers and site managers. We incorporate multi-scale uncertainties in physical simulations: uncertain global climate projections and uncertain local soil and subsurface properties. Specifically, we develop a multi-scale digital twin with two aspects: 1) **ML-based surrogate model**: we develop a ML surrogate model using neural operator learning [10, 9, 15] to solve Partial Differential Equations (PDEs) for complex flow and transport physical simulations rapidly, and 2) **Unsupervised climate data clustering**: we perform an unsupervised clustering of climate data from all different global climate models and classify the United States into a set of representative climate regions to facilitate a rapid access to the climate data for any target location.

We first run stochastic simulations of groundwater flow and contaminant transport physics, using Amanzi [13] at the testbed: the Department of Energy’s Savannah River Site F-Area [11, 16] with uncertain subsurface, soil and climate factors. We then develop a surrogate model to address the computational challenge of solving a complex groundwater flow and contaminant transport model. In parallel, to address the one of intractable data size of global climate model datasets [8], climate data clustering quantifies uncertain projections of precipitation and evapotranspiration (ET), thereby providing climate inputs for groundwater systems without downloading large amount of climate data.

2 ML-based surrogate modeling for groundwater systems

The intersection of physics and ML provides a rich space to build models with the advantages of high-dimensional data-driven learning while maintaining (and even guiding) physical constraints and laws. One popular method is called neural operator learning [12, 10, 9], using neural network to learn mesh-independent, resolution-invariant solution operators for PDEs. Using neural operator learning, we aim to learn a fast surrogate model for groundwater systems based on the physical simulation datasets of groundwater flow and contaminant transport models. The input of our datasets includes uncertain soil, subsurface, and **climate properties (precipitation and ET: see Section 3)** $m(\mathbf{x}, t)$ for a groundwater system, the output includes the flow and transport properties $y(\mathbf{x}, t)$ such as the spatiotemporal contaminant concentration. \mathbf{x} is the location vector in 2D spatial cross-section, t is the time variable ranging from 1954 to 2100.

2.1 Architectures: U-FNO-3D and U-FNO-2D

We build two different neural operator architectures (U-FNO-3D and U-FNO-2D) to predict spatial-temporal contaminant concentration and groundwater flow properties. Both of these two architectures have Fourier Neural Operator (FNO) with enhanced U-Net architecture (U-FNO) [10, 15]. FNO has linear transformations in Fourier frequency modes so that FNO can represent any resolution with additional back transformation. U-FNO adds an additional U-Net architecture for each Fourier layer, achieving a lower test error for multi-phase flow predictions [15].

Our two architectures address the temporal dimension differently. **U-FNO-3D** takes all input time-series groundwater model parameters $m(\mathbf{x}, t)$ and predicts all contaminant concentrations for different time steps together $y(\mathbf{x}, t)$. **U-FNO-2D** uses a recurrent network: U-FNO-2D learns the physics between every fixed time interval Δt . The initial stage $t = 0$ for U-FNO-2D does not have the prediction from the previous step. Then we train another U-FNO-2D model (the green block in Figure 1b) for the initial stage $t = 0$ with the input $m(\mathbf{x}, 0)$ and the output $y(\mathbf{x}, 0)$. The advantage of U-FNO-2D is that we can preserve the time dependency, where the output $y(\mathbf{x}, t^*)$ is only determined by the input before and at time t^* . However, this additional time dependency makes the training of U-FNO-2D practically harder and hence the training takes longer time. There are also accumulated errors through the recurrent neural network.

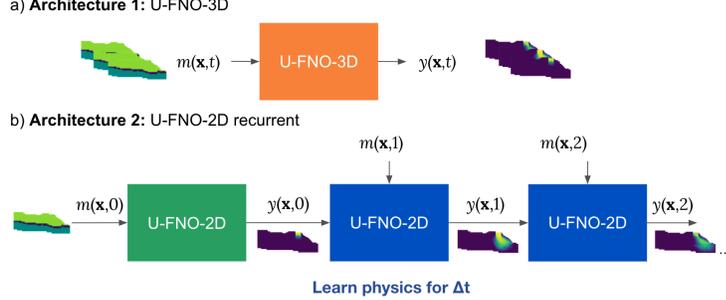


Figure 1: U-FNOs architectures

2.2 Hybrid physics-based and data-driven loss functions

We introduce four different loss functions that include both data-driven factors and physical boundary constraints. Our surrogate model predicts transient flow $\hat{h}(\mathbf{x}, t)$ and transport $\hat{c}(\mathbf{x}, t)$ properties $\hat{y}(\mathbf{x}, t) = \{\hat{h}(\mathbf{x}, t), \hat{c}(\mathbf{x}, t)\}$ where the ground truth are from numerical solvers. Therefore the designed loss functions target the data-driven mismatch between predictions $\hat{y}(\mathbf{x}, t)$ and $y(\mathbf{x}, t)$, and more interestingly, the physical constraints for solving PDEs such as boundary conditions.

Mean Relative Error: We first quantify the data-driven mismatch using the mean relative error (MRE) between ℓ_2 norm:

$$\mathcal{L}_{MRE}(y, \hat{y}) = \frac{\|y - \hat{y}\|_2}{\|y\|_2}. \quad (1)$$

Spatial derivatives: Additional mismatch on first derivatives in the horizontal direction x and the vertical direction z are also included.

$$\mathcal{L}_{der}(y, \hat{y}) = \frac{\|\partial y / \partial x - \partial \hat{y} / \partial x\|_2}{\|\partial y / \partial x\|_2} + \frac{\|\partial y / \partial z - \partial \hat{y} / \partial z\|_2}{\|\partial y / \partial z\|_2}. \quad (2)$$

Spatial derivatives on the contaminant boundary: Maximum Contaminant Level (MCL) is the highest level of a contaminant that is allowed in drinking water recommended by the Environmental Protection Agency (EPA) [11]. Therefore, predicting the boundary of contaminant with higher concentration than the MCL is essential for site managers to protect water supply. We add first derivatives on the contaminant boundary $c \geq MCL$.

$$\mathcal{L}_{conc}(y, \hat{y}) = \frac{\|\partial c' / \partial x - \partial \hat{c}' / \partial x\|_2}{\|\partial c' / \partial x\|_2} + \frac{\|\partial c' / \partial z - \partial \hat{c}' / \partial z\|_2}{\|\partial c' / \partial z\|_2},$$

$$\text{where } c' = \begin{cases} 0, & c < MCL \\ 1, & c \geq MCL \end{cases}, \quad \hat{c}' = \begin{cases} 0, & \hat{c} < MCL \\ 1, & \hat{c} \geq MCL \end{cases} \quad (3)$$

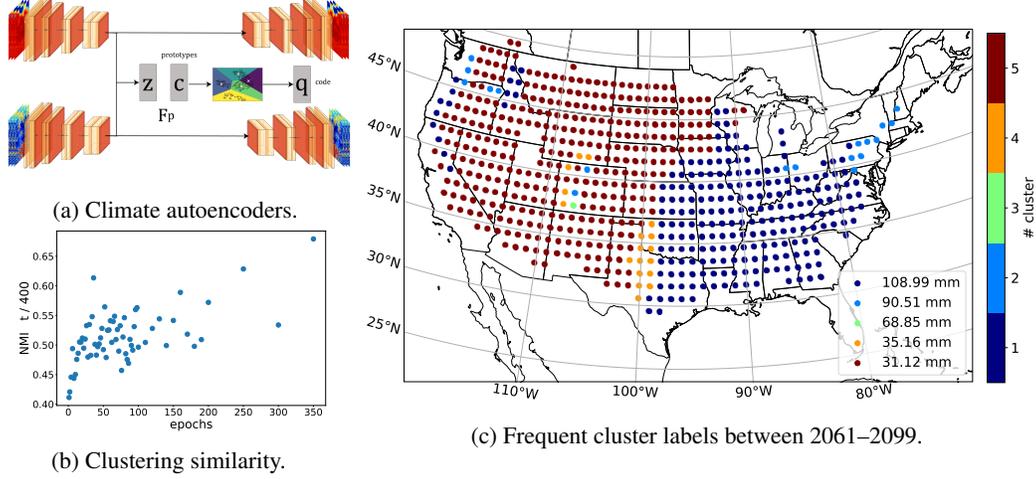


Figure 2: Architecture of climate autoencoders and evaluation of the results. (a) Features from two different autoencoders (i.e., precip. and ET.) are used for online clustering. (b) Similarity of cluster assignments along training. We calculate normalized mutual information (NMI) against labels from 400 epoch. Along training, clustering assignments evolves along training. (c) Spatial pattern of the most frequent cluster per grid between 2061–2099. Note that, we sort the clusters based on the mean precipitation within the cluster in descending order. Thus, small cluster numbers (e.g., #1) represent wet climate. While the frequent cluster patterns dominant by #1 and #5, monthly cluster patterns show spatiotemporal diversity.

Physics-informed boundary conditions: We add no flow boundary condition constraints in loss functions using physics-informed neural networks [14] to help solving the PDEs. The boundary of the spatial domain D is ∂D .

$$\mathcal{L}_{BC}(\hat{y}) = \|\hat{h}|_{\partial D}\|_2. \quad (4)$$

The final loss function combines all above loss functions.

$$\mathcal{L}(y, \hat{y}) = \mathcal{L}_{MRE}(y, \hat{y}) + \beta_1 \mathcal{L}_{der}(y, \hat{y}) + \beta_2 \mathcal{L}_{conc}(y, \hat{y}) + \beta_3 \mathcal{L}_{BC}(\hat{y}) \quad (5)$$

3 Unsupervised climate data clustering

Climate data clustering leverages convolutional autoencoders to reduce the dimensionality of continental-scale climate simulation outputs and performs a spatial-temporal cluster analysis, which help stakeholders to get a realistic recharge rate immediately in order to evaluate the groundwater flow anywhere across the continental US (CONUS). Note that median of precipitation and ET values from each resulting cluster will be used as inputs of the surrogate models (see in Section 2) in future work.

We use monthly precipitation and ET values from GFDL-ESM2G [6, 7] that participated CMIP5 over CONUS. We remove the 3-month-running mean and then standardize the monthly data so that our deseasonalizing approach removes the recurrent patterns but remains the climatological anomalies. We spatially subdivide each three months image into a 16 pixels \times 16 pixels scale, $\approx 2^\circ \times 2^\circ$ area, giving a smaller geographical and temporal unit, *patch*. We train our approach with 70% of only historical (1950–2020) patches (we leave 30% for inference) for 400 epochs on 4 K80 NVIDIA GPUs at a GCP instance to account for the future climate impact in inference.

While scalable clustering algorithms are widely available [2], an end-to-end autoencoder (see Figure 2a) and clustering training can further benefit the scalability. A joint loss function [1] formulates a combination of both reconstruction and clustering loss terms:

$$\mathcal{L}_{\text{joint}}(\theta) = \lambda_{\text{reconst}} \mathcal{L}_{\text{reconst}}(\theta) + \lambda_{\text{clustering}} \mathcal{L}_{\text{clustering}}(\theta), \quad (6)$$

Table 1: Mean relative errors (MRE) and mean squared errors (MSE) for all experiments.

Index	Architectures	Epochs	Loss ($\beta_1, \beta_2, \beta_3$)	MRE	MSE	Dataset
1	FNO-2D	30	(0,0,0)	0.051	2.71e-4	Validation
2	FNO-3D	30	(0,0,0)	0.055	2.98e-4	
3	U-FNO-2D	30	(0,0,0)	0.035	1.51e-4	
4	U-FNO-3D	30	(0,0,0)	0.037	1.29e-4	
5	U-FNO-3D	30	(0.1,0,0)	0.029	8.83e-5	
6	U-FNO-3D	30	(0,0.1,0)	0.033	1.10e-4	
7	U-FNO-3D	30	(0,0,0.1)	0.034	1.27e-4	
8	U-FNO-3D	30	(0.1,0.1,0.1)	0.028	8.14e-5	
9	U-FNO-2D	150	(0.1,0.1,0.1)	0.020	4.49e-5	Test
10	U-FNO-3D	150	(0.1,0.1,0.1)	0.014	2.44e-5	

where $\mathcal{L}_{\text{reconst}}$ corresponds to L_2 loss that quantifies the difference between a training data x and output of autoencoder $D_\theta(E_\theta(x))$; $\mathcal{L}_{\text{clustering}}$ term learns clustering assignments via online fashion; two coefficients λ_{reconst} and $\lambda_{\text{clustering}}$ balance the two terms to achieve smoother optimization and better representations. The learned representation via the end-to-end clustering approach reflects the association of group features of clusters by optimizing the joint loss function with Eq. 6.

Our $\mathcal{L}_{\text{clustering}}$ is motivated by an online method [4] with simplifying the cross entropy loss between two cluster assignment: “codes” q_c via the Sinkhorn-Knopp algorithm [5] and p_c that is computed as “prediction” obtained with a softmax of the dot product of K trainable prototype $\{c_1, \dots, c_K\}$ and the latent representation $z_c = E_\theta(x)$. We calculate the dot product with an output layer from a single layer perceptron F_p such that $z_c^\top c = F_p(E_\theta(x))$. Thus the second loss term in Eq. 6 is

$$\mathcal{L}_{\text{clustering}}(\theta) = - \sum_{k \in K} q_c^{(k)} \log(p_c) \quad \text{where} \quad p_c^{(k)} = \frac{\exp(\frac{1}{\tau} z_c^\top c_k)}{\sum_{k'} \exp(\frac{1}{\tau} z_c^\top c_{k'})}. \quad (7)$$

Figure 2c shows a spatial distribution of the most frequent cluster at each patch location for 2061–2099. Cluster numbers are sorted in descending order by the mean precipitation per patch. Our cluster’s climatological field is dominant by cluster #1 (i.e., the wettest cluster) and #5 (i.e., the driest climate cluster) over the CONUS. We also observe that #1 overlaps with humid subtropical climate zone (Cfa) and humid continental climate zone (Dfa) of a new and improved Koppen-Geiger climate classification [3] and #5 is spatially associated with their semi-arid (BS) and desert (BW) climate types, suggesting that our clusters capture physically meaningful spatial patterns.

4 Evaluation of different architectures and loss functions

We have in total 664 physical simulations from the Amanzi model with uncertain soil, subsurface and climate inputs. We split our dataset into training, validation, and testing (8:1:1) subsets. Table 1 (Index 1-4) shows the additional U-Net architecture gives us lower MRE and MSE on the validation dataset for both FNO-2D and FNO-3D architectures. In practice, U-FNO-3D trains around $4 \times$ faster (30 epochs, U-FNO-3D: 72 minutes, U-FNO-2D: 273 minutes). Therefore, we test our hybrid physics-based and data-driven loss functions on U-FNO-3D (Index 5-8 in Table 1). Every part of loss functions in Eq. 5, when $\beta_i \neq 0$, reduces the MRE and MSE error. We have the lowest validation error when adding all spatial derivatives and no flow boundaries ($\beta_1 = \beta_2 = \beta_3 = 0.1$, Index 9-10, in Table 1). U-FNO-3D with all three combined loss functions achieves the lowest MRE and MSE after training 150 epochs on a A-100 NVIDIA GPU at a GCP instance.

In summary, we have successfully developed the ML-based multi-scale digital twin. At the site scale, our ML-based surrogate model predicts spatiotemporal flow and contaminant transport rapidly, which saves computational times to solve PDEs, and thereby supports rapid decision-making for site managers. At the CONUS scale, our proposed unsupervised approach reduces the dimensionality of the vast historical and projected climate data to capture five unique climate patterns and provides lightweight climate projections for the site-scale model. We believe that more climate resilience analysis for other contamination sites can benefit from our method in this paper to develop the groundwater flow and contaminant transport surrogate model, accounting for the climate uncertainty.

Broad Impact

With tools for fast and reliable contaminant plume prediction under future climate scenarios, site managers and decision makers can evaluate the potential consequences and take rapid actions. We believe that more climate resilience analysis for other contamination sites can benefit from the method utilized in this paper to develop the groundwater flow and transport surrogate model. The use of clustering in the latent space of autoencoders on climate data for building representative climate regions can be extended to other applications in addition to using AI to create surrogate models.

Acknowledgments and Disclosure of Funding

This work was conducted at the Frontier Development Laboratory (FDL) USA 2022. The FDL USA is a public / private research partnership between NASA, the SETI Institute, Trillium Technologies Inc and private sector partners including Google Cloud, Intel, IBM, Lockheed Martin, and NVIDIA. These partners provide the data, expertise, training, and compute resources necessary for rapid experimentation and iteration in data-intensive areas. This material was based upon work supported by the Department of Energy [National Nuclear Security Administration] under Award Number DE-AI0000001. This work is also supported by the U.S. Department of Energy, Office of Environmental Management as a part of the Advanced Long-term Monitoring Systems (ALTEMIS) project under Award Number DE-AC02-05CH11231 to Lawrence Berkeley National Laboratory. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under contract no. DE-AC02-05CH11231. This research also used the Lawrence computational cluster resource provided by the IT Division at the Lawrence Berkeley National Laboratory (supported by the Director, Office of Science, Office of Basic Energy Sciences, of the U.S. Department of Energy under contract no. DE-AC02-05CH11231).

Disclaimer: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the US Government nor any agency thereof, nor any of their employees makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus product or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the US Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the US Government or any agency thereof.

References

- [1] R. Aparna and S. M. Idicula. Spatio-temporal data clustering using deep learning: A review. In *2022 IEEE International Conference on Evolving and Adaptive Intelligent Systems (EAIS)*, pages 1–10, 2022. doi: 10.1109/EAIS51927.2022.9787701.
- [2] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii. Scalable k-means++. *arXiv preprint arXiv:1203.6402*, 2012.
- [3] H. E. Beck, N. E. Zimmermann, T. R. McVicar, N. Vergopolan, A. Berg, and E. F. Wood. Present and future köppen-geiger climate classification maps at 1-km resolution. *Scientific data*, 5(1):1–12, 2018. doi: 10.1038/s41597-020-00616-w.
- [4] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33:9912–9924, 2020.
- [5] M. Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

- [6] J. P. Dunne, J. G. John, A. J. Adcroft, S. M. Griffies, R. W. Hallberg, E. Shevliakova, R. J. Stouffer, W. Cooke, K. A. Dunne, M. J. Harrison, et al. Gfdl’s esm2 global coupled climate–carbon earth system models. part i: Physical formulation and baseline simulation characteristics. *Journal of climate*, 25(19):6646–6665, 2012.
- [7] J. P. Dunne, J. G. John, E. Shevliakova, R. J. Stouffer, J. P. Krasting, S. L. Malyshev, P. Milly, L. T. Sentman, A. J. Adcroft, W. Cooke, et al. Gfdl’s esm2 global coupled climate–carbon earth system models. part ii: carbon system formulation and baseline simulation characteristics. *Journal of Climate*, 26(7):2247–2267, 2013.
- [8] R. Knutti and J. Sedláček. Robustness and uncertainties in the new cmip5 climate model projections. *Nature climate change*, 3(4):369–373, 2013.
- [9] A. Lavin, H. Zenil, B. Paige, D. C. Krakauer, J. E. Gottschlich, T. G. Mattson, A. Anandkumar, S. Choudry, K. Rocki, A. G. Baydin, C. Prunkl, O. Isayev, E. J. Peterson, P. L. McMahon, J. H. Macke, K. Cranmer, J. Zhang, H. M. Wainwright, A. Hanuka, M. M. Veloso, S. A. Assefa, S. Zheng, and A. Pfeffer. Simulation intelligence: Towards a new generation of scientific methods. *ArXiv*, abs/2112.03235, 2021.
- [10] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.
- [11] A. Libera, F. P. de Barros, B. Faybishenko, C. Eddy-Dilek, M. Denham, K. Lipnikov, D. Moulton, B. Maco, and H. Wainwright. Climate change impact on residual contaminants under sustainable remediation. *Journal of contaminant hydrology*, 226:103518, 2019.
- [12] L. Lu, P. Jin, and G. E. Karniadakis. Deeponet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *arXiv preprint arXiv:1910.03193*, 2019.
- [13] J. D. Moulton, S. Molins, J. N. Johnson, E. Coon, K. Lipnikov, M. Day, and E. Barker. Amanzi: An open-source multi-process simulator for environmental applications. In *AGU Fall Meeting Abstracts*, volume 2014, pages H51K–0758, 2014.
- [14] M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.
- [15] G. Wen, Z. Li, K. Azizzadenesheli, A. Anandkumar, and S. M. Benson. U-fno—an enhanced fourier neural operator-based deep-learning model for multiphase flow. *Advances in Water Resources*, 163:104180, 2022.
- [16] Z. Xu, R. Serata, H. Wainwright, M. Denham, S. Molins, H. Gonzalez-Raymat, K. Lipnikov, J. D. Moulton, and C. Eddy-Dilek. Reactive transport modeling for supporting climate resilience at groundwater contamination sites. *Hydrology and Earth System Sciences*, 26(3):755–773, 2022.

Check list:

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

1. For all authors...
 - (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
 - (b) Did you describe the limitations of your work? [Yes]
 - (c) Did you discuss any potential negative societal impacts of your work? [N/A]
 - (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
 - (a) Did you state the full set of assumptions of all theoretical results? [N/A]
 - (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
 - (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] Codes and simulation data are proprietary but we are planning to make them publicly available.
 - (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
 - (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
 - (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section 4
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
 - (a) If your work uses existing assets, did you cite the creators? [Yes]
 - (b) Did you mention the license of the assets? [Yes]
 - (c) Did you include any new assets either in the supplemental material or as a URL? [No]
 - (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
 - (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
 - (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
 - (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
 - (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]