Imperial College London
Department of Earth Science and Engineering
MSc in Geo-energy with Machine Learning and Data Science

Independent Research Project
Final Report

Rapid modelling of ATES using Machine Learning

by
Hadrian Fung Nok Hei

Email: hadrian.fung23@imperial.ac.uk
GitHub username: gems-hf923
Repository: https://github.com/ese-msc-2022/irp-hf923

Supervisors:
Prof. Gege Wen
Prof. Matthew Jackson

August 2024

Table of Contents

Abstract

The project proposed a machine learning model based on Graph Neural Network to increase the efficiency of ATES simulation while retaining its accuracy from current numerical simulations. With a wide range of simulation results obtained from different parameters and aquifer configurations by in-house numerical model of IC-FERST, the study focused on the implementation of Graph Neural Network (GNN) on auto-regressive approach to predict the pressure and temperature from the output of numerical simulation on irregular spatial mesh. The proposed model is capable of capturing the physics behind and replicate the results from numerical simulations with a promising performance, while significantly accelerated the runtime from days to minutes. As a pioneering model in the field, the potential of machine learning models is unleashed to accelerate current numerical simulations and to provide instant insights on ATES installations, ultimately leading to a smoother transition to a low carbon future. The project not only enabled the rapid modelling of ATES, but also demonstrated the possibility of creating a digital twinning to replicate the complicated physics under various conditions, and the potential to accelerate numerical simulations in other fields with similar approaches in Graph Neural Networks.

1 Introduction

1.1 Background and Problem statement

ATES provides a low carbon technology solution to regional heating and cooling at the heart of energy transition. The design of ATES installation involves numerical simulations on subsurface fluid flow and heat transfer within the aquifer, which the numerical simulations are often computationally expensive with its fine spatial grid and closely spaced time steps to ensure accuracy.

To facilitate fast ATES simulation for providing instant insights on installation, the study proposed Machine Learning (ML) models to increase the efficiency from numerical simulations while retaining its accuracy. As the numerical simulation by in-house model of IC-FERST is implemented on adaptive mesh and irregular time steps, this work will evaluate the capability of proposed models to capture the physics of subsurface fluid flow and the thermodynamics under unstructured mesh (graph). ML Models including Graph Neural Networks (GNN) allow extra flexibility in deforming the mesh adaptively in each time steps, while conventional Convolutional Neural Networks (CNN) which takes in a regular mesh grid may require extra steps in pre-processing including casting an unstructured graph to a structured grid before training the model.

As GNN requires equivalent input and output shape for the mesh, this work proposed a framework to project auto-regressive pairs of graphs across 240 timesteps to the same mesh for prediction with Graph Neural Networks.

1.2 Review of existing work

As the project involves dealing with unstructured graph data for spatial-temporal modelling, review of existing work will focus mainly on the implementation of machine learning models including Graph Neural Networks (GNNs), Convolutional Neural Networks (CNNs) and their variations in other applications.

Previous work demonstrated a representation of unstructured mesh graph data by deploying GNNs to capture the time evolution of subsurface CO2 plume migration (Ju et al., 2023). GNNs capture the spatial-temporal relation in the subsurface even with complex geometry involving faults, in which CNNs in regular cartesian mesh grid may not be able to handle such heterogeneity accordingly (Ju et al., 2023). The outstanding generalisation capability leveraged by GNN can potentially be incorporated into different model architecture to enhance efficiency but retaining certain accuracy.

1.3 Novelty of work

With all the existing work successfully model the subsurface multi-phase fluid flow, the project developed a machine learning approach that effectively models subsurface multi-phase fluid flow by capturing spatial-temporal features from unstructured graph data. The proposed method not only accelerates ATES simulations but also serves as a transferable surrogate model for complex systems. Comparing with conventional simulations like IC-FERST, which may take days to process 240 timesteps, our approach delivers accurate predictions within minutes using a pre-trained model. Additionally, while IC-FERST's reliance on CPU computation limits hardware parallelization, our ML-based model inherently supports GPU parallelization, significantly reducing runtime without sacrificing accuracy.

1.4 Proposed model and solution

The proposed model involves a U-net architecture with Graph Convolutional layers in both encoder and decoder, connected by skip connections. This model architecture will serve as a working solution for auto-regressive approach that takes in the graph of previous timestep and predict the next timestep on the same mesh, as U-net architecture provides a solid grounding for spatial graph reconstruction with skip connections. Graph Convolution layers allow the model to aggregate the information from nearby nodes and perform message passing in a relatively computational efficient way. As a result, the proposed U-GCN will allow the model to capture the physics behind while retaining spatial features, addressing the problems in our use case.

2 Methodology

2.1 Data Acquisition

As the model will be trained on simulation input parameters, mesh, and simulation results, 840 sets of numerical simulations from in-house code IC-FERST are prepared as training dataset. The figure below outlines the data involved in each simulation scenario:

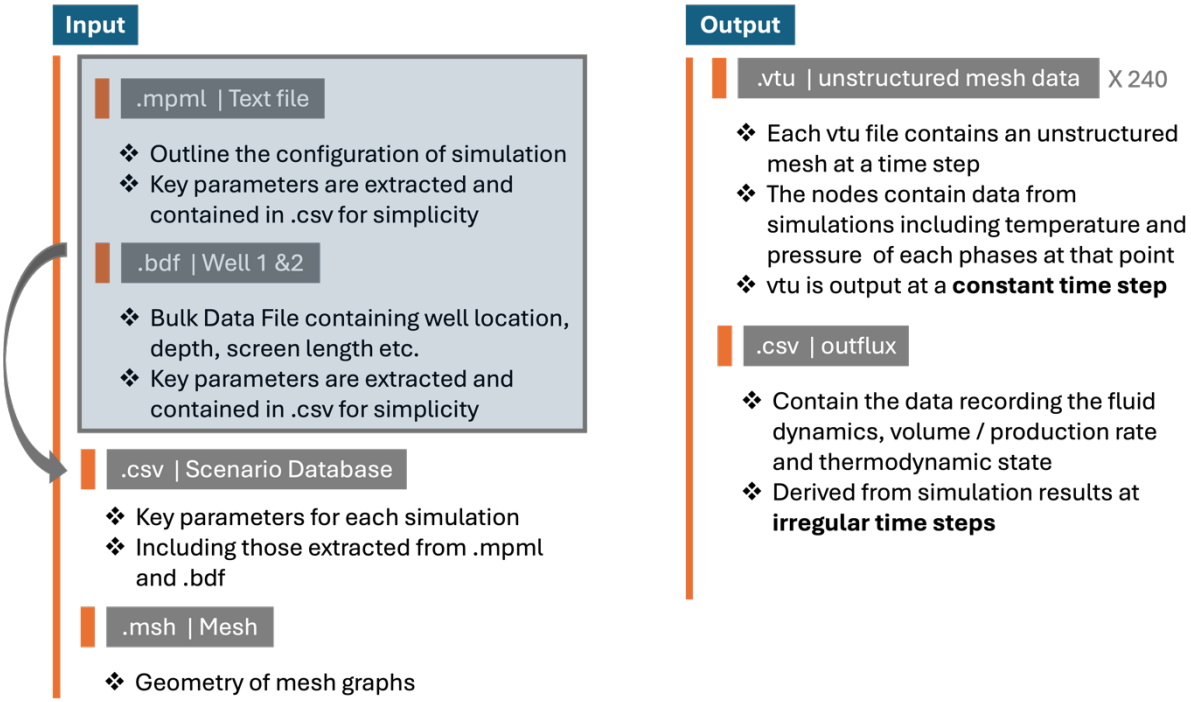

Figure 1 | Input and output of the model from simulation data

The figure outlines the required data from IC-FERST simulation for each simulation, which the model will be trained on (1) parameters from scenario database, and (2) mesh to predict 240 timesteps of simulation results in autoregressive pairs.

2.2 Approaches and workflow

This work proposed an auto-regressive approach for Machine Learning (ML) based ATES simulation. The pre-trained ML model takes in the graph containing the state variables of physical quantities including temperature, pressure, vertical permeability of different layers of current timestep and predict the graph of the next timestep on the same mesh. The predicted timestep subsequently serves as the input for the next iteration, enabling the generation of a sequence of predicted graphs across any user-defined number of timesteps.

2.3 Data Pre-processing

2.3.1 Graph creation

To prepare the dataset for training the Graph Neural Network (GNN), each VTU file, spanning timesteps 1 to 240 across different scenarios, is processed into graph data. Each VTU file contains a mesh (comprising nodes and edges), tetrahedron cell attributes, and node attributes. The mesh is combined with the geological properties of the reservoir and well configurations, which are extracted from a scenario database (CSV), to create the graph object. The thermodynamic states (Phase 1 Temperature and Pressure) along with other node attributes are then assigned to the nodes of the graph accordingly. The injection phase corresponding to each timestep is extracted from the predefined injection strategy sequence and assigned as a node attribute in the graph. After assigning the node attributes, the Cartesian coordinates of the nodes are extracted to calculate the Euclidean distance for the edges within the graph's topology.
Tetrahedron cell attributes (controlled volume) can be incorporated into the graph data by projecting them to all 6 corresponding bi-directional edges accordingly and store as edge attributes in the graph object. It allows the model to capture the fluid dynamics throughout the message passing in Graph Neural Network (GNN).

2.3.2 Mesh projection for auto-regressive pairs

After creating graphs that represent the state of each timestep, consecutive graphs are grouped in auto-regressive pairs such that the graph of former timestep serves as the input of model, and the latter one acts as the prediction target (ground truth). The two graphs are then projected to the same mesh for training Graph Neural Networks, either by backward projection (projecting the target graph to the mesh of input graph) or forward projection (projecting the input graph to the mesh of target graph) with nearest neighbour interpolation.

2.3.3 Standardization and train-test split

To evaluate the capability of generalization of model, the dataset is split into train and test set in which the test set constitutes approximately 10% of the full dataset. Both train and test set are standardized respectively, whereas the test set is standardized with the mean and standard deviation of train set to avoid data leakage throughout the training process, aligning with standard ML practice.

2.4 Model Architecture

2.4.1 U-GNN: Graph Neural Network with classical U-net architecture

The U-GNN model presented in this work adopts the classical U-Net style architecture that is designed to handle spatial reconstruction and perform temporal prediction in an auto-regressive approach tailored for rapid ATES simulations. The architecture leverages the strengths of Graph Convolutional Networks (GCNs), multi-hop aggregation layers, and Graph Attention Networks (GATs) to effectively capture the complex physics of plume diffusion and fluid dynamics in a porous medium.

The architecture of U-GNN consists of an encoder block, processor, and decoder block, with skip connections linking each layer of the encoder and decoder blocks to facilitate the spatial reconstruction of the ATES aquifer. The encoder and decoder blocks each comprise three layers of graph convolution, using either GCN Convolution, GAT Convolution, multi-hop aggregation GCN Convolution, or a hybrid approach. The encoder effectively extracts and aggregates the spatial features of the input graph, generating an embedding that is passed through the subsequent processor and decoder blocks.

The processor block may involve temporal attention mechanisms or temporal convolutional blocks, allowing the model to learn the temporal evolution from one timestep to the next. The processed graph embeddings in the latent space of the processor are then passed to the decoder block, where they are combined with skip connections from the encoder block to reconstruct the graph's spatial features.

2.4.1.1 Graph Convolution Neural Network (GCN)

The Graph Convolutional Network (GCN) is the fundamental building block of the U-GNN architecture. GCNs generalize the concept of convolution to graphs, allowing the model to aggregate information from a node's local neighbourhood. Specifically, in GCNs, each node updates its feature representation by aggregating features from its immediate neighbours, weighted by the graph's adjacency matrix. This message-passing mechanism is computationally efficient and captures local spatial dependencies in the data, making it well-suited for tasks involving spatial relationships, such as modelling the diffusion of temperature and pressure plumes in ATES simulations.

In the context of the U-GCN model, the GCN layers are used to encode the input graph data by progressively capturing higher-order dependencies through multiple layers stacked in the encoder. The encoder path comprises three GCN layers, each capturing progressively broader spatial contexts. This structure ensures that the model effectively learns the spatial correlations and interactions within the ATES reservoir, which is crucial for accurately capturing the physics behind.
The strength of GCNs lies in their ability to efficiently model graph-structured data by aggregating local information in a relatively computationally efficient way. Compared to standard CNNs, which are limited to grid-like structures, GCNs can naturally handle irregular graph topologies, making them intrinsically ideal for the mesh-based data used in ATES simulations. By capturing the spatial dependencies in the reservoir, GCNs contribute significantly to the model's ability to perform accurate spatial reconstruction in an auto-regressive framework.

2.4.1.2 Multi-hop aggregation Graph Convolution layers

The multi-hop aggregation GCN layers enhance the standard GCN by allowing the model to aggregate information over multiple hops, thereby expanding the receptive field of each node. By replacing the standard GCN layer with the Multi-Hop GCN convolution layer, the message-passing process is iterated over multiple hops (denoted by K) within the single layer in forward pass. Each hop represents an additional step in the propagation of information across the graph, enabling the model to capture more distant dependencies that are critical in simulating complex physical processes. Specifically for ATES simulation, the diffusion of temperature and pressure in the reservoir is influenced not just by immediate neighbours but also by nodes that are several steps away in the graph. By aggregating information over multiple hops, the model can more accurately simulate these processes, leading to improved spatial reconstruction and predictive performance.

The strength of multi-hop aggregation lies in its ability to model complex, multi-scale interactions within the graph, which are essential for capturing the dynamics of fluid in ATES simulations. This makes it a powerful extension of the standard GCN, particularly in scenarios where capturing long-range spatial dependencies is crucial including the timesteps involving injection phase change.

2.4.1.3 Graph Attention (GAT) Layers

The Graph Attention Network (GAT) introduces attention mechanisms to graph convolutions, allowing the model to weigh the relative importance of neighbouring nodes differently. Unlike GCNs, which treat all neighbours equally, GATs learn to assign attention scores to different edges, effectively focusing more on the most relevant nodes. This ability to dynamically prioritize and focus on information from different parts of the graph is particularly advantageous in complex environments like ATES simulations, where certain areas of the reservoir including the well screen or high gradient areas may have a more significant impact on the overall system dynamics.

The GAT layers are particularly effective in scenarios where the importance of neighbouring nodes is not uniform, especially when simulating the influence of localized high-permeability zones or injection wells in ATES reservoirs. By focusing on the most relevant nodes, GATs may be more advantageous by incorporating attention mechanisms in tasks involving complex, heterogeneous interactions.

2.4.1.4 Hybrid Layers (GAT | GCN)

The combination of GCN and GAT layers in the encoder and decoder balances computational efficiency with model expressiveness, wherein GCN layers capture general spatial patterns of immediate neighbouring nodes with greater computational efficiency, and GAT layers are applied after GCN layers to further refine these patterns by emphasizing the most critical interactions.

The rationale for deploying a hybrid architecture in the encoder and decoder lies in its expressiveness to capture local refinement with a larger receptive field while retaining computational efficiency. By using GCN layers where broad aggregation is sufficient, followed by GAT layers where fine-tuned attention is needed, the model achieves a promising performance without unnecessary computational overhead. This approach makes the model an effective solution for the complex, multi-scale nature of ATES simulations. By accurately modelling both local and global interactions, it addresses the intricate dynamics inherent to these systems.

2.4.2 Temporal Attention Mechanism

The Temporal Attention Mechanism is incorporated within the U-GCN architecture to enhance the model's ability to focus on the most influential temporal features during the spatial-temporal modelling process. This mechanism assigns varying degrees of importance to different timesteps, allowing the model to concentrate on those that are most relevant, particularly during changes in the injection strategy. By applying a weighted aggregation of temporal features, the attention mechanism effectively distils critical temporal information, improving the model's accuracy in predicting how these changes influence the reservoir's thermodynamic states, and leading to more precise spatial-temporal reconstructions.

2.4.3 Temporal Convolutional Neural Network

The Temporal Convolutional Neural Network (TCN) in the U-GCN architecture is designed to efficiently capture both short-term and long-term temporal dependencies across sequential data. By using dilated convolutions, the TCN processes the temporal evolution of physical states in the reservoir, particularly accommodating the impact of varying injection strategies over time. This approach allows the model to consider a broader temporal context while maintaining computational efficiency, ensuring accurate predictions of the reservoir's behaviour, and enhancing the overall effectiveness of the spatial-temporal modelling process.

3 Results

3.1 Model Performance

R-squared score

	Train dataset		Test dataset	
	Temperature	Pressure	Temperature	Pressure
U-GCN (forward)	88.14%	92.54%	85.67%	91.52%
U-GCN (backward)	93.93%	93.44%	91.65%	90.58%
Hybrid U-GAT (forward)	89.94%	92.58%	86.87%	91.66%
Hybrid U-GAT (backward)	93.93%	91.20%	92.20&	90.08%
Multi-hop GCN (K = 2 | backward)	93.88%	92.10%	91.21%	89.92%
Multi-hop GCN (K = 3 | backward)	93.80%	90.81%	91.98%	89.66%

Table 1 | R2 score of different model architecture for train and test set in Phase 1 pressure and temperature respectively.

The R2 score of different model architecture shows that all models can reach a high accuracy, where Hybrid architecture that replace the last encoder and decoder layer to GAT is the most consistent model with the highest generalization power. Besides, the increase in hop may not necessarily boost the performance in a monotonic way given the complexity of data. Also, the R2 score of backward projection is higher than forward projection, but it does not necessary mean that backward projection performs better, as backward projection may miss out the new nodes at the high gradient area and lead to a higher R2 score.

3.2 Model Visualizations (by Projection schemes)

3.2.1 Forward Projection

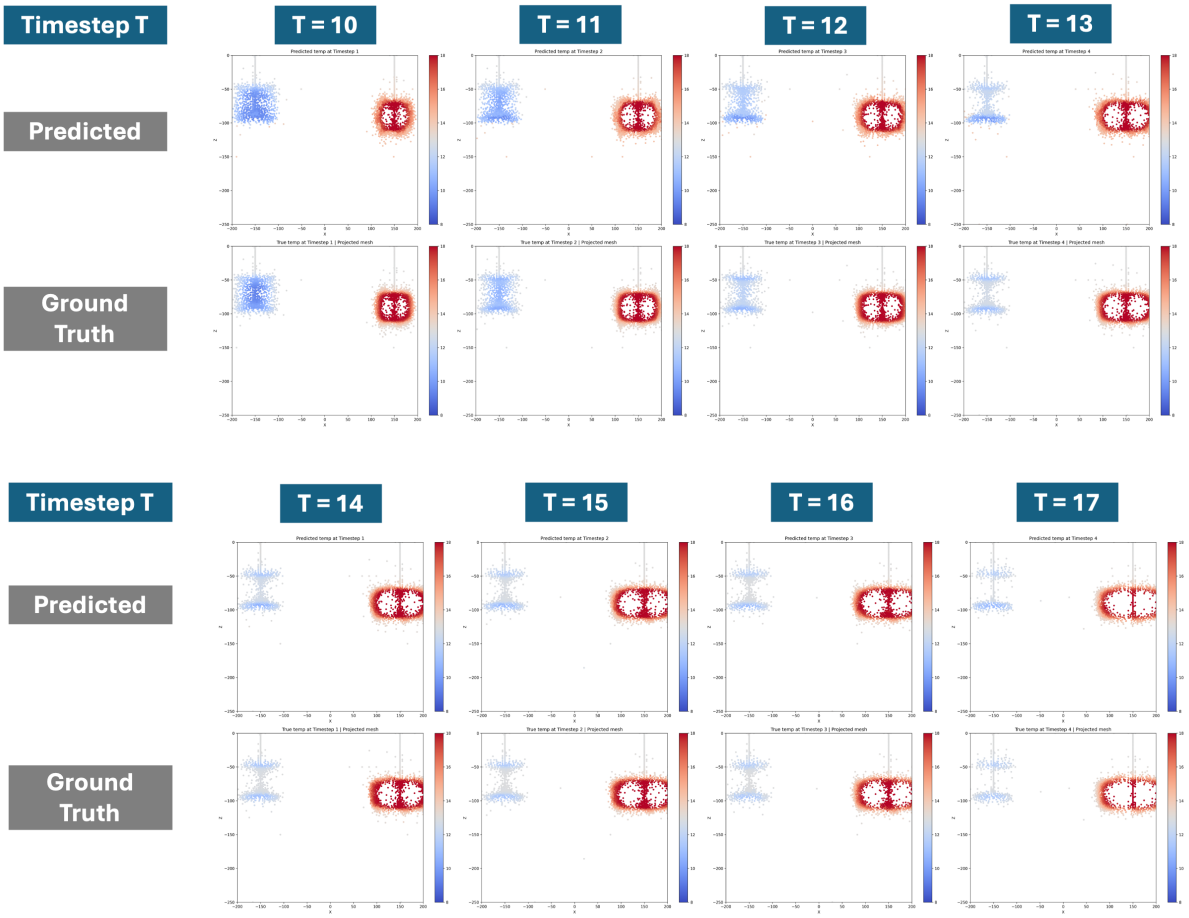

Figure 2 | Visualization of model predictions from timestep 10 (top left) – 17 (bottom right). For each timestep, upper row shows the model prediction and bottom row shows the ground truth. It shows a series of plume diffusion with injection at warm well.

3.2.2 Backward Projection

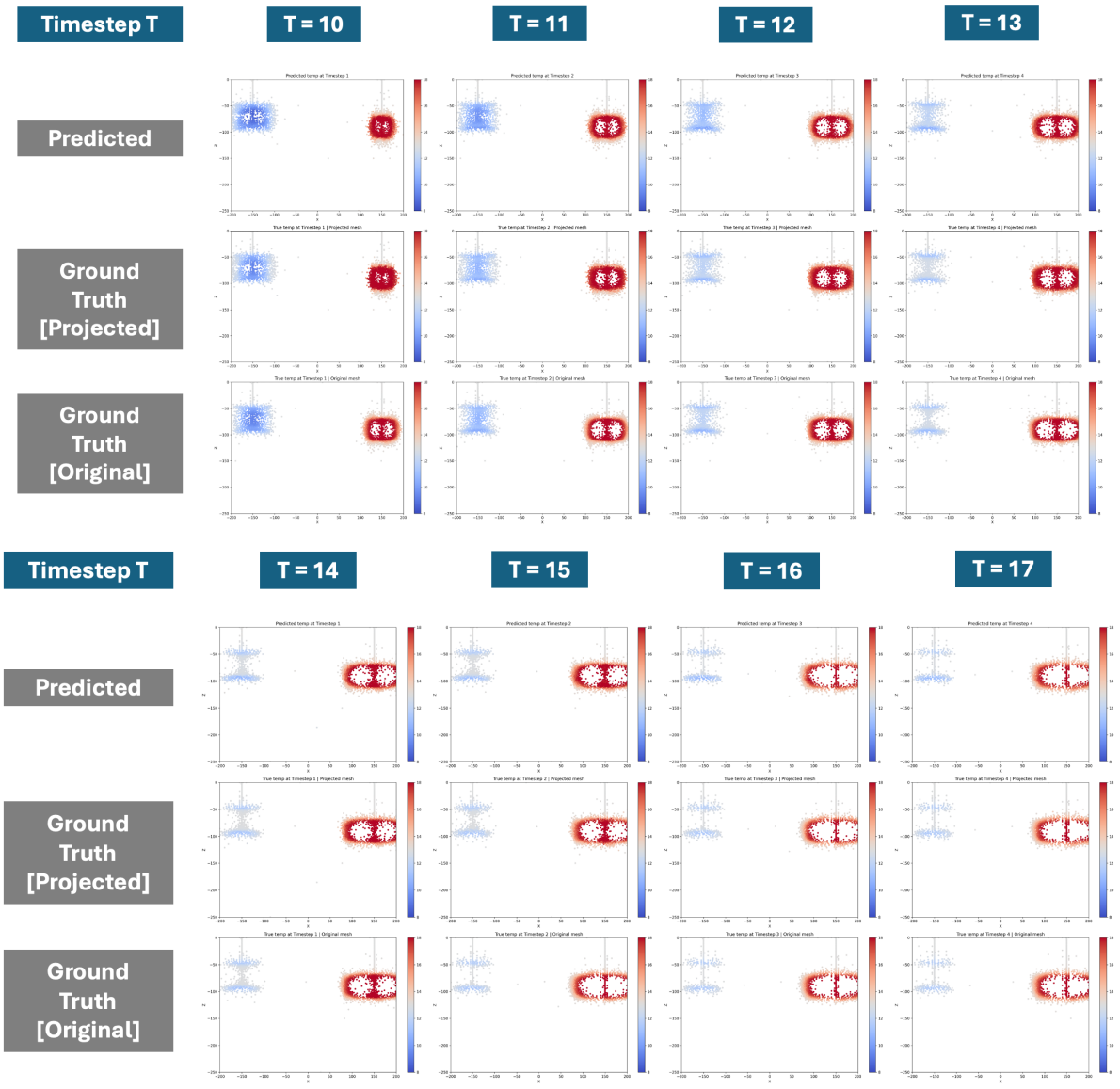

Figure 3 | Visualization of model in backward projection predictions from timestep 10 (top left) – 17 (bottom right). For each timestep, upper row shows the model prediction, middle row shows the ground truth on projected mesh and bottom row shows the ground truth on original mesh.

4 Discussion

4.1 Interpretation of results on fluid flow physics in porous medium

4.1.1 Heat Plume diffusion and Geo-mechanical pressure build-up distribution

The model demonstrates significant accuracy in predicting phase 1 temperature and pressure, effectively capturing the complex dynamics of heat plume diffusion and subsurface pressure, in alignment with fluid flow physics in porous media.
Given the non-linear dynamics of heat plume diffusion and geo-mechanical pressure build-up in a coupled system, the model successfully extracts spatial features from graph inputs, learning the temporal evolution of temperature and pressure, particularly at the fluid flow front and in high-gradient areas.
During the steady injection timesteps at the well screen, the predicted phase 1 temperature accurately reflects the heat plume expansion, demonstrating the model's precision in capturing both the temperature and geometry of the plume. Notably, the model excels in capturing abrupt temperature gradient changes at the heat plume's fringe. The phase 1 pressure variations, confined around the well screen with constant wellbore injection, exhibit exceptional predictability under steady state injection, emphasizing the model's reliability in these scenarios.

4.1.2 Injection phase change

While the proposed model effectively predicts the spatial and temporal evolution of temperature and pressure during steady-state injection—capturing the dynamics of heat plume diffusion—its performance diminishes at transition timesteps, particularly during abrupt injection starts in subsequent timesteps. The injection strategy employed in the dataset alternates between 8 timesteps of injection in the cold well, followed by 4 timesteps of rest, and then 8 timesteps of injection in the warm well, followed by another 4 timesteps of rest. The model struggles most notably during the transition from rest to injection, where the dynamics deviate markedly from other timesteps.

At transition timesteps, the model is expected to capture the diffusion of residual heat plume from previous injections and to form a new local injection plume near the well screen. Although the model accurately captures residual heat plume diffusion at the background fringe, it fails to respond effectively to the injection strategy as indicated by the input nodes at the well screen.

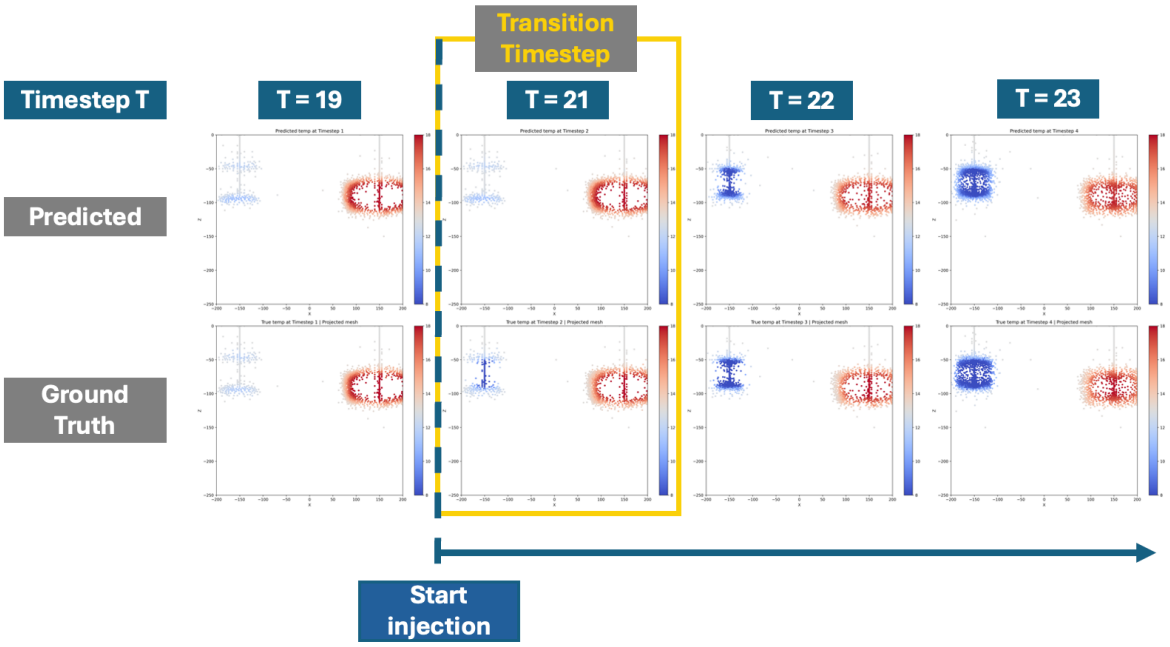

Figure 4 | Evolution of phase 1 temperature from timestep 19 to 23, in which injection in the cold well starts from timestep 21. The model prediction on the top row shows that it fails to capture the injection information at the core of the well screen.

This limitation can be attributed to both machine learning and physics perspectives. From a machine learning standpoint, the model is purely data-driven without any pre-informed physics. As such, its predictions are heavily dependent on the dataset. In this case, only 19 transition timesteps—out of a total of 240—represent non-steady-state injection scenarios. This imbalance biases the model towards steady-state conditions, impeding its ability to generalize to the transition from non-injection to injection. Additionally, the structural difference between pre- and post-injection states may be too significant for the model to reconcile under its current architecture, limiting its ability to capture both plume diffusion and the abrupt onset of injection.

From a physics perspective, the partial differential equations governing steady-state and transient-state solutions often involve different boundary conditions and solution derivations. As a result, there is no singular solution that can effectively address both steady-state and transient scenarios. While the model excels in generalizing heat plume diffusion and pressure build-up under steady-state conditions, it may not be universally applicable to transition phases that require a transient-state solution. Addressing this limitation might necessitate the development of a new model or architecture that can effectively capture the unique dynamics of these transition phases.

4.2 Significance of work and transferability

The proposed model delivered a promising prediction for a rapid ATES simulation, in which the significance of the work does not only stay within the field of geophysical fluid simulations and providing insight on a transferable approach for replicating complicated system by graph neural network, but also creating a broader societal impact to enable a smoother transition to a low carbon future at the heart of energy transition.

4.2.1 Significant acceleration on numerical simulations

for IC-FERST without Scalability for hardware acceleration (High GPU suitability by neural network, instead of algorithms)

The proposed model can successfully replicate the physics and deliver accurate prediction with a significantly faster runtime comparing with conventional numerical simulations. To generate a full 240 timesteps of simulations on ICFERST, it often takes up to days to complete the entire run, while the proposed model can produce an instant insight of simulations within minutes on a pre-trained model. Conventional numerical simulations may not be able to benefit from GPU hardware acceleration, in which the computationally expensive operations can only rely on CPU threads. However, ML based approaches for simulations intrinsically enable parallelization on GPU cores, in which heavily mundane work that requires huge computational power can be distributed to multiple cores to accelerate the computation significantly, yielding a high scalability for the proposed model.

4.2.2 Transferable approach for accelerating systems with coupled physics

The work provided an insight of a transferrable approach to accelerate complicated systems that may be computational expensive to simulate in an algorithmic approach. The full workflow adopted in the work from attaining data, pre-processing by projection, to the variants of auto-regressive graph neural networks is not confined only to the field of geophysical fluid flow dynamics, but may also be adopted to aerodynamics, meteorology and climate modelling, traffic system simulation in town planning and more potential applications in different fields. These systems are often entangled with sets of coupled partial differential equations which requires sophisticated resolution and accuracy. Graph neural network can be a computational efficient alternative to conventional numerical simulations, which allows variable resolution and topology to enable extra flexibility and expressiveness in replicating the model predictions. Intrinsic hardware acceleration by GPU parallelized computing in ML-based approach may also be beneficial in fields like meteorology or weather forecast, which requires an accurate spatial temporal simulation at fine grids. As numerical modelling in weather forecast requires lead time ahead of predicted weather events, GNNs are proven to be the next-generation scalable solutions for variable resolution considering its potential in accelerating numerical simulations, such that GNN-based auto-regressive models including GraphCast from Google Deepmind are already adopted in actual meteorological application in Global Forecast System (GFS) (Lam et al., 2023). The proven success of the approach in the work is expected to be transferrable to more applications in different fields.

4.2.3 Energy transition: Rapid modelling of ATES for instant insight

The proposed model offers a rapid prediction capability for the simulation of Aquifer Thermal Energy Storage (ATES) systems, significantly enhancing the potential deployment of geothermal solutions across a broader range of locations. Given that ATES systems require carefully tailored configurations to optimize the energy harnessed from seasonal groundwater storage, rapid modelling can provide immediate insights into various design options, thereby facilitating the adaptation of ATES configurations to diverse local conditions. As an innovative approach in the realm of next-generation energy solutions, the instant insights provided by this model could play a pivotal role in accelerating the energy transition towards a low-carbon future.

4.3 Further advancement for current constraints:

Reinforcement Learning for mesh adaptivity (+ Transferred Learning)

On top of the proposed GNN, further advancement on adaptive mesh by Reinforcement Learning can be a possible way to develop onwards. As the proposed model delivers predictions on the same mesh topology, the introduction of adaptive mesh refinement may unleash the full potential of the model with optimized adapted mesh in a full ML-based approach.

4.3.1 Current constraints: constant mesh required by GNN

One of the constraints encountered by the current model is that the U-GNN structure intrinsically requires the same input and output size, which the proposed workflow solve by projecting both input and predicted timestep on the same mesh topology to address the constraints. Constant mesh is often limited in lacking the optimal nodes to represent the predictions, and we may need much more nodes to ensure the resolution is fine enough to represent the entire model space.

4.3.2 Possible solution: Reinforcement Learning for adaptive mesh refinement

If adaptive mesh refinement (AMR) can be adopted combining the U-GNN architecture as a two-stage model, it will serve as a powerful solution to the above problem such that the model could refine the resolution at the area of high gradient or fine-grained changes, or coarsening the mesh at the area where there are no significant changes to increase computational effectiveness. We suggest that AMR can be implemented by introducing rewards, policy and specifically designed loss in Reinforcement Learning (RL) Agent, such that the model will optimize every move of refining, coarsening, displacing nodes, or moving edge to reach the balance between computational efficiency and accuracy to represent the whole reservoir.

4.3.3 Current approach for mesh refinement on ML

There are different state-of-the-art approaches to deal with adaptive mesh currently, including supervised learning and reinforcement learning for mesh adaptivity. Both approaches demonstrated extraordinary performance in delivering the optimal mesh topology for predictions in graphs.

4.3.3.1 Supervised Learning with fixed number of nodes and edges

Supervised learning allows the model to learn the ground truth mesh from existing adaptive meshing scheme, such that in the use case of IC-FERST, the ML model can learn the adaptive meshing directly from the ground truth from vtu files. However, supervised learning approaches are currently confined to constant size on input and output, limiting its ability to add or remove nodes from mesh.

The Universal Mesh Movement Network (UM2N) is one of the state-of-the-art models described in the paper "Towards Universal Mesh Movement Networks," representing a supervised learning approach to adaptive mesh refinement (AMR). UM2N uses a Graph Transformer-based encoder and a Graph Attention Network (GAT)-based decoder to facilitate mesh movement (Zhang et al., 2024). The network is trained to predict the optimal relocation of mesh nodes to improve the numerical solution of partial differential equations (PDEs). This model excels in adapting the mesh to various PDEs without requiring re-training, ensuring that computational resources are focused on regions of high importance or dynamics. The key innovation from the work is the use of element volume loss instead of traditional coordinate loss, which helps the model maintain mesh integrity by reducing the likelihood of inverted elements (Zhang et al., 2024). Once trained, UM2N can be applied in a zero-shot manner across different problems, making it a versatile tool in scenarios where high computational efficiency and accuracy are necessary.

4.3.3.2 Reinforcement Learning for mesh refinement including Lagrangian dynamics on world and mesh space

Reinforcement learning is a computationally efficient yet promising choice in modelling mesh refinement. MeshGraphNets , as detailed in the paper "Learning Mesh-Based Simulation with Graph Networks," present a reinforcement learning approach to adaptive mesh refinement that can include dynamics in Lagrangian systems (Wu et al., 2023). This method operates by encoding the simulation state into a graph and performing message passing in both mesh-space (capturing internal dynamics in Lagrangian reference frame) and world-space (handling external dynamics). The reinforcement learning framework allows MeshGraphNets to dynamically adjust the mesh resolution during the simulation, allocating computational resources to critical regions where higher accuracy is needed (Wu et al., 2023). This adaptivity is particularly useful in simulations involving complex physical systems like fluid dynamics and structural mechanics. The approach leverages the flexibility of graph neural networks (GNNs) to handle irregular meshes, allowing for the refinement of meshes in response to evolving simulation conditions. This method is highly efficient, running significantly faster than traditional simulation methods, and provides a scalable solution to high-dimensional physical simulations.

4.3.4 Reinforcement Learning on adaptive meshing with variable number of nodes and edges on optimal topology

Building on the strengths of state-of-the-art models that integrate machine learning with Graph Neural Networks (GNNs), the application of Reinforcement Learning (RL) to GNNs presents a promising avenue for further performance enhancement while optimizing computational resources. This approach has the potential to incorporate adaptive mesh refinement strategies within the IC-FERST, including the addition of new nodes and edges, into the RL framework. By designing reward mechanisms that balance computational efficiency with accuracy, RL can be effectively combined with the existing auto-regressive GNN framework. This integration could yield more precise results by dynamically optimizing the mesh topology, thereby reducing computational overhead while improving expressiveness for predictions.

5 Conclusion

The proposed U-GNN model has demonstrated significant potential in accelerating ATES simulations, providing accurate predictions of subsurface temperature and pressure evolution with drastically reduced computation times. The model's capability to capture complex spatial-temporal dynamics, especially during steady-state operations, showcases its practical utility in real-world applications. Furthermore, the integration of adaptive mesh refinement techniques through Reinforcement Learning suggests a promising path forward to enhance the model's performance under varying conditions. This work not only advances the efficiency of ATES simulations but also sets the groundwork for broader applications in fields requiring rapid and accurate numerical simulations.

Reference

Chu, A.K., Benson, S.M. and Wen, G. (2022) 'Deep-learning-based flow prediction for CO2 storage in shale–sandstone formations', *Energies*, 16(1), p. 246. doi:10.3390/en16010246.

Ju, X. *et al.* (2023) *Learning CO2 plume migration in faulted reservoirs with Graph Neural Networks* [Preprint].

Lam, R. *et al.* (2023) 'Learning skillful medium-range global weather forecasting', *Science*, 382(6677), pp. 1416–1421. doi:10.1126/science.adi2336.

Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., and Battaglia, P.W. (2021) 'Learning mesh-based simulation with graph networks', *International Conference on Learning Representations (ICLR).* Available at: *https://sites.google.com/view/meshgraphnets*.

Wen, G. *et al.* (2022) 'U-fno—an enhanced fourier neural operator-based deep-learning model for multiphase flow', *Advances in Water Resources*, 163, p. 104180. doi:10.1016/j.advwatres.2022.104180.

Wen, G. *et al.* (2023) 'Real-time high-resolution CO2 geological storage prediction using nested Fourier neural operators', *Energy & Environmental Science*, 16(4), pp. 1732–1741. doi:10.1039/d2ee04204e.

Wen, G., Hay, C. and Benson, S. (2021a) *CCSNet: A deep learning modeling suite for CO2 storage* [Preprint]. doi:10.26226/morressier.612f6737bc981037241008ce.

Wen, G., Hay, C. and Benson, S. (2021b) *CCSNet: A deep learning modeling suite for CO2 storage* [Preprint]. doi:10.26226/morressier.612f6737bc981037241008ce.

Wu, T., Maruyama, T., Zhao, Q., Wetzstein, G., and Leskovec, J. (2023) 'Learning controllable adaptive simulation for multi-resolution physics', *International Conference on Learning Representations (ICLR).* Available at: *https://openreview.net/forum?id=PbfgkZ2HdbE*

Zhang, M., Wang, C., Kramer, S., Wallwork, J.G., Li, S., Liu, J., Chen, X., and Piggott, M.D. (2024) 'Towards Universal Mesh Movement Networks' [Preprint].