STUDENT SUMMER INTERNSHIP TECHNICAL REPORT

Deep Learning Based Surrogate Modeling for Supporting Climate Resilience at Groundwater Contamination Sites

DOE-FIU SCIENCE & TECHNOLOGY
WORKFORCE DEVELOPMENT PROGRAM

Date submitted:

December 16, 2022

Principal Investigators:

Aurelien Meray (DOE Fellow Student)
Florida International University

Zexuan Xu, Ph.D. (Mentor)
Lawrence Berkeley National Laboratory

Haruko Wainwright, Ph.D. (Mentor)
Massachusetts Institute of Technology

Ravi Gudavalli, Ph.D. (Program Manager)
Florida International University

Leonel Lagos, Ph.D., PMP® (Program Director)
Florida International University

Submitted to:

U.S. Department of Energy
Office of Environmental Management
Under Cooperative Agreement # DE-EM0005213

FIU | Applied Research Center
FLORIDA INTERNATIONAL UNIVERSITY

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

During the summer of 2022, a DOE Fellow, Aurelien Meray, spent ten weeks in the Bay Area of California where he spent the first two weeks creating a documentation page for the PyLEnM python package at Lawrence Berkeley National Laboratory, and the remaining eight weeks participating in an Artificial Intelligence Bootcamp called the Frontier Development Laboratory (FDL). He was under the supervision and guidance of Research Scientists, Dr. Zexuan Xu and Dr. Haruko Wainwright. During the FDL research period, Aurelien worked with three other Ph.D. students from across the United States. This report will outline the work performed over the summer but mainly cover the individual contributions made by Aurelien.

Contamination of soil and groundwater is a major issue worldwide. It frequently takes decades to clean up contaminated sites and to keep track of natural attenuation. Due to extreme changes in climate i.e., evapotranspiration/precipitation, this can remobilize contaminants and spread them throughout affected groundwaters. Site managers and decision-makers can assess the potential effects and take prompt action using technologies for quick and accurate pollutant plume prediction under future climatic scenarios using machine learning techniques. The Fourier Neural Operator (FNO), a recent advancement in machine learning, has proven to be successful at learning partial differential equations (PDEs) and is suitable for use in this context.

In this work, two versions of the FNO enhanced with U-Net architectures were used to model multiple dimentions: UFNO-3D and UFNO-2D. With these networks, surrogate flow and transport models were created using physics simulations representing the Savannah River Site (SRS) F-Area.

1. INTRODUCTION

1.1. PyLEnM Python Package

The Python package, PyLEnM (Python for Long-term Environmental Monitoring), is a comprehensive machine learning (ML) framework for long-term groundwater contamination monitoring [1]. PyLEnM seeks to create a smooth data-to-ML pipeline with a number of useful features, including QA/QC, coincident/co-located data identification, automatic ingestion and processing of publicly accessible spatial data layers, and unique data summarization/visualization. During the first two weeks of the summer experience, Mr. Meray worked on creating a documentation page for the package.

1.2. Frontier Development Laboratory (FDL)

1.2.1. Program Description

Frontier Development Laboratory (FDL) uses artificial intelligence (AI) to advance scientific research and create new tools that can be used to address some of the most pressing problems facing humanity. These issues span a wide range, from the results of climate change to forecasting space weather. NASA, the Department of Energy (DOE), and the European Space Agency (ESA) are partners in the public-private FDL partnership. The program brings together some of the most brilliant minds in the fields of space science, artificial intelligence, and business with the support of DOE's Artificial Intelligence & Technology Office (AITO), NASA HQ, NASA ARC, NASA MSFC, the SETI Institute, and commercial AI partners Google Cloud, Nvidia, USGS, Luxembourg Space Agency, Pasteur Labs & Institute for Simulation Intelligence Intel, and Planet.

1.2.1. Research Project Introduction

Millions of people worldwide face a serious health danger from groundwater contamination. Industrial or mining operations, and nuclear waste storage grounds are just some of the locations that can present a range of health and environmental risks due to presence of harmful chemicals. Toxic substances can enter surrounding ground or surface waters, contaminating a source of drinking water for people, and they can also be ingested by plants and animals. Hazardous materials must therefore be properly managed at contaminated sites to stop them from harming people, animals, or natural systems. Due to global climate change, which has already generated visible consequences including temperature increase, sea level rise, increased glacier retreat, increased extreme weather events, and many more, additional challenges occur and increase hazards.

This research aims to speed up the assessment of climate change effects on groundwater polluted sites. Extreme weather conditions will alter when and how contaminants are released, and they most likely will have a significant effect on groundwater flow and contamination transport by raising the recharge rate (which is related to precipitation). It would eventually result in quicker contaminant transport and plume remobilization, increasing the risk to the local environment as the contaminating plume moved away from the contaminated location.

Using physical simulations of groundwater concentrations and contaminant movement like Amanzi is one approach to evaluate this issue [2]. Nevertheless, the influence of climate change on groundwater is still difficult to predict. We are dealing with multiscale challenges in addition

to numerous uncertainties in both physical simulation of groundwater flow and contaminant transfer and uncertainties in climate models. Models for predicting climate change are worldwide with an uneven resolution (10–100 km), whereas the issue of water contamination is local (1-10km).

Our main objective was to develop a surrogate model that can help in addressing the impact of climate resilience on groundwater flow and contamination transport on the Savannah River site, with the hope of eventually expanding the model to more general contaminated locations. An approximate technique that simulates the behavior of an expensive computation or process is known as a surrogate model [3]. The development of an unsupervised methodology to produce data-driven climate patterns without querying a large spatiotemporal climate dataset is a secondary but crucial goal. Climate resilience in this context is the capacity of contaminated places to return to their original conditions after being impacted by the effects of climate change. With the use of this tool, we hope to assist site managers in making appropriate and swift action.

2. RESEARCH DESCRIPTION

2.1. PyLEnM Documentation

The process of properly documenting the code was essential to ensuring that it could be updated and used by other scientists in the future. The Google docstring format was applied in order to be consistent with the documentation. This format consists of a description of the class or function, "Args," which contains the name, data type, and a brief description of each parameter, and "Returns," which summarizes the results of the program's termination. Figure 1 below displays two functions, "setData" and "getConstructionData," with the corresponding docstrings highlighted in yellow.

```
# SETTING DATA
def setData(self, data: pd.DataFrame, verbose: bool = True) -> None:
    """Saves the dataset into pylenm

    Args:
        data (pd.DataFrame): Dataset to be imported.
        verbose (bool, optional): Prints success message. Defaults to True.

    Returns:
        None:
    """
    validation = self.__isValid_Data(data)
    if(validation[0]):
        # Make all columns all caps
        cols_upper = [x.upper() for x in list(data.columns)]
        data.columns = cols_upper
        self.data = data
        if(verbose):
            print('Successfully imported the data!\n')
        self.__set_units()
    else:
        print('ERROR: {}'.format(validation[1]))
        return self.REQUIREMENTS_DATA()

def setConstructionData(self, construction_data: pd.DataFrame, verbose=True):
    """Imports the addtitional well information as a separate DataFrame.

    Args:
        construction_data (pd.DataFrame): Data with additonal details.
        verbose (bool, optional): Prints success message. Defaults to True.

    Returns:
        None:
    """
    validation = self.__isValid_Construction_Data(construction_data)
    if(validation[0]):
        # Make all columns all caps
        cols_upper = [x.upper() for x in list(construction_data.columns)]
        construction_data.columns = cols_upper
        self.construction_data = construction_data.set_index(['STATION_ID'])
        if(verbose):
            print('Successfully imported the construction data!\n')
    else:
        print('ERROR: {}'.format(validation[1]))
        return self.REQUIREMENTS_CONSTRUCTION_DATA()
```

Figure 1: Example functions with docstring format.

An advantage of having proper docstring documentation of code is the ability to automatically generate an HTML page with the documentation in a clear layout. This page was generated using a system called Read the Docs, an open-source software documentation hosting and versioning service [4]. This feature helps others find out how to use the functions that were written. Figure 2 shows the same two functions described previously in the autogenerated HTML page using Read the Docs.

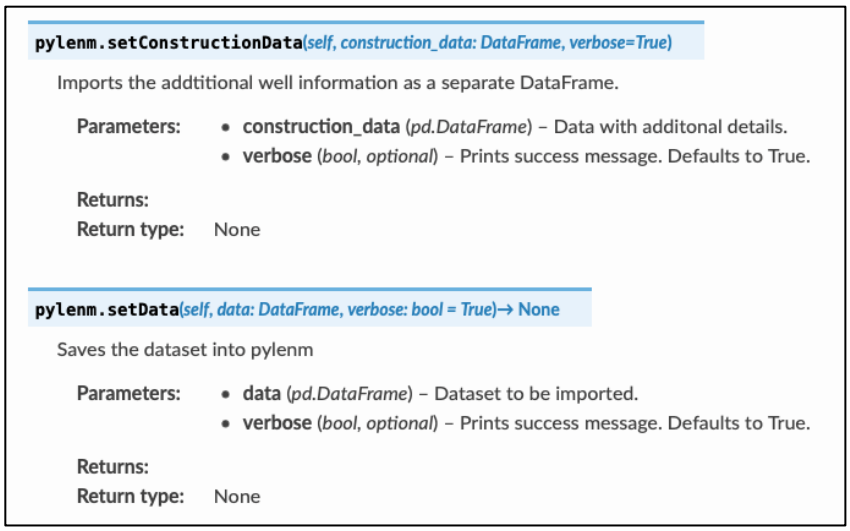

Figure 2: Read the Docs page showing the function documentation.

2.2. FDL Research Project

This section will highlight the information and procedures that were used to forecast specific flow and transport properties.

2.2.1. Simulation Data

Customized contamination flow and transport physics simulations were produced using Amanzi-ATS software to construct the surrogate model for this project. The simulations represent the groundwater contamination location at the Department of Energy's (DOE) Savannah River Site (SRS) F-Area. To provide variety to the dataset, a total of 1000 simulations were performed with input parameters that were randomly selected from a range. The 664 successful simulations out of the 1000 that were attempted were used for this research. There are 12 input values for each sample, including subsurface and climate variables like the upper layer's permeability and porosity as well as the measure of the pore-size distribution (represented as m), residual water content, recharge history, mid-century recharge, late-century recharge, seepage rate, and initial contaminant concentration (represented as seepage concentration). The minimum and maximum values from the 664 samples are displayed as *U* in Table 1. (a, b).

Table 1: Simulations of flow and transport are conducted using physical model parameters. U (a, b) represents the uniform distribution where a is the lowest value and b is the highest value. For various physical simulations, we choose random samples from those uniform distributions of model parameters.

Hydrostratigraphic unit	Upper aquifer	Tan clay	Lower aquifer
Porosity (-)	$U(3.1e{-}1, 4.7e{-}1)$	0.39	0.39
Permeability (m^2)	$U(2.5e{-}12, 7.5e{-}12)$	1.98e-14	5e-12
α (-)	$U(3.2e{-}4, 4.8e{-}4)$	5.1e-5	5.1e-5
Residual water content [Sr] (-)	$U(1.4e{-}1, 2.2e{-}1)$	0.39	0.41
m (-)	$U(4e{-}1, 6e{-}1)$	0.5	0.5
Seepage concentration (mol/L)	$U(1e{-}9, 1e{-}8)$	-	-
Seepage rate (kg-water m^{-2} s^{-1})	$U(1e{-}4, 2.5e{-}4)$	-	-
Cap rate (kg-water m^{-2} s^{-1})	$U(2e{-}9, 1e{-}8)$	-	-
Time-varying recharge (kg-water m^{-2} s^{-1})	$U(2e{-}6, 2e{-}5)$	-	-

A two-dimensional cross-section of the SRS F-Area in time is the result of the simulations. Despite the fact that the raw output files had several output variables, only four of them were used for this project. These include hydraulic head, tritium concentration, and Darcy velocity in both the horizontal and vertical directions (x and z). The simulation's outputs include yearly data from 1954 to 2100; however, for modelling reasons, we selected 5-year intervals beginning in 1955.

2.2.2. Transformations

Several modifications were used to make the input and output files useful for machine learning. The scalar input variables were broadcasted into the output size to guarantee that the inputs and outputs had the same dimensional shape. The same scalar values were projected to the output number of grids z and x, which was 24 by 257 respectively. Constant values were imputed into various strata of the cross-section thanks to extra domain knowledge of the subsurface at this specific site. The first five columns of Figure 3 input display this layering.

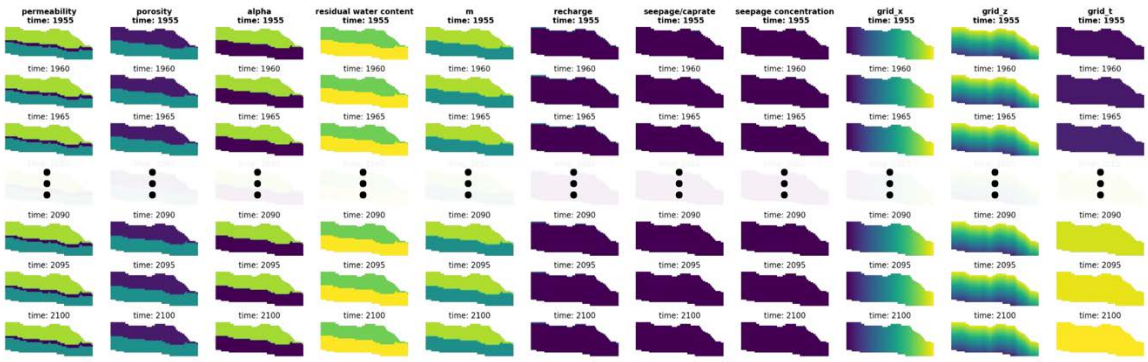

Figure 3: Input after applying the transformations.

The three values indicating the historical (1954-2020), mid-century (2020-2060), and late-century (2060-2100) were combined into one channel because recharge is a time-varying attribute. The seepage rate and cap rate underwent the same transition, with seepage only applying prior to 1988 and cap rate only applying following 1988. The grid's temporal dimension, together with its horizontal and vertical directions (designated as $grid\_x, grid\_z$, and $grid\_t$), were also recorded. After undergoing these changes, each sample's final dimensional shape can be expressed as $m(x,t) = (nz, nx, nt, nc)$, where nz denotes the size of z, nx denotes the size of x, nt denotes the number of time steps, and nc is the number of encoded parameters. Our input shape is $664 \times (24, 257, 30, 11)$ consequently.

2.2.3. Model architecture and approach

To simulate the flow and transport model, we tried two neural network designs U-FNO-3D and U-FNO-2D. Both are U-Net enhanced versions (U-FNO) of the Fourier Neural Operators (FNO). The FNO is the first ML-based technique to successfully predict turbulent flows with zero-shot super-resolution [5]. It accomplishes this by mapping input-output pairs between spaces with unlimited dimensions. Wen et al. introduced a U-Net architecture to each Fourier transform layer and observed better performance on their simulation data [6]. Despite losing the ability to train with data of varied resolutions, this additional U-Net convolutional mapping achieved a lower training/test error for multi-phase flow predictions. For these reasons, our team adapted two

versions of U-FNO as U-FNO-3D and U-FNO-2D where the latter is a recurrent version. Conceptual versions of the two architectures can be seen in Figure 4.

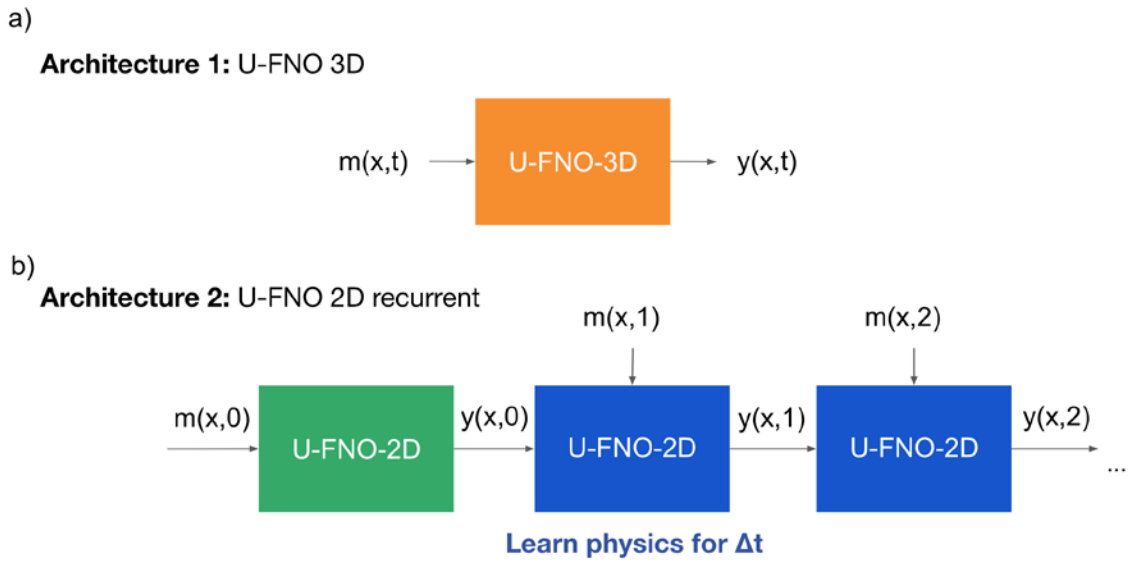

Figure 4: U-FNO 3D (a) and 2D (b) architectural models.

2.2.4 Loss Function

To optimize the performance of our model, four loss functions were established for testing. We included both data-driven factors and physical constraints in our loss functions. Loss functions are essentially the mis-fit between our prediction and the physical simulation. Below are simplified explanations of each of the losses:

- $\boldsymbol{L_0}$**:** *Mean Relative Error (MRE)*: mean relative error under L2 normalization.
- $\boldsymbol{L_1}$**:** *Spatial Derivatives*: first derivative if the x and z directions under L2 normalization.
- $\boldsymbol{L_2}$**:** *Spatial Derivatives on boundary*: same as L_1 but for contaminant values larger than the Maximum Contaminant Level (MCL).
- $\boldsymbol{L_3}$**:** *No flow boundary*: physical constraints for Darcy velocity and hydraulic head

The final loss function is the summation of the four losses above:

$$\boldsymbol{L = L_0 + \beta_1 L_1 + \beta_2 L_2 + \beta_3 L_3}$$

where the β's are the factors to weigh each loss differently.

3. RESULTS AND ANALYSIS

3.1. PyLEnM Documentation

The finalized version of the documentation page was published at https://pylenm.readthedocs.io/. Users can navigate through the webpage and discover how to install the package, how to import it in their python environment, and see a detailed description of each function available for use. Users can also see example Jupyter notebooks that go over most of the package's core functionality. The documentation homepage can be seen in Figure 3.

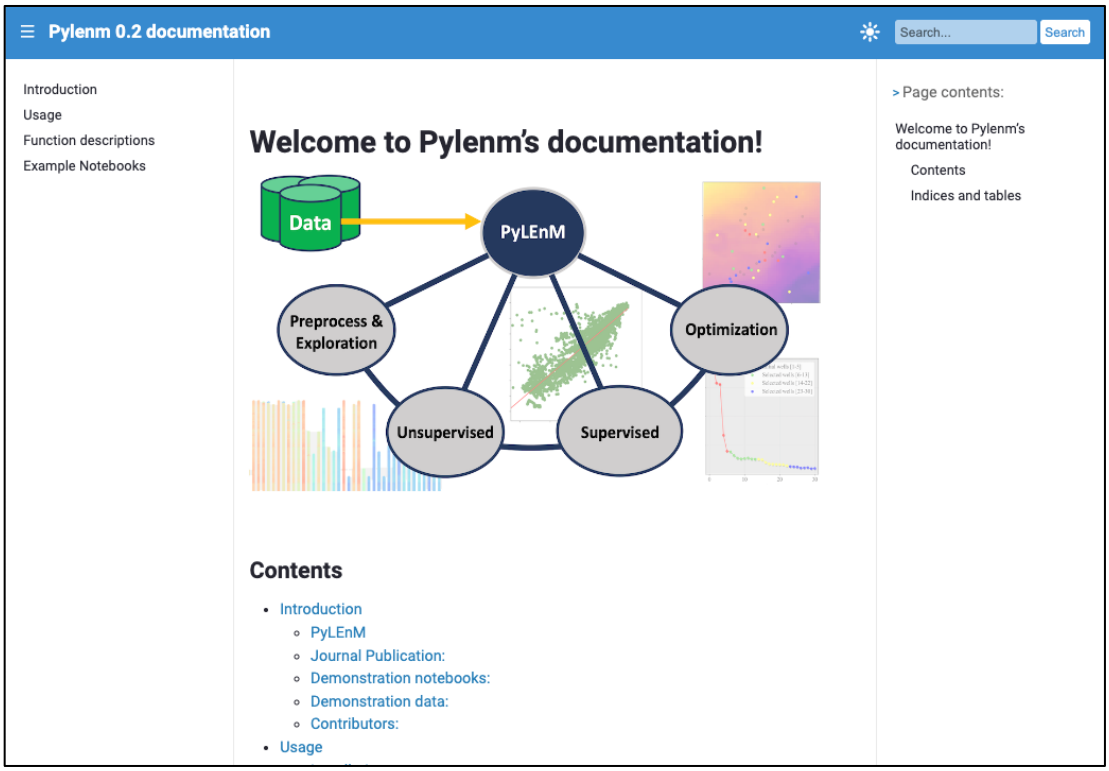

Figure 5: PyLEnM Documentation Home Page.

3.2. FDL Research Project

3.2.1. Architecture Evaluation

We checked to see if our predictions outperformed those of the original Fourier Neural Operator. As can be observed in Figure 6, the UFNO-2D and UFNO-3D versions of our architectures outperformed the non-U-Net version.

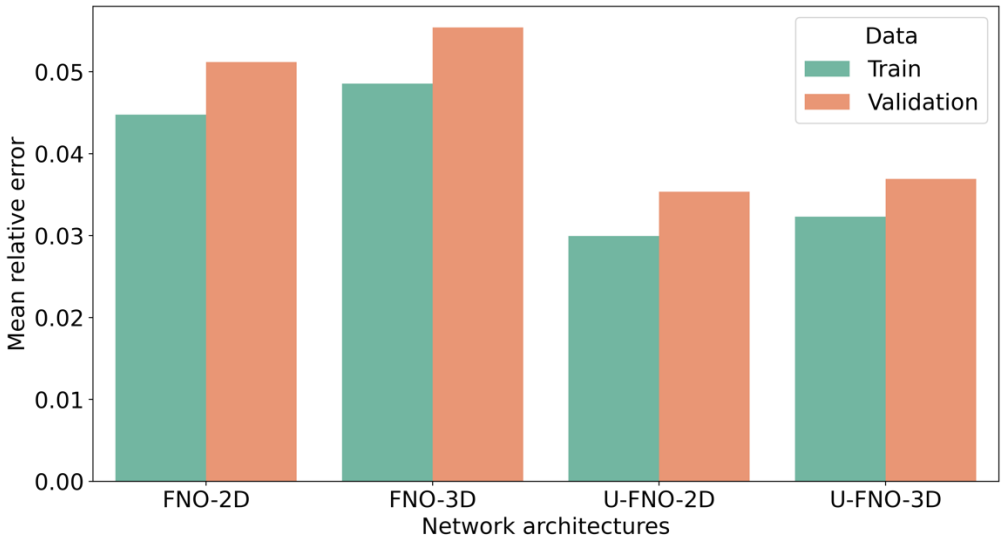

Figure 6: Evaluation of U-FNO and FNO.

3.2.2. Loss Function Evaluation

We performed a number of tests to identify the components of our loss function that had the greatest impact on reducing the overall error. To provide a baseline for comparison, we first set the betas to zero, which is equal to employing the MRE as the sole loss function. Then, in order to see the various contributions, we tried applying weights to each beta value independently. This demonstrated to us that the strongest influence on lowering the MRE was the $\boldsymbol{L_1}$ where we take the first derivative of the horizontal and vertical direction. However, this beat the single best loss function outcome when all three losses were taken into account. This is the loss function that we ultimately used, with each beta equal to 0.1. These results are shown in Figure 7.

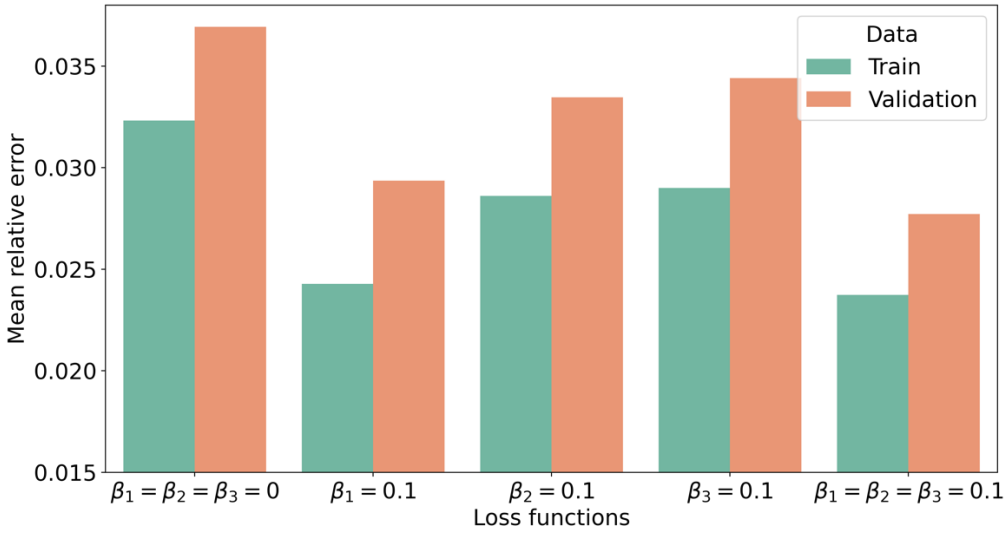

Figure 7: Comparing the different loss functions.

Table 2: All the experiments compared using the MRE and MSE

Index	Architectures	Epochs	Loss $(\beta_1, \beta_2, \beta_3)$	MRE (test for 9-10, val for the rest)	MSE (test for 9-10, val for the rest)
1	FNO-2D	30	(0,0,0)	0.051	2.71e-4
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
10	U-FNO-3D	150	(0,1,0,1,0.1)	**0.014**	**2.44e-5**

Once we determined that the combination of $\boldsymbol{\beta_1 = \beta_2 = \beta_3 = 0.1}$ produced the best results, we trained the U-FNO-2D and 3D for a total of 150 epochs to have the finals prediction models. The results for all of the experiments are shown in Table 2. With these trained models, we can now use them to predict on inputs the model has never been seen before but within the sample space provided in Table 1. Below in Figure 8 is an example of what the model predicts in comparison to the ground truth.

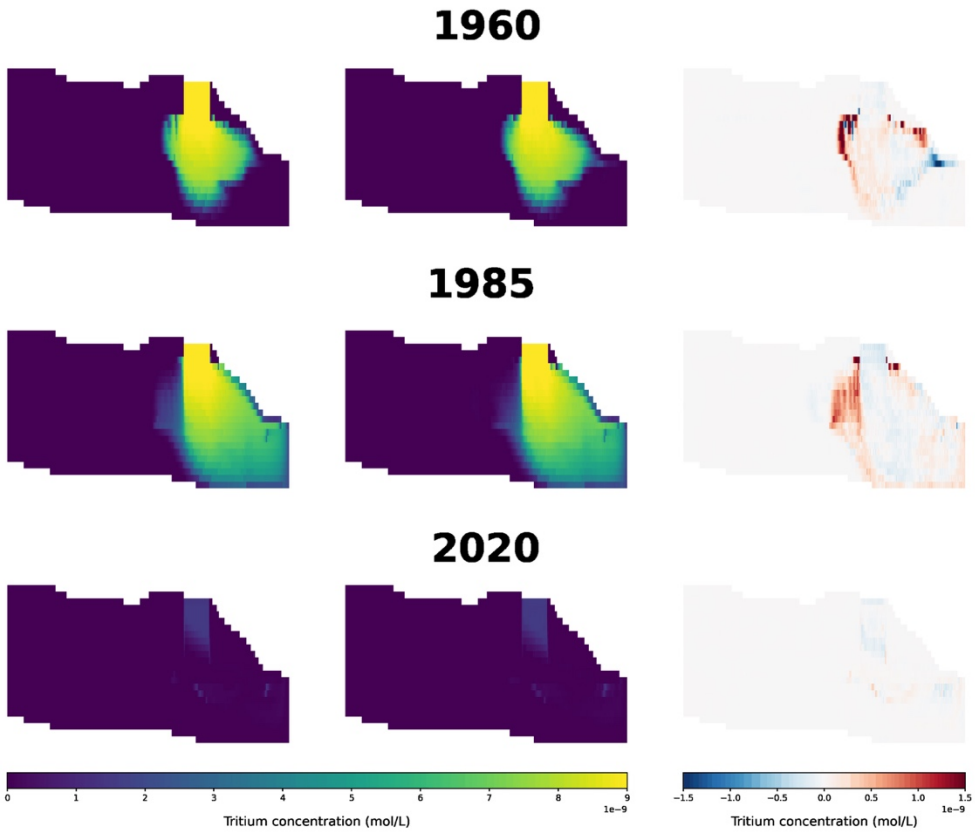

Figure 8: Tritium plume prediction on a sample from the test set using the U-FNO-3D. Left most column is the ground truth, middle column is the prediction, and right most column is the difference between the ground truth and the prediction.

4. CONCLUSION

The machine learning-based multi-scale digital twin has been successfully created. By using a surrogate model, we lessen the computational load of flow and transport calculations. For site managers, our physics-based surrogate model facilitates quick decision-making. With a wide variety of conceivable combinations of climatic and subsurface uncertainty, they may immediately evaluate spatial-temporal contaminant fluctuations. To provide safe drinking water in the face of climate change uncertainty, the movement of harmful substances in groundwater needs to be carefully studied. The availability of this technology for any site worldwide is our long-term objective for the team. More generalized, location-independent physical simulations are required, along with distinctive global climatic patterns, to achieve this.

Participating in the FDL program summer was a great experience. Aurelien has learned a lot more than he expected to and enjoyed his time meeting other researchers from different walks of life.

Figure 9: DOE Fellow, Aurelien (right), with his FDL team onsite the SETI Institute in Mountain View, California.

5. REFERENCES

[1] A. O. Meray *et al.*, "PyLEnM: A Machine Learning Framework for Long-Term Groundwater Contamination Monitoring Strategies," *Environ Sci Technol*, vol. 56, no. 9, 2022, doi: 10.1021/acs.est.1c07440.

[2] Z. Xu *et al.*, "Reactive transport modeling for supporting climate resilience at groundwater contamination sites," *Hydrol Earth Syst Sci*, vol. 26, no. 3, pp. 755–773, 2022.

[3] A. Lavin *et al.*, "Simulation intelligence: Towards a new generation of scientific methods," *arXiv preprint arXiv:2112.03235*, 2021.

[4] "Read the Docs." https://readthedocs.org/ (accessed Sep. 23, 2022).

[5] Z. Li *et al.*, "Fourier neural operator for parametric partial differential equations," *arXiv preprint arXiv:2010.08895*, 2020.

[6] 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," *Adv Water Resour*, vol. 163, p. 104180, 2022.

6. ACKNOWLEDGEMENTS

DOE Fellow Aurelien wishes to thank all members of the Digital Twin team for their collaboration and friendship. Sincerest gratitude towards Lijing Wang, Takuya Kurihana, Ilijana Mastilovic, Satyarth Praveen, Zexuan Xu, Haruko Wainwright, and Alexander Lavin.

This work has been enabled by the Frontier Development Lab (FDL.ai). FDL USA is a collaboration between several government agencies, Department of Energy (DOE), National Aeronautics and Space Administration (NASA), and U.S. Geological Survey (USGS), SETI Institute, and Trillium Technologies Inc., in partnership with private industry and academia. This public/private partnership ensures that the latest tools and techniques in Artificial Intelligence (AI) and Machine Learning (ML) are applied to basic research priorities in support of science and exploration of material concerns to humankind.