Water Resources Research®

RESEARCH ARTICLE

10.1029/2023WR035278

Contaminant Transport Modeling and Source Attribution With Attention-Based Graph Neural Network

Min Pang[1,2,3], **Erhu Du**[2,3], **and Chunmiao Zheng**[3,4,5]

[1]College of Hydrology and Water Resources, Hohai University, Nanjing, China, [2]The National Key Laboratory of Water Disaster Prevention, Hohai University, Nanjing, China, [3]Yangtze Institute for Conservation and Development, Hohai University, Nanjing, China, [4]Eastern Institute for Advanced Study, Eastern Institute of Technology, Ningbo, China, [5]Guangdong Provincial Key Laboratory of Soil and Groundwater Pollution Control, School of Environmental Science and Engineering, Southern University of Science and Technology, Shenzhen, China

Key Points:

- A novel graph-based deep learning method is proposed for modeling contaminant transport constrained by monitoring data
- The proposed model quantifies the contribution of each potential contaminant source to the observed concentration at an arbitrary location
- The deep learning method substantially reduces the computational cost compared with a physics-based contaminant transport model

Correspondence to:

E. Du and C. Zheng,
erhudu@hhu.edu.cn;
zhengcm@sustech.edu.cn

Citation:

Pang, M., Du, E., & Zheng, C. (2024). Contaminant transport modeling and source attribution with attention-based graph neural network. *Water Resources Research*, *60*, e2023WR035278. https://doi.org/10.1029/2023WR035278

Received 6 MAY 2023
Accepted 10 MAY 2024

Author Contributions:

Conceptualization: Min Pang, Erhu Du, Chunmiao Zheng
Data curation: Min Pang
Formal analysis: Min Pang, Erhu Du
Funding acquisition: Min Pang, Chunmiao Zheng
Methodology: Min Pang, Erhu Du
Software: Min Pang
Supervision: Erhu Du, Chunmiao Zheng
Writing – original draft: Min Pang
Writing – review & editing: Min Pang, Erhu Du, Chunmiao Zheng

Abstract Groundwater contamination induced by anthropogenic activities has long been a global issue. Characterizing and modeling contaminant transport processes is crucial to groundwater protection and management. However, challenges still exist in process complexity, data constraint, and computational cost. In the era of big data, the growth of machine learning has led to new opportunities in studying contaminant transport in groundwater systems. In this work, we introduce a new attention-based graph neural network (aGNN) for modeling contaminant transport with limited monitoring data and quantifying causal connections between contaminant sources (drivers) and their spreading (outcomes). In five synthetic case studies that involve varying monitoring networks in heterogeneous aquifers, aGNN is shown to outperform LSTM-based (long-short term memory) and CNN- based (convolutional neural network) methods in multistep predictions (i.e., transductive learning). It also demonstrates a high level of applicability in inferring observations for unmonitored sites (i.e., inductive learning). Furthermore, an explanatory analysis based on aGNN quantifies the influence of each contaminant source, which has been validated by a physics-based model with consistent outcomes with an R^2 value exceeding 92%. The major advantage of aGNN is that it not only has a high level of predictive power in multiple scenario evaluations but also substantially reduces computational cost. Overall, this study shows that aGNN is efficient and robust for highly nonlinear spatiotemporal learning in subsurface contaminant transport, and provides a promising tool for groundwater management involving contaminant source attribution.

Plain Language Summary Groundwater contamination caused by human activities is a longstanding global challenge. Accurately characterizing and modeling the movement of contaminants is crucial for the protection and management of groundwater resources. However, the complexity of the processes, limitations in data availability, and high computational demands pose significant challenges. In the age of big data, machine learning offers new avenues for exploring contaminant transport in groundwater. In this study, we introduce a novel machine learning model called an attention-based graph neural network (aGNN) designed to model contaminant transport with sparse monitoring data and to analyze the causal relationships between contaminant sources and observed concentrations at specific locations. We conducted five synthetic case studies across diverse aquifer systems with varying monitoring setups, where aGNN demonstrated superior performance over models based on other approaches. It also proved highly capable of making inferences about pollution levels at unmonitored sites. Moreover, an explanatory analysis using aGNN effectively quantified the impact of each contaminant source, with results validated by a physics-based model. Overall, this study establishes aGNN as an efficient and robust method for complex spatiotemporal learning in subsurface contaminant transport, making it a valuable tool for groundwater management and contaminant source identification.

1. Introduction

Groundwater, the hidden water resource that globally accounts for nearly 50% of drinking water, 40% of irrigation water, and over 30% of industrial water supply, is vital to human welfare (United Nations WATER, 2018). However, anthropogenic contamination has emerged as a significant concern that threatens groundwater resource sustainability. Groundwater contamination issues are further compounded by the "tragedy of the commons" where stakeholders prioritizing their individual interests contribute to cumulative pollution challenges in shared aquifers (Andrews & Hennet, 2022; Gorelick & Zheng, 2015). Groundwater quality management requires a clear understanding of the “source-pathway-receptor” connection in the system (Soriano et al., 2021; Zheng & Bennett, 2002), which can be interpreted as the causal connections of contaminant releases to the impact of pollution at locations of concern through hydrogeological pathways. Since direct measurements are not sufficient for quantifying the effects in “source-pathway-receptor,” groundwater models play an important role in elucidating the coupled human and natural dynamics to inform contamination management (Lall et al., 2020).

Researchers, engineers, and practitioners often use physics-based groundwater contamination models to better understand and address groundwater contaminant problems. However, several challenges usually arise in the development and application of physics-based models (Harbaugh, 2005; Markstrom et al., 2005; Refsgaard et al., 2010; Zheng & Wang, 1999). First, developing physics-based models requires thorough knowledge of the physical, biological, and chemical processes involved in contaminant transport and a precise formulation of these processes by using a set of complex mathematical equations (Tong et al., 2022; Zheng & Bennett, 2002). Second, building, calibrating, and validating physics-based models rely on various data sources, including site hydrogeology, aquifer geometry, well observations, and initial and boundary conditions, which may not be readily available in the study area (Pang et al., 2022; Pietrzak, 2021). Third, large-scale physics-based models typically require a lengthy simulation time for scenario analysis, which makes the study computationally expensive and time-consuming (Gorelick & Zheng, 2015; Pang & Shoemaker, 2023; Xia & Shoemaker, 2021).

Recent breakthroughs in data science and information technology have prompted the application of machine learning (ML) in many research areas (Reichstein et al., 2019; Shen, 2018). ML models, often considered surrogates or supplements to physics-based models of water resources (Huang et al., 2021), can intelligently integrate multi-source data. Thus, the advantage of ML lies in its ability to automatically learn from data, enable models to learn patterns and make predictions without explicit rule-based programming, as required in physics-based models. Additionally, well-trained ML can quantitatively describe system dynamics with accuracy at a computational cost much lower than that of physics-based models (Sun & Scanlon, 2019). Thus, while physics-based models may have prohibitive computational demands, ML models exhibit high efficiency with substantially reduced computational requirements. Furthermore, ML's adaptability to diverse data and tasks underscores its greater flexibility, and the transferability of ML allows the reuse of learned knowledge across related domains, enhancing its overall versatility.

Despite a wide range of applications of ML in water resources (Babakhani et al., 2017; Jing et al., 2023; Mugunthan et al., 2005; Najah Ahmed et al., 2019; Razavi et al., 2012; Sajedi-Hosseini et al., 2018; Yu et al., 2023), the conventional ML methods are not particularly suitable for learning spatiotemporal patterns (Gentine et al., 2018; Yu et al., 2021). Deep learning (DL) methods have been developed that are highly effective at modeling spatial or temporal patterns. Long-short term memory (LSTM) and convolutional neural networks (CNN) are two widely used DL methods (Hakim et al., 2022; He et al., 2021; Kratzert et al., 2019; Mo et al., 2019; Pang et al., 2023; Wunsch et al., 2021). LSTM is capable of learning temporal sequential data (Hochreiter & Schmidhuber, 1997); however, it cannot directly take spatial information into consideration (Zhong et al., 2019). CNN can employ convolutional operators to extract spatial information from image-like data (LeCun et al., 1989), but it assumes spatial regularity and locality; thus, it cannot capture complex spatial connectivity when the data are irregular in space (Babaeian et al., 2022; Mohammed & Corzo, 2024; Ni et al., 2020; Sun et al., 2021).

Graph neural networks (GNNs) have emerged as a way to generalize learning into non-Euclidean space (Bronstein et al., 2017). This approach makes it possible to learn complex graph relationships in spatially irregular data. Despite monitoring networks/systems providing limited observations, GNN-based methods are highly effective at capturing dependencies in graphs (Fan et al., 2019; Li et al., 2021; Wu et al., 2021, 2022), and have demonstrated efficacy in groundwater modeling (Alzahrani et al., 2023; Bai & Tahmasebi, 2023; Feng et al., 2023). GNNs can be adapted for modeling solute transport in the subsurface by using nodes and edges that are analogous to contamination monitoring and solute migration. However, there are several specific challenges: (a) the groundwater flow pathway and the regimes of preferential flow and transport are hidden underground; therefore, they are ambiguous to determine; (b) the contaminant transport process involves various mechanisms, such as advection and dispersion, that can produce high nonlinearity in spatiotemporal patterns; and (c) contaminant propagation relies on collections of data about various human activities and natural responses, which are complex to learn.

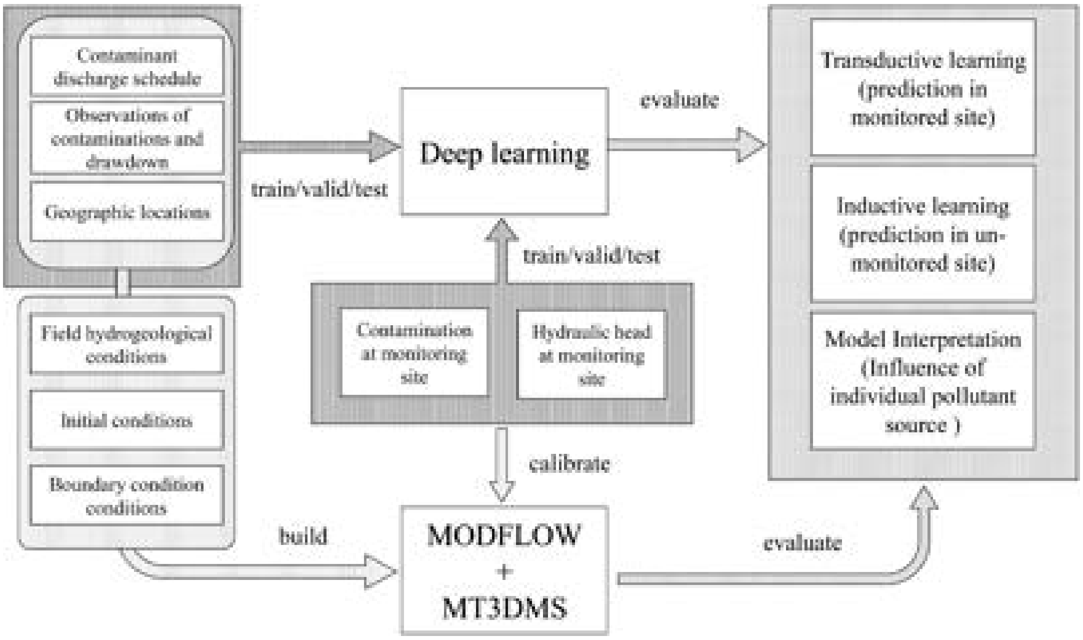
Contaminant discharge schedule
Observations of contaminations and drawdown
Geographic locations
Field hydrogeological conditions
Initial conditions
Boundary condition conditions
train/valid/test
Deep learning
evaluate
train/valid/test
Contamination at monitoring site
Hydraulic head at monitoring site
calibrate
build
MODFLOW
+
MT3DMS
evaluate
Transductive learning (prediction in monitored site)
Inductive learning (prediction in un-monitored site)
Model Interpretation (Influence of individual pollutant source)

Figure 1. The overview of workflow and data for contaminant transport modeling by using two approaches: deep learning and physics-based model (MODFLOW and MT3DMS). The models are assessed in three tasks: transductive learning, inductive learning and model interpretation.

To address these challenges, we propose aGNN, a novel attention-based graph neural modeling framework that combines (a) a graph convolutional network (GCN), (b) an attention mechanism and (c) embedding layers to simulate contaminant transport processes in groundwater systems. GCN extracts graph information by message passing through nodes and edges to effectively learn spatial pattern (Kipf & Welling, 2016). The attention mechanism is the key component in transformer network that is adept in sequential analysis (Vaswani et al., 2017). Embedding layers are latent-space learning mechanisms that represent high dimensionality in spatiotemporal processes (Battaglia et al., 2018). Studies of traffic and pedestrian trajectories have shown that attention-based graph neural networks exhibit competitive performance in the task of single-process spatio-temporal prediction (Guo et al., 2021; Zhou et al., 2021). In this study, we extend its application to learning multiple processes in groundwater flow and solute transport problems. In addition, new coordinate embedding method is employed for inductive learning at unmonitored contamination locations that have yet to be studied.

The objectives of this study are threefold. First, we investigate the performance of aGNN in multi-process involved contaminant transport modeling. GNN-based, CNN-based, LSTM-based methods are adapted for the same end-to-end learning task of multi-step ahead spatial prediction to gain insights into how well each model performs. Second, we evaluate the ability of aGNN to transfer the knowledge learned from monitoring data to the unmonitored site via inductive learning depending on the availability of data and the heterogeneity of the aquifer. Third, we employ an explainable AI technique, namely Shapley value, which originated from the concept of cooperative game theory. It calculates the contribution of each attributor to predictions, and in this study, Shapley value represents the contaminant source attribution in the case of multi-source discharge. We also assess the time efficiency of using aGNN compared to the use of a physics-based model. From three different aspects, we demonstrate that the attention-based graph model is a prospective tool for contamination modeling and that can inform policy makers in groundwater contamination management.

2. Methodology

2.1. Overview

Figure 1 shows the workflow of two approaches (i.e., DL methods and physics-based models) to model the spatiotemporal variability in groundwater quality in response to pollutant discharge from multiple sources. DL models (including our proposed aGNN, as well as CNN-based and RNN-based models) do not use detailed

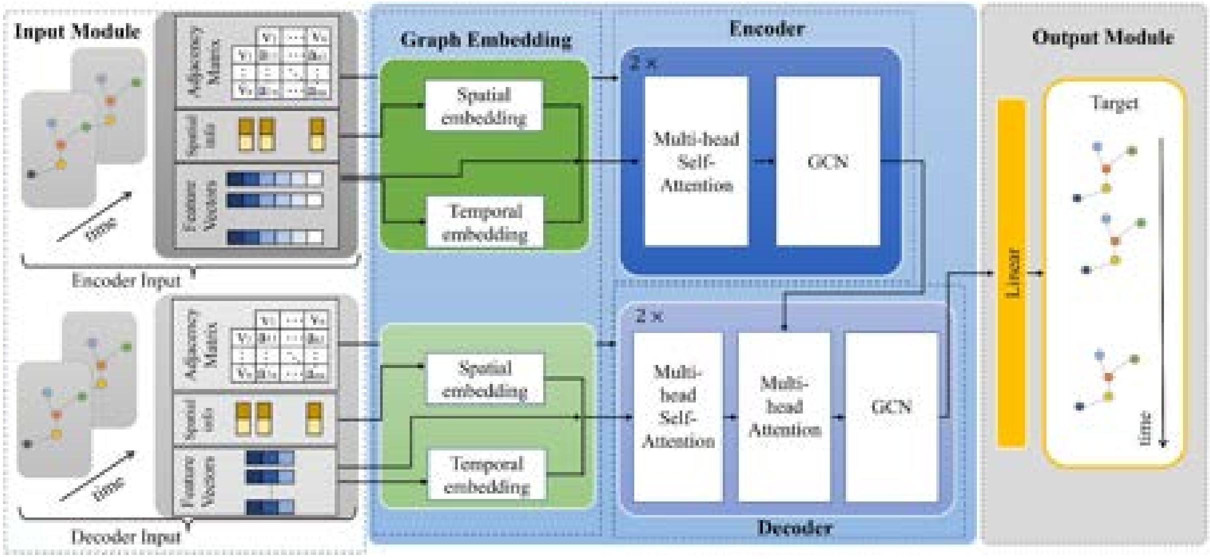

Figure 2. The architecture of aGNN with five modules: (1) input module that constructs node feature vectors, spatial information, and adjacency matrix for both encoder and decoder, (2) graph embedding module that builds spatial embedding as well as temporal embedding from raw data, (3) encoder module that contains two layers of GCN coupled with multi-head attention mechanism, (4) decoder module that couples two layers of GCN with attentions as well as states and hidden information from encoder, (5) output module that generates predicted multi-step spatiotemporal graphs.

information on hydrogeological conditions and initial or boundary conditions as inputs, as opposed to physics-based models, such as MODFLOW for groundwater flow (Harbaugh, 2005) and MT3DMS for contaminant transport (Zheng & Wang, 1999). In essence, DL models engage in end-to-end learning of integrated MODFLOW and MT3DMS based on the provided input and output data sets. The input data of DL include schedules of water discharge, pollutant release, and observations of contamination concentration and groundwater drawdown. The physics-based model provides the ground truth results, which can be used to evaluate the performance of DL models (the details of two benchmark DL models, the diffusion convolutional recurrent neural network (DCRNN) and convolution long short-term memory (ConvLSTM), are described in Appendix A and B, respectively.

We assess the effectiveness of the graph-based DL model by applying three types of learning tasks: transductive learning, inductive learning and model interpretation. In transductive learning, the same graph topology (i.e., the way that nodes and links are arranged) is assumed in both training and testing so that the predictions can be made at observed locations. Inductive learning adopts variable graph topology; thus, the model can infer spatiotemporal variations at unobserved locations from the knowledge learned via the graph used in the training process (Sun et al., 2021; Veličković et al., 2017). The model explanatory task aims to analyze the attribution of multi-point pollution sources. We use the Shapley value method for model interpretation to study the impact of individual pollution sources (details provided in Section 2.3). The post hoc interpretation of the DL model evaluates the influence of each source at key locations after the model is applied. This allows us to quantify the vulnerability of areas to different contaminating activities, providing insights based on the hindsight of the DL model.

2.2. aGNN

Figure 2 illustrates the architecture of aGNN, which is built on the encoder-decoder framework (Sutskever et al., 2014). aGNN consists of five modules: input module, graph embedding module, encoder module, decoder module, and output module. The input module contains two components, that is, encoder input and decoder input, both of which are presented in the spatiotemporal graph. For these high-dimensional inputs, the graph embedding module combines spatial embedding and temporal embedding to transform the raw inputs into feature representations that integrate the cues from the recharge schedule, flow dynamics, and propagation of contaminants. The obtained representations are input into the encoder and decoder modules to extract their interrelations. Both the encoder module and decoder module contain two layers of a graph convolution network (GCN) with attention mechanisms. The advantage of this network is that (a) the attention mechanism flexibly learns interrelations by dynamically focusing on the most relevant parts of the inputs and (b) GCN is beneficial to extract the topological connections of the graph. The output from the final layer of the decoder module goes into the output module, which generates target sequence predictions as induced contaminant movement in space and time. The details of each component are described in the following sections.

2.2.1. Input and Output Modules

The input and output modules construct spatiotemporal graphs that are represented by nodes describing the spatial locations of observation wells and feature vectors characterizing the variations in nodal information in the monitoring network. In data-driven approaches, monitoring data can be represented by spatial graphs, especially for data from irregularly distributed spatial monitors. In mathematics, a graph stores the information of positions, relationships, and measurements of the observations. The structure of a graph is comprised of collections of nodes Vs and edges Es (i.e., $G = (V, E, A)$). In this study, the set of nodes V represents N observation sites, and the node representation x_i of a node V_i can be a vector that stores all the observations. E is the edge linking two nodes (V_i, V_j). The adjacency matrix (A) of dimension $N \times N$ denotes the dependency between every two nodes, thus depicting the topological structure of the graph.

For each node V_i, we define $x_i^{en}(t) \in \mathbb{R}^{d_{en}}$ as the encoder input at a past time step t and $x_i^{de}\left(t'\right) \in \mathbb{R}^{d_{de}}$ as the decoder input at a future time step t'. The feature vector of node V_i is a d_*-dimensional vector (d_{en} for the encoder and d_{de} for the decoder) at each time step. The time length of the encoder input is T_{en} and the decoder input is of length T_{de}. Therefore, the encoder input is represented as $X^{en} \in \mathbb{R}^{N \times T_{en} \times d_{en}}$ (i.e., $X_i^{en}(t) = \left\{x_i^{en}(t - T_{en}), x_i^{en}(t - T_{en} + 1), \ldots, x_i^{en}(t - 1)\right\}$), and the decoder input is represented as $X^{de} \in \mathbb{R}^{N \times T_{de} \times d_{de}}$ (i.e., $X_i^{de}(t) = \left\{x_i^{de}(t), x_i^{de}(t + 1), \ldots, x_i^{de}(t + T_{de} - 1)\right\}$). For a graph of N nodes, its node features are defined as a set i.e., $\{X(t)\} = \left\{X_1^{en}(t), X_1^{de}(t), X_2^{en}(t), X_2^{de}(t), \ldots, X_N^{en}(t), X_N^{de}(t)\right\}$. The inputs are subsequently processed through a dense layer applied to both the encoder and decoder feature inputs. This step aims to transform them into a high-dimensional feature space (32 dimensions in this study), to ensure consistency in spatial and temporal embedding d_{emb} (in Section 2.2.2).

To establish the graph structure, we use a weighted adjacency matrix for an undirected graph. Each entry (i, j) in the adjacency matrix is assigned a non-zero weight if there is an edge connecting node i and node j; otherwise, it is set to 0. The weight for each entry is determined by the distance between node i and node j. Further details about the weight assignment and graph configuration are provided in Section 3.3. The outputs are multi-step predictions presented as spatiotemporal graphs (Y). These predictions cover all nodes, with one or more targets at each node. Thus, the output $y_i(t)$ in each node V_i can be a multi-dimensional vector at time step t. Overall, the output of a node V_i is represented as $Y_i(t) = [y_i(t + 1), y_i(t + 2), \ldots, y_i(t + T_{de})]$, T_{de} is the prediction horizon, and $Y(t) = \{Y_1(t), Y_2(t), \ldots, Y_N(t)\}$ represents the output graph at a snapshot.

2.2.2. Embedding Module

In aGNN, we develop and incorporate an embedding module to encode the spatial heterogeneity in the spatial graph and the order information in the temporal sequence. The spatial heterogeneity embedding applies to geographic coordinates and is constructed with a radial basis network layer. Temporal order information is built on positional embedding that exhibits a synergistic effect combined with the attention mechanism in its applications (Vaswani et al., 2017).

As described in Section 2.1, the hydrogeological setting of an aquifer is heterogeneous and informative in physics-based modeling but can hardly be available for the entire spatial domain. This limitation poses challenges for using vanilla GCNs to incorporate spatial patterns, and the adjacency matrix, primarily based on closeness, may be inadequate for describing implicit spatial dependencies. Furthermore, vanilla GCNs lack layers for transforming spatial coordinate information into latent space, making them less informative for representing implicit spatial dependencies. Several previous studies have addressed spatial heterogeneity through unsupervised graph embedding methods such as DeepWalk (Perozzi et al., 2014), graph Laplacian regularization (Belkin & Niyogi, 2001), or additional GCN layers to learn representations of spatial nodes (Guo et al., 2021). However, these methods are difficult to generalize to variable graph topology, thus inhibiting inductive learning.

Our proposed spatial heterogeneity embedding method that addresses these difficulties is inspired by the positional encoder for geographic coordinates (Klemmer et al., 2022). We first formulate the spatial coordinate matrix

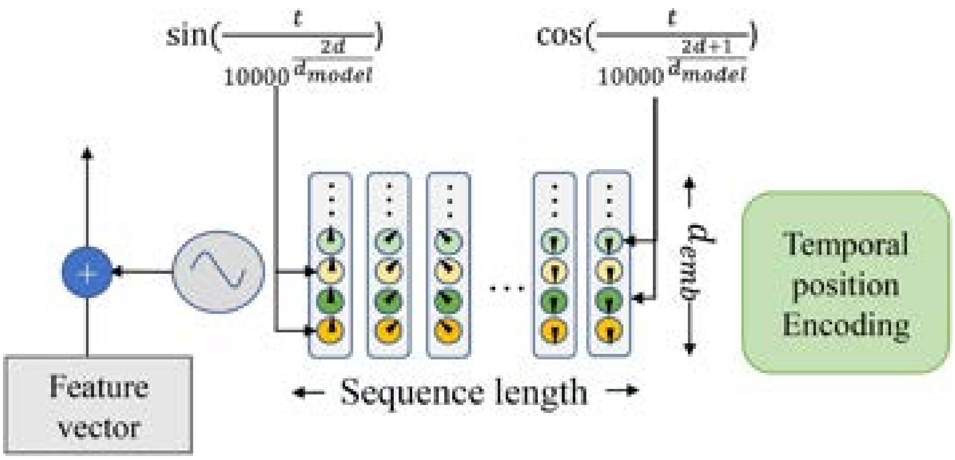
sin(t / 10000^(2d / d_model))
cos(t / 10000^(2d+1 / d_model))
+
d_emb
Temporal position Encoding
Feature vector
Sequence length

Figure 3. Temporal position embedding in aGNN.

$\boldsymbol{C} \in \mathbb{R}^{N\times 2}$, which consist of geographic coordinates (i.e., longitude and latitude) for all nodes $c_1,\ldots c_N$. We then define the spatial heterogeneity embedding as $SE = \varphi(F(\text{C},\text{lo}_{max},\text{la}_{max}),\ \Theta_{\text{sh}})$, where F is a feature function to extract information from the longitude and latitude as $F = \left[\frac{\boldsymbol{C}^{lo}}{\text{lo}_{\max}}; \frac{\boldsymbol{C}^{la}}{\text{la}_{\max}}\right]$, and $\text{lo}_{max},\text{la}_{max}$ are the maximum values of longitude and latitude in the area, respectively, for normalization. The function φ is an RBN network that maps spatial features into an embedding space of dimensionality d_{emb} (i.e., $\varphi : \mathbb{R}^2 \rightarrow \mathbb{R}^{d_{emb}}$). ρ is a quadratic RBF function. φ is implemented as a fully connected neural network for spatial heterogeneity encoding with learnable parameters, including $a \in \mathbb{R}^{N_{ctr}\times d_{emb}}$ as the parameters in the neural network, and $c^{ctr} \in \mathbb{R}^{N_{ctr}\times 2}$ as the parameters of the points in the input space around which the radial basis functions are centered. Hence, N_{ctr} is the number of centers, defined as the number of nodes in this study (i.e., $N_{ctr} = N$) (in Equation 1).

$$\varphi(c_x) = \sum_{i=1}^{N_{ctr}} a_i \rho\left(\left\| c_x - c_i^{ctr} \right\|\right) \tag{1}$$

where a_i is the parameter, and d_{emb} is a hidden dimension (defined as 32 in this study). $\left\| c_x - c_i^{ctr} \right\|$ is the Euclidian distance centered at point c_i^{ctr}.

Temporal embedding enables the use of temporal order information, which is lacking in attention-based temporal learning (Vaswani et al., 2017). Given a time sequence $S = (s_0,s_1,\ldots,s_T)$, the temporal embedding layer forms a finite dimensional representation to indicate where s_i is in the sequence S. Inspired by the positional encoding technique in transformer model (Guo et al., 2021; Vaswani et al., 2017), temporal embedding in our study is a concatenation of sinusoidal transformation to a time order that forms a matrix $TE \in \mathbb{R}^{T\times d_{emb}}$, where T and d_{emb} are the time length and vector dimension, respectively. TE is designed in Equation 2 and Equation 3, where 2d and 2d + 1 denote the even and odd dimensions, respectively, and t is the temporal position in the time sequence. The dimension of the temporal embedding is $d_{emb} \times T$. As depicted in Figure 3, each element in temporal embedding combines the information of the temporal order position and the feature space.

$$TE_{(t,2d)} = \sin\left(\frac{t}{10000^{\frac{2d}{d_{emb}}}}\right) \tag{2}$$

$$TE_{(t,2d+1)} = \cos\left(\frac{t}{10000^{\frac{2d+1}{d_{emb}}}}\right) \tag{3}$$

Sinusoidal functions are used to address an arbitrary sequence length. Unlike spatial embedding, which requires parameter learning, temporal embedding is fixed so that no extra training process is needed. Both temporal embedding and spatial embedding are added to the raw data to form a hidden representation that depicts the spatiotemporal dynamic changes.

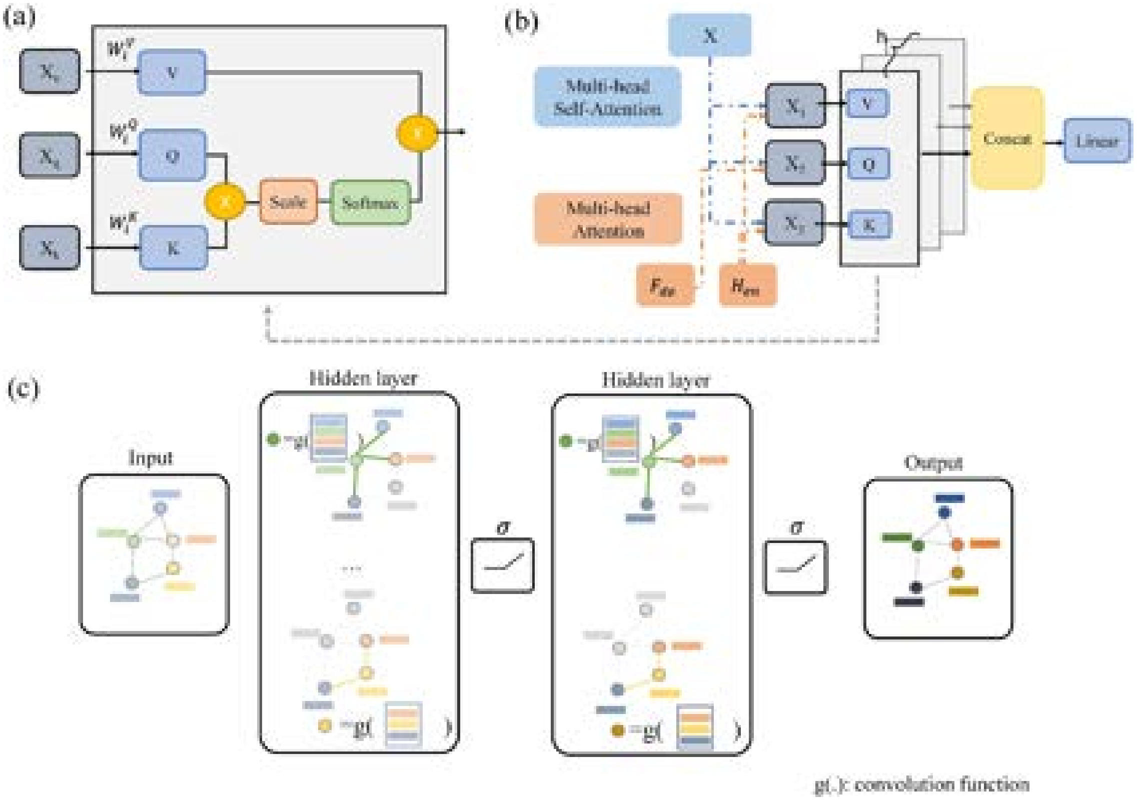

Figure 4. Illustration of attention mechanisms and GCN. (a) Single-head scaled dot-product attention. (b) Multi-head attention mechanisms, where the inputs colored in blue represent multi-head self-attention, and the inputs colored in orange indicate multi-head attention in an encoder-decoder structure, with Q taking the decoder input feature (F_{de}) and K and V taking the encoder hidden feature (H_{en}). (c) Illustration of GCN involving two hidden layers. As in GCN, for a target node, the features of its neighborhood nodes undergo convolutional functions, as in CNN, and propagate to the next layer as the feature input for that specific target node. The hidden layers are used to extract the features from the inputs, ultimately leading to the generation of the output graph.

2.2.3. Encoder-Decoder Module: GCN With Multi-Head Attention

The core of aGNN is the encoder-decoder module. Both the encoder and decoder consist of layers of building blocks, including multi-head self-attention (MSA), GCN and multi-head attention (MAT) blocks. The MSA models the dynamic correlation in the time sequence itself, GCN attempts to capture the spatially relevant dependencies among observations from the monitoring stations, and MAT transfers information from the encoder to the decoder.

2.2.3.1. Attention Mechanism

Attention mechanism is a fundamental operation in our proposed model for learning sequential dependencies. Recurrent neural networks (RNNs), particularly long short-term memory networks (LSTMs), are targeted at sequential modeling but are either uniform or biased against long-term dependencies by designing specific components such as forget gates and input gates in the model (Hochreiter & Schmidhuber, 1997). In contrast, attention mechanisms enable the autonomous prioritization and selection of relevant information in time sequences; these mechanisms are flexible in architecture and fast in training for long-term prediction (Vaswani et al., 2017).

In the attention mechanism, the input data are defined as three different types: queries (Q), keys (K) and values (V). The idea is to map a Q and a set of K-V pairs to an output such that the output represents a weighted sum of V. The weights are determined by the corresponding K and Q. For instance, in scaled dot-product attention (in Figure 4a), Q, K, and V have dimensions of d_{model}. The weights are calculated by taking the dot product between Q and K (Equation 4), dividing it by $\sqrt{d_{model}}$ and then applying a Softmax function to normalize the weight values. Consequently, each weight signifies the strength of the relationship between Q and each K-V pair.

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_{model}}}\right)V \tag{4}$$

2.2.3.2. Multi-Head and Self-Attention Mechanism

Instead of performing a single attention function keys, values, and queries, “multi-head” attention mechanism runs through several attention functions in parallel. This processing technique enables the model to attend to different parts of the sequence independently, accommodating variations such as longer-term dependencies versus shorter-term dependencies. In MAT, Q, K, and V are projected h times through distinct learned linear transformations (Equation 5 and Equation 6). The resulting sequences are concatenated and subjected to a subsequent linear transformation, as depicted in Figure 4b. In multi-head attention mechanism illustrated in the encoder-decoder structure (Figure 2), Q is associated with the decoder input, and K and V are related to the hidden features generated by the encoder. As an additional neural network layer, the multi-head attention layer in the encoder-decoder structure enhances the decoder by allowing it to utilize information from both the final hidden layer produced by the encoder and a specific subset of the decoder input. In essence, attention functions as a filter for extracting relevant contextual information (Luong et al., 2015)

$$\text{multihead}(Q,K,V) = \text{concat}(\text{head}_1,\ldots,\text{head}_h)\,W^m \tag{5}$$

$$\text{head}_i = \text{Attention}\left(X_q W_i^Q, X_k W_i^K, X_v W_i^V\right) \tag{6}$$

MSA specifically focuses on the self-attention mechanism, which applies to inputs to interact with itself and to determine which segments should be given more attention (Figure 4b). Mathematically, Q, K, and V take the same raw input (as $X_q = X_k = X_v$ in Equation 6). MSA allows the model to capture different aspects and dependencies within the input sequence, providing a more comprehensive understanding of the relationships between feature elements.

2.2.3.3. GCN

After MSA blocks, GCN extracts the intermediate representations by exchanging information among nodes through the graph structure to model spatial dependencies. GCNs use graph convolution filters that are designed to model nodal dependencies (Kipf & Welling, 2016). The main idea of GCN is to construct a message passing network, in which the information propagates along neighboring nodes within the graph. The term “convolution” in GCN is similar to that used in CNN (Huang et al., 2016) in terms of weight sharing. However, unlike CNNs, which operate as “fully connected” networks on regular grids (i.e., structured data), GCNs generalize the operations to graphs, applying them to a node's local neighborhood (Kipf & Welling, 2016). The insertion of the adjacency matrix (A) in the graph enables GCN to learn the features of neighboring nodes. From a node perspective, GCN first aggregates feature representations from neighboring nodes and updates the state of each node through linear transformation and nonlinear activation (in Figure 4c). All nodes evolve via links in a graph and transform across one or multiple layers of GCN, allowing complex dependencies among graph-structured data sets to be learned. Analogous to geostatistical interpolation methods (e.g., kriging), GCN can generalize the extracted dependency representations to unseen nodes via inductive learning. Furthermore, GCN has been demonstrated to be an expressive model for knowledge transfer across space because it enables learning through an embedded latent space, thus eliciting highly complex relationships (Bronstein et al., 2017; Sun et al., 2021).

For the input matrix $Z \in \mathbb{R}^{N\times d_{emb}}$, GCNs aggregate the neighboring features of a node and its features through an adjacency matrix (A) with added self-loops (that is, $\tilde{A} = A + I$), and the information propagation is expressed as $\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}Z$, which utilizes graph Laplacian matrix to gather the neighboring information. $\tilde{D}$ is the trace of $\tilde{A}$ that calculates the sum of all the diagonal elements of $\tilde{A}$. A linear projection followed by a nonlinear transformation is subsequently applied, as shown in Equation 7, where σ is the nonlinear activation function and $W \in \mathbb{R}^{d_{emb}\times d_{emb}}$ is the learnable linearization parameter.

$$\mathrm{GCN}(Z) = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}ZW\right) \tag{7}$$

In summary, MAT serves as the link between the encoder and the decoder. The stacked output of the encoder is passed to MAT as V and K, and attention scores are assigned to the representation of the decoder input (i.e., Q). MSA and GCN in the decoder conduct a learning process analogous to the machine translation task, in which the decoder input represents "sentences in a language" that needs to be translated into another "language." Specifically, the output of the decoder constitutes a latent representation that can be further transformed into our target predictions. This transformation involves passing the output through an additional fully connected neural layer in the output module (in Figure 2), ensuring alignment with the targeted dimensions.

2.3. Model Interpretation: SHAP

The Shapley value method (SHAP) quantifies the contribution of each participant in a cooperative game when their actions result in joint outcomes (Shapley, 1953). It can be used to measure feature importance in DL methods (Lundberg et al., 2020; Lundberg & Lee, 2017; Xiong et al., 2022). In this study, we evaluate the influence of each pollutant source via SHAP based on the groundwater contamination at spatial locations of interest predicted by our proposed aGNN.

Essentially, SHAP assesses the importance of each player by considering the average expected marginal contribution of one player's actions in all possible combinations. In a field with N pollution sources/points, let SN denote a subset of the N points and $f_{i,j}(\mathrm{SN})$ denote the induced contaminant concentrations at a given location cell (i,j) when only pollutant sources in subset SN contaminate the groundwater. The Shapley value $\Phi_{i,j}$ for a specific point d ($d = 1,\ldots, n$) can be represented by Equation 8.

$$\Phi_{i,j} = \sum_{\mathrm{SN}\subseteq N\backslash} \frac{|\mathrm{SN}|!(n-|\mathrm{SN}|-1)!}{N!}\left[f_{i,j}(\mathrm{SN}\cup\{d\}) - f_{i,j}(\mathrm{SN})\right] \tag{8}$$

where $|SN|$ is the number of pollution points in subset SN and $f_{i,j}(\mathrm{SN}\cup\{d\}) - f_{i,j}(\mathrm{SN})$ is the marginal contribution of point d represented by its induced contaminations $f_{i,j}$. The value of $f_{i,j}(.)$ can be obtained by the physics-based groundwater flow and contaminant transport model (i.e., MODFLOW and MT3DMS) as well as aGNN. Therefore, we can not only obtain the influences of each individual contaminant outlet but also compare the influences interpreted based on aGNN against the ground truths provided by the physics-based model.

3. Case Studies

3.1. Groundwater Flow and Contaminant Transport Model

We consider a contaminant transport process in response to transient stresses due to varying anthropogenic pollution activities. Contaminant transport may be affected by multiple processes, such as advection, dispersion, molecular diffusion, and chemical reactions. In this study, we examine advection and dispersion (advection dominating over dispersion) in contaminant transport, two dominant processes at the field scale, while ignoring molecular diffusion and chemical reaction processes. Physics-based models (i.e., MODFLOW and MT3DMS) can simulate the groundwater flow and solute transport in the study area, accounting for the effects of groundwater and pollution source discharge on the movement of contamination plumes. The natural porous formations of aquifers and sources of contamination are heterogeneous. MODFLOW (Harbaugh, 2005) constructs heterogeneous geologic media with a field of spatially varying hydraulic conductivity. MT3DMS (Zheng & Wang, 1999) models multiple contamination sources that are located at different locations and have different release strengths that drive pollutant movement in contamination transport modeling.

3.2. Study Area and Scenario Design

In this study, we design two synthetic study sites featuring unconfined aquifers for method development and validation. The first study site covers an area of 497,500 m^{2,} as shown in Figure 5a, which is discretized into 30 columns and 15 rows (each cell is 50 m by 50 m) in MODFLOW. The hydrological boundary is modeled as two sides of no-flux boundaries and two sides of constant head (i.e., 100 and 95 m, respectively) (Figure 5a), and the

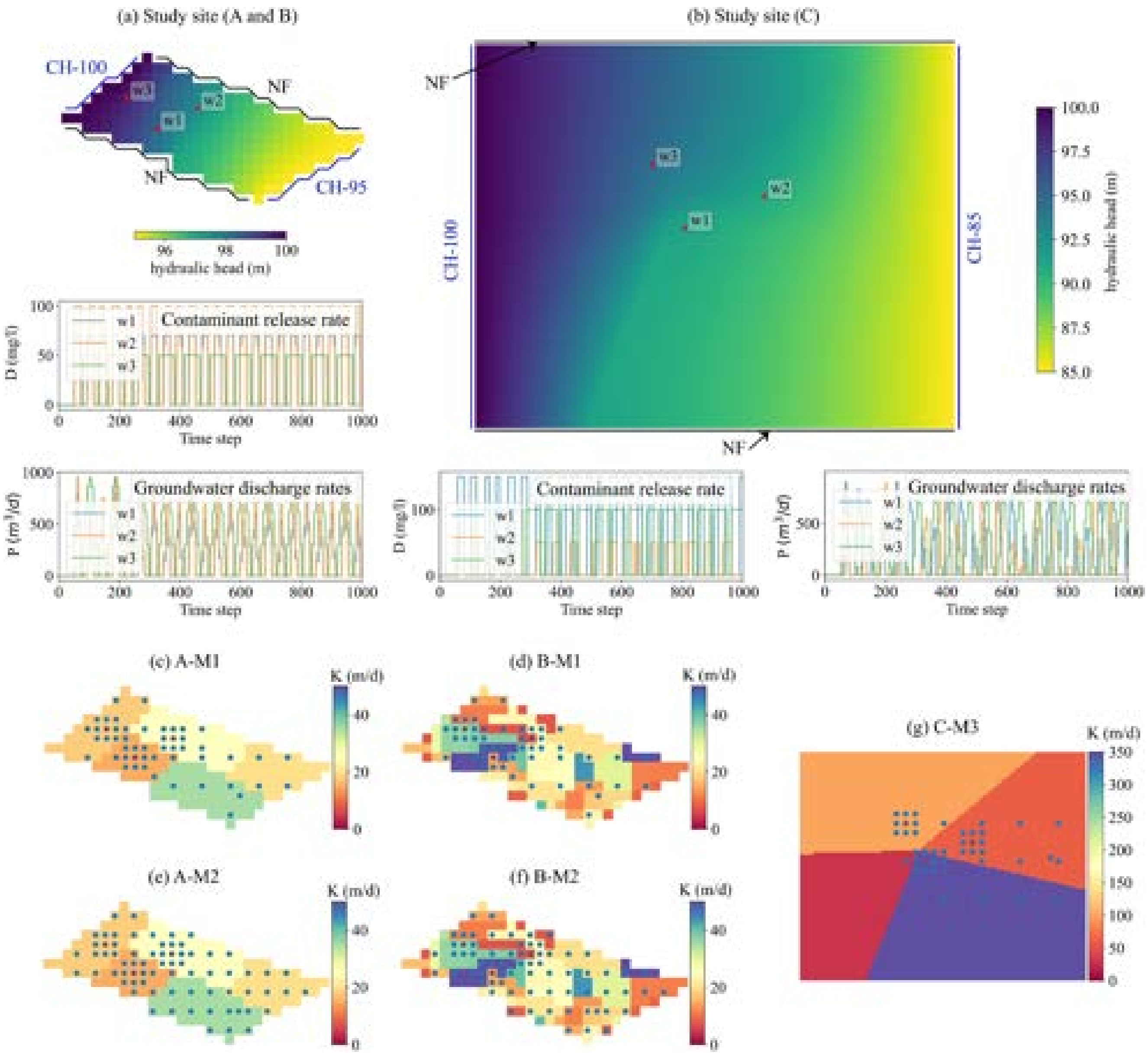

Figure 5. Study sites with varying conductive heterogeneities and different monitoring networks. (a) The study area in Scenarios A and B, which are bounded by constant-heads (CHs) and no-fluxes (NF), and three contaminant discharge locations (i.e., W1, W2, and W3 in red dots); the colored area represents the initial hydraulic head. The two subfigures under the study site depict the contaminant release rates of three contaminant sources. (b) The study area in Scenario C is much larger than that in the first case study, and the proportional size is not exactly reflected in the plot. The two subfigures under the study site show the contaminant release rates of three contaminant sources. Five cases with two fields of hydraulic conductivity (scenarios A, B, and C) depicted in colored areas and two spatial arrangements of observation wells (M1, M2, and M3) with blue dots as observation wells are shown in (c) A-M1, (d) B-M1, (e) A-M2, (f) B-M2, and (g) C-M3.

natural hydraulic gradient drives the groundwater flow. The hydrogeological settings are as follows: the specific yield is 0.3, the specific storage is 0.0001 1/m, and the porosity is 0.3. To investigate the influence of hydraulic conductivity (HC) heterogeneity on contaminant transport modeling, two fields of hydraulic conductivity are considered in this study: (a) Scenario A: a field of five different zones with hydraulic conductivity varying from 15 to 35 m/day (Figures 5c and 5e) and (b) Scenario B: a field of more diversified hydraulic conductivity ranging from 0 to 50 m/day (Figures 5d and 5f). In MT3DMS, contaminant transport is modeled with a uniform longitudinal dispersity of 30 m across the study area. The polluting activities in the study domain involve three injection wells intermittently discharging polluted water into the groundwater aquifer, which are represented as W3 located upstream and W1 and W2 located relatively downstream (Figure 5a). The periodic discharge schedule and contaminant release rates of the three wells are shown in Figure 5a, which depict partial sequences of an overall observation period of 2,100 d. Both the groundwater flow and contaminant transport models are transient with 2,100 time steps, each with a time length of 1 d.

The second study site (scenario C) covers an expansive area of 180 km^2, which is approximately 360 times larger than the first study site (Figure 5b). It is discretized into 120 columns and 150 rows, with each cell measuring 100 m by 100 m in MODFLOW. The hydrogeological boundary is modeled with two sides as no-flux boundaries and two sides as constant-head boundaries (specifically, 100 and 85 m, respectively). The range of hydraulic conductivity heterogeneity is large, as represented by four distinct zones with hydraulic conductivities varying from 30 to 350 m/day (Figure 5g). The remaining hydrogeological settings are consistent with those in the first study site. Similarly, polluting activities involve three injection wells (i.e., W1, W2, and W3) intermittently discharging polluted water into the groundwater aquifer following a periodic discharge schedule and specific contaminant release rates, as depicted in Figure 5a.

In this study, monitoring systems (i.e., spatial arrangements of monitoring locations) provide the data, including groundwater drawdowns (GDs) and contaminant concentrations (CCs), on a daily basis. Figures 5c–5g demonstrate three different spatial arrangements of observation wells (i.e., M1, M2, and M3) that contain 51 71, and 41 observation wells, respectively. Notably, in the significantly larger study area of Scenario C, there are fewer observation wells. Therefore, in comparison, sparsely distributed observed data pose greater challenges in learning data dependency relationships. The goal is to examine how the size of the data influences the learning process regarding contamination movement in response to discharge. In summary, we observe five different cases, including three monitoring networks in three fields of hydraulic conductivities, referred to as A-M1, A-M2, B-M1, B-M2, and C-M3.

3.3. Data Preparation

The contaminant transport data sets are generated by MODFLOW and MT3DMS simulations for the five cases. Of all the data samples, 80% are used for training DL models, and the remaining 20% are used for performance evaluation. All the DL models are trained via batch optimization for large spatiotemporal data sets with a batch size of 16 for 400 epochs, and the outputs are the predictions of GD and CC at the observed locations with a time horizon of 50 time steps.

The DL models include DCRNN (Li et al., 2017) (described in Appendix A), aGNN, aGNN-noE (i.e., a variant of aGNN that has no embedding module), and ConvLSTM (Shi et al., 2015) (i.e., a CNN-based algorithm depicted in Appendix B). All the algorithms use the encoder-decoder framework, but differences exist in the design of the inputs. Specifically, aGNN, aGNN-noE and ConvLSTM receive external inputs in both the encoder and decoder, but DCRNN takes only external inputs from the encoder. For all the algorithms, the input features consist of three types, that is, static feature (S), historical behavior (H) and plan feature (F). S is a feature that represents coordinate information, including longitude and latitude. H contains 4 different inputs, detailing the two plans for groundwater discharge and contaminant release individually, and two inputs as the monitored GD and CC. F contains the planned groundwater discharge and contaminant release in the period for which we predict the field of CC. The sequence lengths of the encoder input (i.e., T_{en}) and the decoder input (i.e., T_{de}) vary among the different algorithms depending on the features used. In aGNN, aGNN-noE and ConvLSTM, the encoder modules take H and S as input with a length of 20 time steps. Specifically, S input is duplicated for the length of the time steps according to H, and then concatenated to form an input of size 6. The decoder module takes F and S as inputs (i.e., with a size of 4) for 50 time steps. Due to the algorithm settings in DCRNN, the encoder module takes H, S and F as inputs (i.e., with a size of 8) that are of the same length as 50 time steps, and no input is required in the decoder. The shapes of the input features also differ case by case due to the type of algorithm used and the monitoring system considered. Graph-based algorithms store the input features in nodes, so the spatial dimensions are 51, 71, and 41 for cases M1, M2 and M3, respectively. In contrast, ConvLSTM uses an image-shaped matrix to represent the 2D research area of shape 15 × 30 in Scenario A and B, and 8 × 11 in Scenario C. Table 1 summarizes the input dimensions for all four algorithms.

Graph-based DL learns from a predesigned graph topology. In this study, we use an undirected graph, making no assumptions about the direction of contaminant movement. To ensure concise and representative connections in the monitoring network, in Scenarios A and B, each node's neighborhood includes adjacent nodes within a range from 1 cell length (50 m) to a maximum distance of 3 cell lengths (150 m, with each cell measuring 50 m),

Table 1
Dimensions of Inputs and Number of Parameters in Different Algorithms for Three Monitoring Networks

		ConvLSTM	DCRNN	aGNN(/-noE)
Encoder input length		20	50	20
Encoder input feature	M1	15 × 30 × 6	51 × 8	51 × 6
	M2	15 × 30 × 6	71 × 8	71 × 6
	M3	8 × 11 × 6	41 × 8	41 × 6
Decoder input length		50	50	50
Decoder input feature	M1	15 × 30 × 4	–	51 × 4
	M2	15 × 30 × 4	–	71 × 4
	M3	8 × 11 × 4	–	41 × 4
Num of parameters	M1–3	59,266	97,888	58,072 (54,978)

incorporating a finer grid around the injection wells. In Scenario C, the neighborhood expands from 5 cell lengths (500 m, with each cell measuring 100 m) to 20 cells (i.e., 10,000 m), also utilizing a finer grid around injection wells. Figures 6a–6c depict the configurations of the three monitoring systems, one for each scenario.

The strength/weight (i.e., w) of a link is the spatial closeness of the attached nodes (i.e., $w(v_i,v_j)$). To ensure that the closer the two nodes are, the greater the strength of the link in between them (i.e., the shorter the distance between the nodes is, the heavier the weight assigned to their link is), we assign weights as long-distances dl subtracted by the cell length between the nodes. In scenario A and B, $dl = 4$ (ensuring the smallest weight is greater than 0), the weight $w(v_i,v_j)$ is determined by $4 - \frac{d(v_i,v_j)}{50}$. In scenario C, $dl = 25$, the weight $w(v_i,v_j)$ is calculated as $25 - \frac{d(v_i,v_j)}{100}$, where $d(v_i,v_j)$ is the Euclidean distance between nodes v_i and v_j.

The node strength (i.e., the sum of the weights of all the node edges) provides information about contaminant propagation at a location in the graph. Figures 6a–6c shows that the contaminant sources and their neighbors have large node strengths (i.e., greater centrality) in the three monitoring networks, so these observation locations are emphasized when training the graph models. In inductive learning, we add unmonitored locations into the graph in transductive learning; thus, a new graph topology is adopted. In Scenarios A and B, all cell locations of

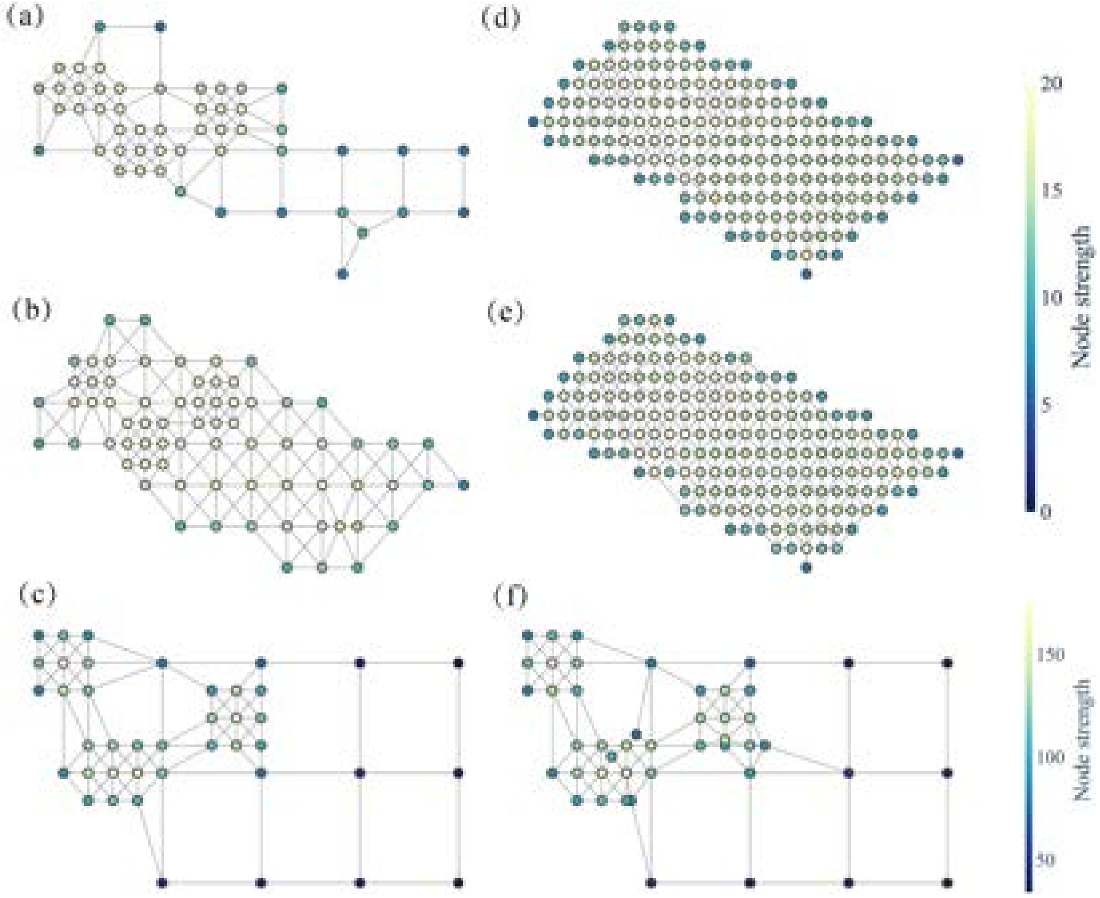

Figure 6. Graph topologies of two monitoring networks for graph-based DL, as (a) M1 in transductive learning, (b) M2 in transductive learning, (c) M3 in transductive learning, (d) M1 in inductive learning, and (e) M2 in inductive learning, and (f) M3 in inductive learning.

Table 2
The Statistics and Modeling Errors (R^2 and RMSE) of Groundwater Drawdown (GD) and Contaminant Concentration (CC) in Five Cases

	A-M1		A-M2		B-M1		B-M2		C-M3	
	GD	CC	GD	CC	GD	CC	GD	CC	GD	CC
Median	1.97	8.15	2.29	7.74	1.41	7.68	1.74	7.16	10.52	0.25
Range	4.80	23.84	4.91	23.90	4.77	31.67	4.80	31.84	14.15	21.89
STD	1.41	4.38	1.50	4.17	1.33	4.98	1.44	4.64	2.462	2.92
RMSE_a										
ConvLSTM	0.271	1.361	0.266	1.066	0.263	1.644	0.205	1.408	0.614	2.467
DCGCN	0.062	0.069	0.041	0.077	0.051	0.118	0.068	0.139	0.301	0.423
aGNN-noE	0.023	0.055	0.024	0.068	0.014	0.056	**0.014**	0.068	0.109	0.116
aGNN	**0.011**	**0.024**	**0.0106**	**0.032**	**0.012**	**0.028**	0.018	**0.041**	**0.026**	**0.065**
R_a^2										
ConvLSTM	96.32%	90.35%	96.85%	93.48%	96.12%	89.11%	98.01%	90.81%	90.27%	53.48%
DCGCN	98.71%	99.92%	99.93%	99.90%	99.84%	99.82%	99.77%	99.74%	99.63%	88.26%
aGNN-noE	99.97%	99.98%	99.96%	99.98%	99.99%	99.98%	**99.99%**	99.98%	99.70%	99.85%
aGNN	**99.99%**	**100%**	**99.99%**	**99.99%**	**99.99%**	**100%**	99.98%	**99.99%**	**99.98%**	**99.95%**

Note. The bold values indicate the results of the best performing model in each case.

MODFLOW are considered as unmonitored sites, whereas for a significantly larger site in Scenario C, four unmonitored sites are added. To preserve the centrality of the contaminant sources (i.e., with relatively larger node strengths) in the graph, the added links are defined as follows: an unmonitored node links only to the monitored node or other unmonitored nodes within a neighboring distance of one cell length. Its link strength $w(v_i, v_j)$ is set according to the assignment rule used in each scenario. The graphical structure used in inductive learning is shown in Figures 6d–6f, representing the predictive area for Cases M1, M2, and M3.

4. Results and Discussion

4.1. aGNN Modeling Performance

The values of our modeling targets (i.e., GD and CC) differ by case due to heterogeneities in aquifers (scenarios A, B, C) and the observations in different monitoring systems (M1–M3). Table 2 provides an overview of the statistical characteristics of our modeling targets across the entire data set, partitioned into 80/20 training and testing splits. Additionally, the results showcase the performance of aGNN on the test data for five distinct cases. In summary, the range of CCs significantly exceeds that of GDs, reaching approximately 5 times greater than that of GDs, and demonstrating greater dispersion (indicated by a greater standard deviation). The training of the deep learning (DL) model requires the transformation of the multi-target task involving GD and CC into a single-target objective. During this transition, we employ a weighted sum, with the weights determined by the proportion of the two range sizes. In this study, a weight of 5 was assigned to CC, and a weight of 1 was assigned to GD. Table 2 also shows that aquifer heterogeneities affect the induced variations in GD, but to a much lesser extent than those in CC, which is particularly represented in the different value ranges between Scenarios A and B. This is because the hydraulic head is less sensitive to conductivity heterogeneity than to concentration (Kitanidis, 2015; Mo et al., 2019). In Scenario C, the accuracies of all the models decrease, as indicated by both the R^2 and RMSE values. This is attributed to the larger site and fewer monitoring wells, which increase the difficulty of obtaining accurate predictions.

As the aquifer settings and provided data produce varying degrees of impact on the forecasting task, we use these cases to assess the performance of four models, including aGNN and three benchmark models, namely, DCGCN, ConvLSTM, and aGNN-noE (i.e., a variant of aGNN that has no embedding module). We use two measures, that is, R_a^2 and RMSE_a, to analyze the overall spatiotemporal predictions ($x_{s,t}$) of the field across space S and time T (Equations 9 and 10).

$$R_a^2 = 1 - \frac{\sum_{x=1}^{S}\sum_{t=1}^{T}\left(x_{s,t} - \hat{x}_{s,t}\right)^2}{\sum_{s=1}^{S}\sum_{t=1}^{T}\left(x_{s,t} - \frac{1}{T\cdot S}\sum_{s=1}^{S}\sum_{t=1}^{T}x_{s,t}\right)^2} \tag{9}$$

$$\text{RMSE}_a = \sqrt{\frac{1}{S\cdot T}\sum_{x=1}^{S}\sum_{t=1}^{T}\left(x_{s,t} - \hat{x}_{s,t}\right)^2} \tag{10}$$

Table 2 shows that GNN-based algorithms obtain RMSE$_a$ values smaller than 0.15 mg/L for CC and 0.1 m for GD and high R_a^2 values greater than 99% for both GD and CC. The performance results suggest a strong correlation between the predicted values of the GNNs and target values, indicating that the model effectively captures variations and dependencies in the data across space and time; thus, GNNs are beneficial for learning monitoring data that are spatially uneven. The sparsely distributed observations as image-like data sets do not provide enough neighboring information, so the convolutional operator in ConvLSTM cannot extract useful spatial information from the spatial "image." In contrast, GNNs harness graph-structured data with nodes and links that embed spatiotemporal information into node features and adjacency matrices. As a result, GNNs are generalizable as prediction tools for monitoring networks of different topologies. Among all the algorithms, aGNN obtains the lowest RMSE$_a$ and highest R_a^2 in almost all five cases (Table 2), indicating its superior performance in modeling contaminant transport in unevenly distributed monitoring systems compared to that of the other algorithms.

Thereafter, we focus on the results of CC, as it is our main target in modeling contamination transport. We further analyze the characteristics of the various models in their spatial and temporal predictions separately, by using two measures as RMSE$_s$, which (Equation 11) depicts the spatial variations in the modeling accuracy, and RMSE$_t$ (Equation 12), which illustrates the varying temporal prediction precisions.

$$\text{RMSE}_s = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(x_{s,t} - \hat{x}_{s,t}\right)^2} \tag{11}$$

$$\text{RMSE}_t = \sqrt{\frac{1}{S}\sum_{s=1}^{S}\left(x_{s,t} - \hat{x}_{s,t}\right)^2} \tag{12}$$

The fields of RMSE$_s$ in Figure 7 demonstrate the prediction errors of the four models. The RMSE$_s$ of ConvLSTM is high across space (mostly above 1 mg/L). Comparatively, the RMSE$_s$ s of predictions by DCRNN are much smaller. In A-M1, B-M1, A-M2, and B-M2, specifically, DCRNN obtains a field of RMSE$_s$ with values all less than 0.3 mg/L except for some locations near the downstream of the discharge ports. This is because as advection is dominant in the advection-dispersive transport in this aquifer, the downstream areas are more affected by pollution activities over time, whereas the upstream areas are less affected. Therefore, the DL model needs to capture the spatial variation, especially downstream of the three contaminant sources that are more sensitive to pollution release. Since the attention mechanism adjusts the focus on spatial dependencies by dynamically assigning importance to neighbors according to the inputs, this component in the model contributes to better spatial learning. As illustrated in Figure 7, both aGNN-noE and aGNN outperform DCRNN with relatively small variations in RMSE$_s$ over the area, indicating that the attention-based graph convolutional network outperforms the diffusion convolutional graph network. At a substantially larger site in C-M3, where the contaminant spread is less dispersed within the monitoring graph due to increased distance between nodes, the performance ranking of the four algorithms is similarly demonstrated. Of all algorithms, aGNN achieves the least RMSE$_s$ variation over the site, further demonstrating the effectiveness of the attention mechanism and the integrated spatial embedding in capturing spatial variations. In the scenarios with varying complexity, aGNNs obtain small RMSE$_s$ with nuanced discrepancies in Scenarios A and B, illustrating that aGNNs are adaptable for determining the spread of contaminant transport through complicated pathways. Additionally, we employ Relative Absolute Error (RAE) into our analysis, which measures the absolute difference between predicted and true values, normalized by the range of the true value at each spatial location. The findings also highlight the reduced RAE observed when utilizing aGNNs. Additional analyses are provided in Appendix D.

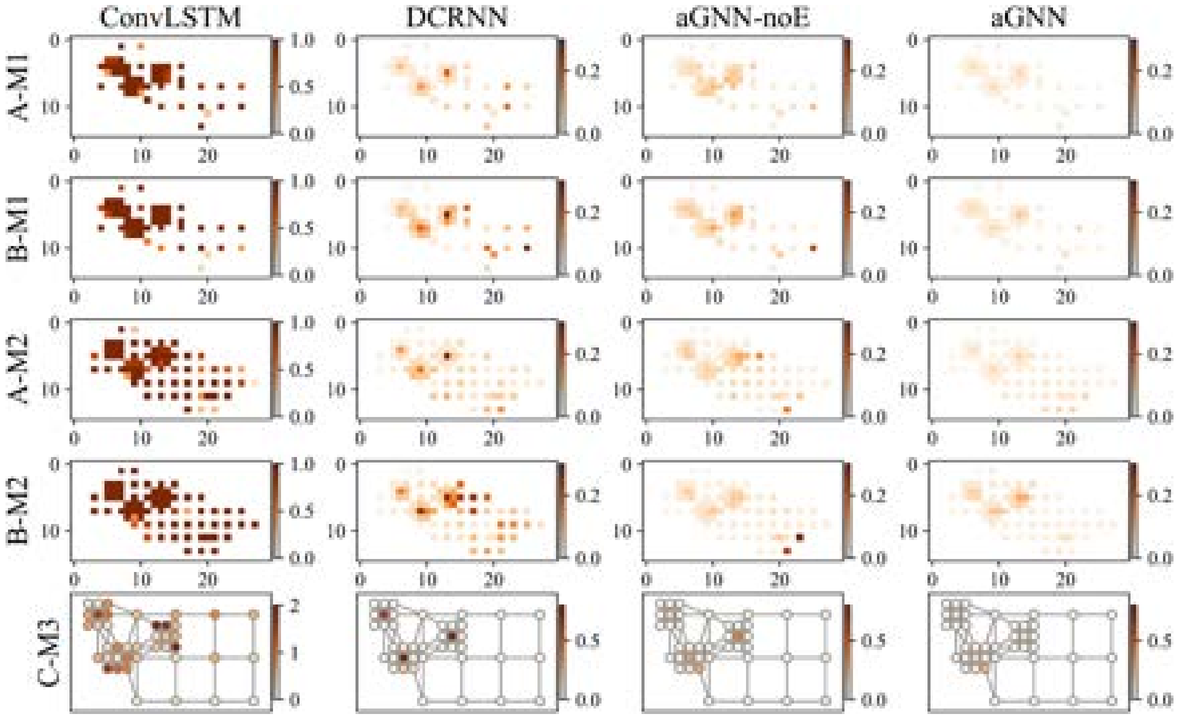

Figure 7. The spatial variation of modeling accuracies in terms of RMSE$_s$ (mg/L) by four algorithms in five cases (i.e., A-M1, A-M2, B-M1, B-M2, C-M3).

In the assessment of temporal variations in modeling accuracy, ConvLSTM performs consistently worse than the other algorithms in multi-step prediction up to 50 time steps (Figure 8). The RMSE$_t$ values obtained by DCRNN are much smaller, but the values vary over time. The unstable performance of temporal modeling by DCRNN is mainly due to the limitations of RNN in learning very long sequences (Bengio et al., 1994). At each time interval, RNN-based methods share the same weights, thus restricting the ability to model the complexity of temporal dynamics. In contrast, both aGNN-noE and aGNN can leverage the self-attention mechanism to adjust the temporal weights dynamically based on time series input; thus, the RMSE$_t$ of aGNNs are lower, especially for near-future predictions. However, the attention mechanism is indifferent to the order of the sequence (Adel & Schütze, 2016), the correlation in neighboring time slices can be ignored, and the long-term trend may be misled by the input data when using the vanilla attention mechanism. As shown in Figure 8, the RMSE$_t$ *s* of aGNN-noE increase over time in different cases, and consistently surpass those of aGNN. The temporal position embedding in aGNN considers the chronological order in the sequence, and it generates a temporal bias in the model, thus inducing more accurate predictions up to 50 time steps.

Moreover, modeling temporal variations in contaminant transport is more difficult in aquifers with higher heterogeneity. Figure 8 shows that the RMSE$_t$ *s* of both ConvLSTM and DCRNN in Scenario B remain larger than the corresponding error terms in Scenario A for different time steps. In Scenario C, the errors RMSE$_t$ *s* either increases or exhibit more fluctuations over time at the much larger site. In contrast, the attention-based GNNs, especially aGNN, exhibit the least significant deviation in prediction accuracy in different cases, which demonstrates the effectiveness of the attention mechanism and the embedding layers in learning temporal transport.

4.2. Inductive Learning

For inductive learning, we analyze the transferability of aGNN to unmonitored sites based on the ground truths provided by the physics-based model, as aGNN obtains the best performance in the transactive task. In the multi-step inductive prediction, our evaluation starts from the time when none of the contaminant source discharges ever begin, to the time when the first contaminant release starts at the beginning of our prediction. That is, the overall sample size is 50, and each sample involves inductive spatial predictions for 50 forward time steps in our evaluations. In addition, the graph topology incorporates the unmonitored site (as shown in Section 3.3); thus, it is different from the topology employed in the training process.

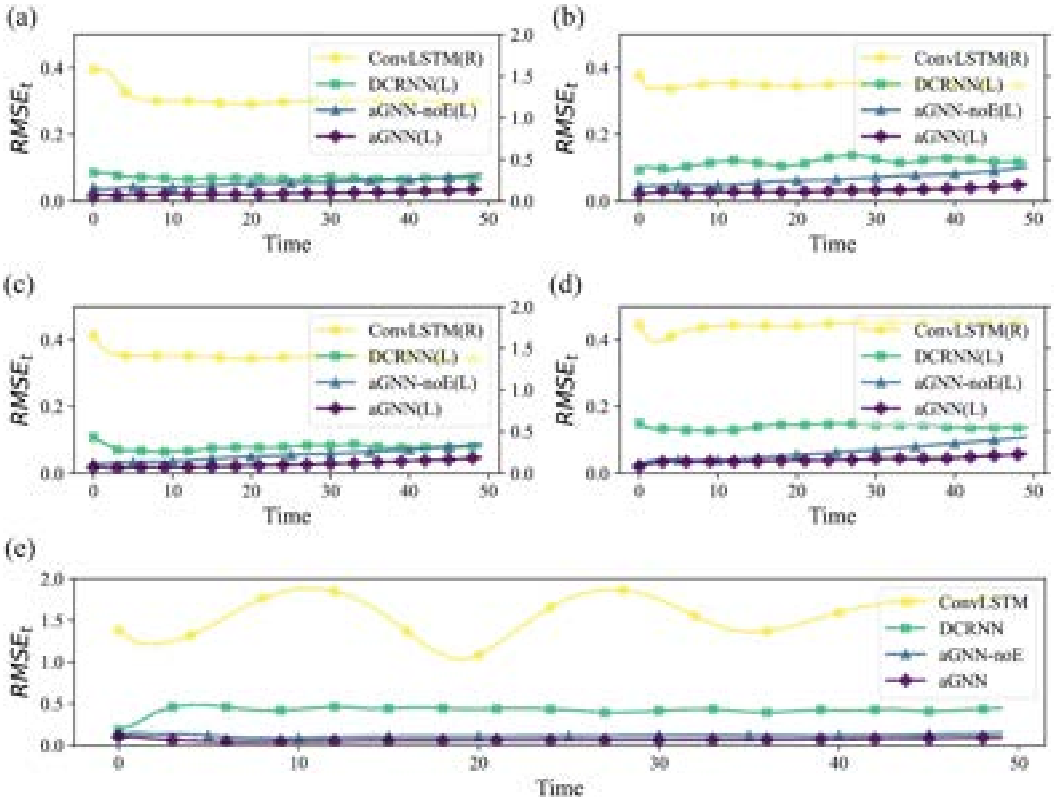

Figure 8. The temporal variations in modeling accuracy of four algorithms in terms of RMSE$_t$ over 50 time steps in (a) A-M1, (b) B-M1 (c) A-M2 (d) B-M2 (e) C-M3. In (a)–(d), RMSE$_t$s of DCRNN, aGNN-noE and aGNN are on left *y*-axis, and RMSE$_t$ of ConvLSTM is one order of magnitude larger than that of the other three algorithms, so they are shown on right *y*-axis.

Figure 9a illustrates that RMSE$_s$ of the near-future predictions (i.e., 10 time steps) are small all over the site, meaning that the unmonitored locations can be precisely predicted regardless of the difficulties in the problem settings brought by the high conductive heterogeneity and the small amount of monitoring data used in training. RMSE$_s$ of distant future predictions (i.e., 50 time steps) can accumulate to relatively large values at some unmonitored locations, especially those close to the contaminant release outlets where CC is sensitive to both the contaminant discharge and the underlying hydraulic conductivities.

Comparing monitoring networks M1 and M2, the spread of high RMSE$_S$ (i.e., greater than 0.3 m) is smaller in M2, as the monitoring system provides more observations than that in M1. The additional monitors help infer values at unmonitored sites with higher accuracy, as expected, especially for relatively distant future predictions (i.e., in 50 time steps); thus, this demonstrates the advantages of using more observations in the training process. In the case of a large site in Scenario C with fewer monitoring wells in M3, we select four unmonitored locations comparatively close to injection wells that exhibit noticeable variations in CC. Overall, for all five cases, most of the area has RMSE$_S$ values smaller than 0.5 mg/L, indicating that aGNN achieves relatively high accuracy for most unmonitored sites.

Figure 9b shows the temporal variations in the prediction accuracy over the whole study domain in terms of RMSE$_t$. In cases A-M1, A-M2, B-M1, and B-M2, long-term inductive predictions are challenging, as RMSE$_t$ increases over time, but all RMSE$_t$ are not greater than 0.2 mg/L, which is a relatively small error compared to the median in Table 2. In addition, RMSE$_t$*s* in A-M2 decrease over time since the conductive heterogeneity in Scenario A is low and the number of monitors is large. For aquifers with different conductive heterogeneities, RMSE$_t$*s* are greater in B-M2 than in A-M2, but the largest discrepancy is less than 0.1 mg/L; additionally, the discrepancies in RMSE$_t$*s* between B-M1 and A-M1 are much less significant, as the two lines coincide with each

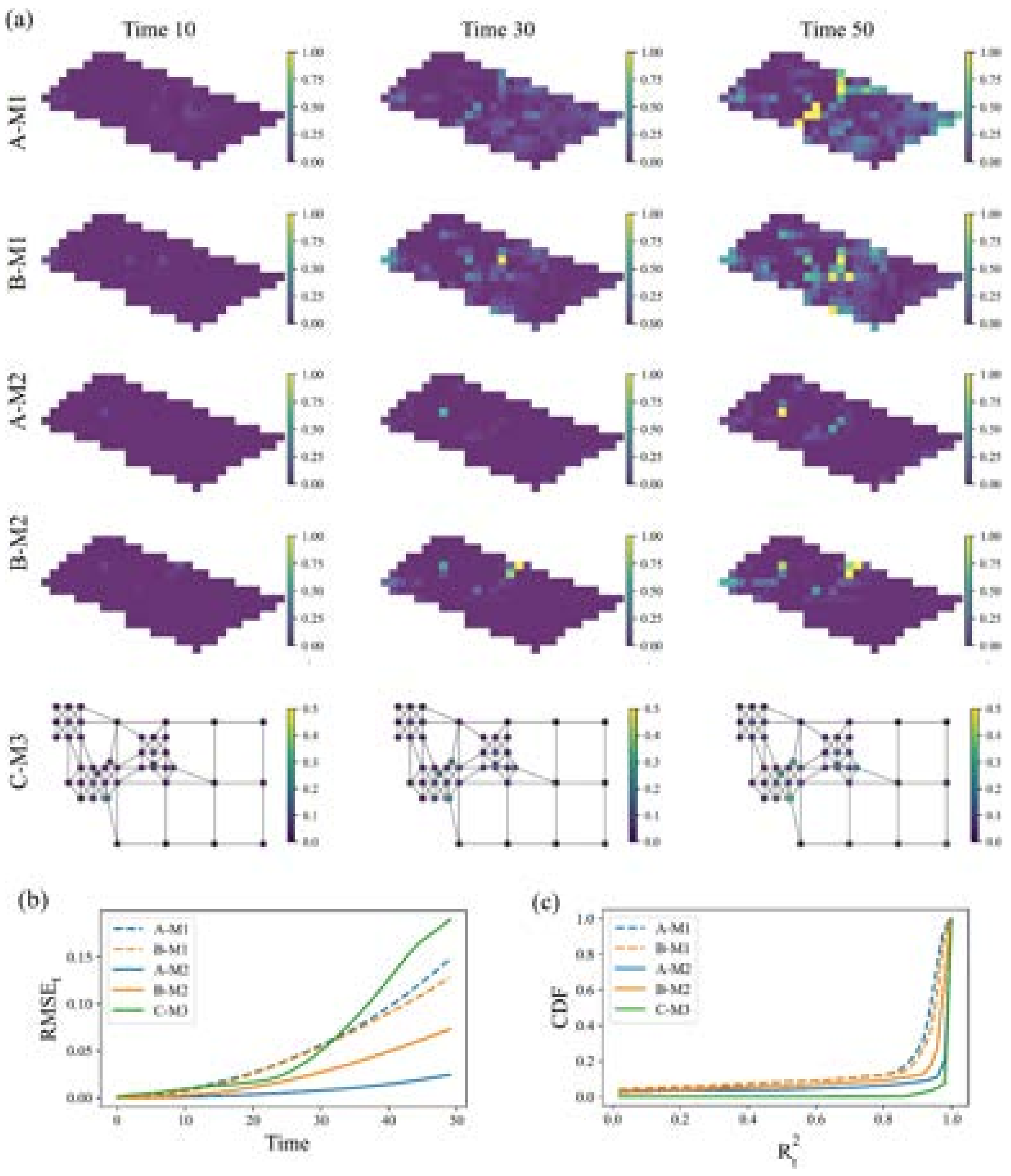

Figure 9. The prediction accuracy in inductive learning of aGNN trained in five scenarios, that is, A-M1, A-M2, B-M1, B-M2, C-M3. (a) Spatial RMSE_s of prediction for three time steps forward (i.e., 10, 30, 50), for Scenario C-M3, the four unmonitored locations are circled (b) the temporal error as RMSE_t along 50 time steps (c) distribution of R_t^2.

other. In C-M3, the presence of fewer monitoring wells in a larger study area intensifies the challenges of long-term inductive predictions. Despite an increase in long-term prediction errors, especially after 20 time steps, near-future predictions remain relatively accurate, with RMSE_t values less than 0.03 mg/L. However, inductive analysis is likely impractical for complex sites exhibiting high heterogeneity. Therefore, it is recommended that a relatively large number of monitoring wells (nodes) be obtained when implementing inductive learning for sites encompassing a large area. In the case of a highly heterogeneous site, the incorporation of hydrogeological information into the modeling process is suggested to improve the inductive results. Figure 9c illustrates the performance shown in terms of R_t^2, which represents how well the variations in CC over space are captured at different time steps (Equation 13). CDFs of R_t^2 show that the majority of R_t^2 values are greater than 80% (as the probability $P\left(R_t^2 \leq 80\%\right) < 20\%$) in all five cases.

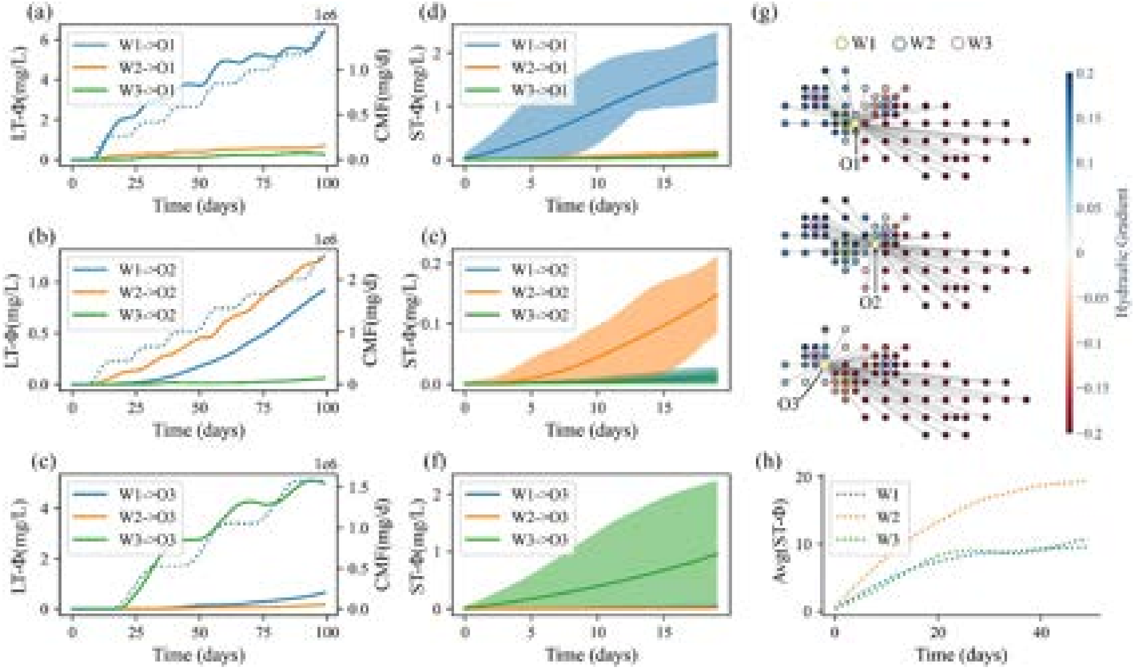

Figure 10. The influence as value from three contaminant sources (W1, W2 and W3) on three observation wells (O1, O2, and O3) by SHAP based on aGNN in B-M2, including long term effects on (a) O1, (b) O2, (c) O3 and short-term effect on (d) O1, (e) O2, (f) O3. The cumulative mass flow (CMF) of W1, W2, and W3 are depicted in dashed lines in (a), (b), (c), respectively. (g) The locations of the three contaminant sources and affected locations, with three subplots representing the hydraulic gradient relative to O1, O2, and O3. (h) The averaged short term Φ value at the contaminant sources effected by its own contaminant release.

$$R_t^2 = 1 - \frac{\sum_{s=1}^{S} (x_{s,t} - \hat{x}_{s,t})^2}{\sum_{s=1}^{S} \left(x_{s,t} - \frac{1}{S} \sum_{s}^{S} x_{s,t} \right)^2} \tag{13}$$

4.3. Influences of Contaminant Sources

In this section, we quantify the influences of each contaminant source (i.e., Φ) by using SHAP based on the predictions of aGNN. We examine two types of influences: the short-term effect (i.e., ST-Φ), which describes the time lagged influence of each source once it ceases to work at any time point, and the long-term effect (i.e., LT-Φ), which denotes the accumulative influence of each contaminant source since the start of pollutant emissions. Mathematically, the term $f_{i,j}(S^*)$ in Equation 8 for ST-Φ is specified as $F_{i,j}^{(t,t+t_k)}(S^*)$ for $t_k \in [1,2,\ldots,20]$. This formulation represents the induced contaminant concentrations at a specific location cell (i,j) when pollutant sources in injection wells set S^* cease releasing contaminated groundwater for t_k time steps starting from time t. For LT-Φ, the contaminant injection is stopped for the initial 100 time steps in Scenarios A and B, expressed as $F_{i,j}^{(0,t_k)}(S^*)$ in Equation 8 for $t_k \in [1,2,\ldots,100]$. In Scenario C, the cessation is extended to 400 time steps, and $F_{i,j}^{(0,t_k)}(S^*)$ holds for $t_k \in [1,2,\ldots,400]$.

In Scenario B-M2, we demonstrate three affected locations (O1, O2 and O3) that are close to the contaminant sources (Figure 10g), that is, O1 is closest to W1, O2 is closest to W2, and O3 is closest to W3. The short-term effect represents the relatively fast impact, and we evaluate each location point that receives the impact exerted by all contaminant sources within 20 days at 50 different time points. The long-term effects on the three observation wells are examined within the first 100 days.

Figures 10a–10f demonstrates the influences of the three injection sources in B-M2. The long-term effects of all the observation wells are dominantly affected by the closest contaminant source (in Figures 10a–10c). As the groundwater table and the hydraulic gradient are important in plume mobility, O1 and O3 are quickly affected by

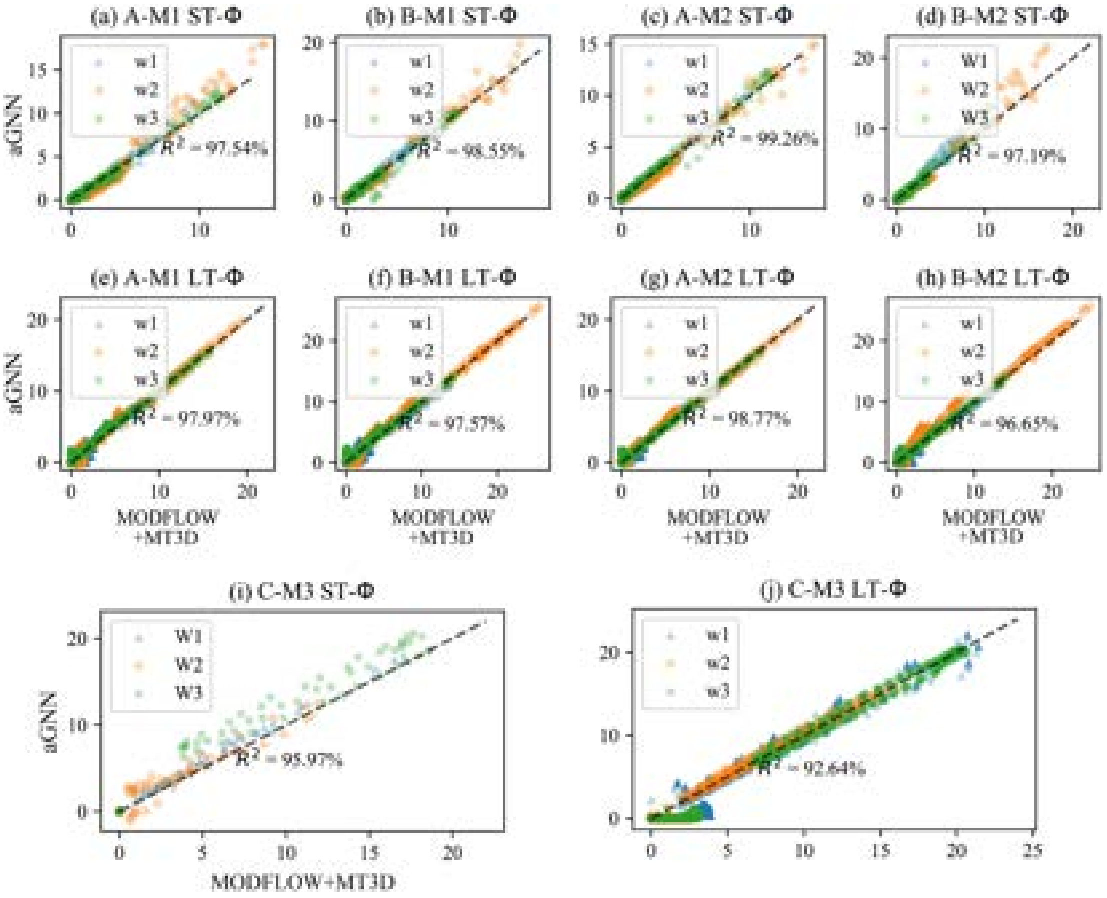

Figure 11. Comparison of both ST-Φ and LT-Φ of three contaminant sources (i.e., W1, W2, and W3) using SHAP based on results from aGNN and ground-truths generated from MODFLOW and MT3DMS in five cases.

pollution because they are located downstream of their closest pollution sources (i.e., W1 and W3) and have large positive hydraulic gradients (Figure 10g). In contrast, the hydraulic gradient from W2 to O2 is negative, indicating that the effect of W2 on O2 is dominated by dispersion rather than advection. Even though W1 is located farther from O2 than W2, O2 is also affected by W1 due to the positive hydraulic gradient. The effect as LT-Φ of $W1 \rightarrow O2$ is of smaller magnitude and has a longer elapsed time than that of $W2 \rightarrow O2$. In addition, LT-Φ of $W1 \rightarrow O1$, $W2 \rightarrow O2$, and $W3 \rightarrow O3$ do not grow as rapidly as the cumulative mass flow (CMF) in Figures 10a–10c, representing the time lagged effect of contaminant release, especially at the locations that are affected by dispersion (O2).

The short-term effects also exhibit similar relative magnitudes and the speeds of effect growth. Figures 10d–10f indicates that only $W1 \rightarrow O1$, $W2 \rightarrow O2$, and $W3 \rightarrow O3$ are significant in the short-term analysis, and the average effect size (ST-Φ) of $W2 \rightarrow O2$ is one order of magnitude smaller than those of $W1 \rightarrow O1$ and $W3 \rightarrow O3$. In addition, the variations in ST-Φ at each observation are largely due to the release plan of the nearest contaminant source. As W3 starts pollution discharge behind the other two wells and its CMF increases least frequently among all wells (Figure 10c), its influence accumulates at varying speeds depending on different examined time points; thus, the time lagged influences on O3 from W3 vary over a wide range. For the CMF of *W*1, its high frequency and large amount enable $W1 \rightarrow O1$ to have relatively small variations but a large impact (as the median line). Overall, the temporal patterns interpreted by ST-Φ are consistent with the physical understanding, indicating that our aGNN models learn the correct knowledge from the observations.

The ground-truth Φ can be computed based on physics-based models (i.e., MODFLOW and MT3DMS), and the effects on all the observations over the area can be compared to estimate the accuracy of interpretation by using aGNN. Figures 11a–11h demonstrate the comparisons of ST-Φ as a short-term effect (20 days) and LT-Φ as a

Table 3
Summary of Computational Time and Speed up

	Short term (300 scenarios)					Long term (6 scenarios)				
	A-M1	A-M2	B-M1	B-M2	C-M3	A-M1	A-M2	B-M1	B-M2	C-M3
$T_{MODFLOW+MT3DMS}$ (s)	4,933		4,961		31,616	240		247		5,977
T_{aGNN} (s)	115	143	115	143	107	68	89	68	89	242
Speed-up	43.5	34.5	43.5	34.5	295.6	3.6	2.7	3.7	2.8	24.7

long-term effect (cumulative 100 days) of all monitor wells onsite in Scenario A and B. Of all three contaminant sources, W2 induces the largest effect (i.e., with the largest ST-Φ and LT-Φ) because it releases contaminants at a rate larger than the other two wells (Figure 5a). In particular, in B-M1 and B-M2, W2 exceeds the other two sources to a large extent concerning the long-term effect. This is mainly because W2 is located in an area with low hydraulic conductivity (Figures 5d and 5f), so the convective flux is slow from source W2, causing the concentration level at W2 to accumulate to a high value. The long-term effects are larger than the short-term effect, as expected, but the effects do not grow proportionally to time since the effects at the injection sources (where a large effect occurs) increase rapidly and level off (Figure 10h).

Figures 11i and 11j depict the short-term (ST-Φ) and long-term (LT-Φ) effects (cumulative 400 days) of all monitoring wells in Scenario C. Due to the sparsely distributed monitoring networks, ST-Φ values above 0 occur only in the injection wells, while other locations remain unaffected. Therefore, ST-Φ represents the effect of injection wells only at their respective locations. Additionally, LT-Φ calculates the effect over a much longer period (400 days) in the future, posing a challenging task for accurate prediction. At some locations where the actual long-term effects are small, aGNN underestimates the effect size. However, 92% of R^2 values indicate that aGNN is capable of capturing the overall long-term effect.

Overall, Φ computed based on aGNN accurately represents the phenomena that are described by physics-based models. In terms of interpretation accuracy, we obtain an R^2 greater than 92% in depicting both long-term effects and long-term effects in all four cases, indicating that the SHAP method applied to aGNN is highly effective in interpreting the influence of the multi-point contaminant sources.

We further analyze the computational efficiency of using aGNN for forward inference in the source attribution of contaminant sources based on wall-clock time analysis. In SHAP method, the three sources form six combinations of contaminant discharging scenarios (i.e., {W1, W2, W3, W1 + W2, W2 + W3, W1 + W3}). However, the numbers of scenario evaluations are different for computing ST-Φ and LT-Φ. For ST-Φ, the integrated MODFLOW and MT3DMS models and aGNN separately evaluate the contamination flow under scenarios in which various contaminant sources stop release at 50 different time points; thus, 300 scenarios need to be evaluated overall to analyze the short-term effect. For long-term effects, the integrated MODFLOW and MT3DMS generate predictions for the first 100 days in Scenarios A and B, and aGNN computes five consecutive 20-day predictions to obtain a prediction of 100 days in length. In Scenario C, 400 days are predicted by integrated MODFLOW and MT3DMS, and aGNN generates 20 consecutive 20-day predictions. The speed-up measures the ratio of computational time required in the physics-based model to that in aGNN (i.e., $T_{MODFLOW+MT3DMS}/T_{aGNN}$). Table 3 illustrates that aGNN reduces the computational time required in physics-based models by more than 60% for long-term effects (i.e., speed-up > 2.5 = 1/(1–0.6)) and by 97% in short-term effects analysis (speed-up > 33.3 = 1/(1%–97%)). This is mainly because both MODFLOW and M3TD require lengthy file preparation and model simulation processes, and are serial computed by CPU. In contrast, aGNN can efficiently handle variable assignments and multi-step predictions in parallel using GPU, resulting in faster processing speed, especially when a large number of model simulations (i.e., short-term effects in multiple trials) are needed. The speed-up is further increased in case M1, which involves fewer observations. The computational budget of aGNN is smaller than that in case M2, while MODFLOW and MT3DMS are the same for both cases. In C-M3, the computation time for physics-based models significantly increases due to the expansive study area. In contrast, the time required by aGNN decreases as fewer monitoring nodes are involved, resulting in a speed-up reaching as high as 295 for short-term analysis and approximately 25 for long-term analysis. In summary, aGNN is highly promising as a surrogate of MODFLOW and MT3DMS and can largely reduce the computational burden for analyzing the influence of each contaminant release activity.

5. Conclusions

Understanding the state and dynamics of groundwater systems in response to anthropogenic pollution activities is essential for predicting environmental impacts and for developing appropriate management strategies. Given that groundwater flow and contaminant transport processes are not readily and fully observable in the subsurface and data on groundwater level and pollution in the entire study area can be difficult and expensive to obtain, the modeling task is complex, and the predictive work is difficult (Lall et al., 2020; Zheng & Bennett, 2002). This study seeks to develop a novel data-driven model, aGNN, for modeling contaminant transport in heterogeneous groundwater aquifers, with an emphasis on situations that involve limited and unevenly distributed spatial monitoring data.

In this study, we demonstrate that GNN-based models are well suited for the challenges outlined above. The node and edge, two important features in graph structure data, enable information transfer in graph networks, which is comparable to physics-based groundwater flow and solute transport movement and can be generalized to water quality monitoring networks in any irregular structure. The main contribution of this study is that we incorporate three important building blocks in aGNN, that is, attention mechanism and spatiotemporal embedding layers and GCN layer that learns the highly nonlinear spatiotemporal dependencies in contaminant transport in response to anthropogenic sources. The three building blocks improve the spatiotemporal learning accuracy by adopting dynamic weight assignment, prior feature transformation and information transfer in the physics-based graph structure of contaminant transport. In our experiment, aGNN achieves an R^2 value as high as 99%, demonstrating the high level of prediction power of the integrated attention mechanism and embedding layers. Our results also illustrate the potential of using aGNN to deduce observations at unmonitored locations from the data provided in monitored locations through knowledge generalization via graph learning. Despite limited data, aGNN can effectively leverage available information to deduce spatiotemporal variations in contaminant movement, even in aquifers with a high degree of heterogeneity, or at large sites with limited monitoring wells. However, it is noteworthy that the amount of available data is essential in accurate reasoning because a greater volume of data leads to improved modeling accuracy, and can enhance the accuracy of transferability and inductive analysis. Further research, such as optimizing the monitoring network system, can be explored to enhance the efficiency of the monitoring system.

Furthermore, aGNN facilitates the analysis of contaminant source attribution. Utilizing the SHAP method, aGNN's interpretations connecting contaminant sources (drivers) to resultant concentrations (outcomes) align with the physical principles governing contaminant transport processes. The performance of aGNN demonstrates its high level of accuracy and efficiency as a surrogate for the numerical simulation models MODFLOW and MT3DMS in analyzing contaminant source attribution. This approach achieves an R^2 value exceeding 92% and significantly reduces the computational burden of physics-based models by 90%. The computational efficiency is particularly noticeable for a large site with fewer monitoring wells, achieving a speed-up of approximately 300 folds. This efficiency can be further amplified in scenarios that encompass a substantial number of injection wells and extended management periods, especially when employing GPU parallel processing. This approach is particularly relevant when undertaking extensive scenario analyses, such as assessing the vulnerability of anthropogenic pollution at a study site with numerous pollution sources.

As groundwater abstraction and anthropogenic pollution continue to increase at the global scale, the availability of clean groundwater is decreasing, affecting urbanization, industrialization, and human health. Thus, the demands for accurate and fast prediction of groundwater quality and for approaches to impartially managing pollution sources continue to increase. The explosion of data resources and monitoring techniques can produce a stronger backbone for a better understanding of coupled human and water resource systems. The proposed aGNN model can play a prominent role in developing groundwater management plans to support environmentally sustainable and socially equitable policies.

Appendix A: Benchmark Models: DCRNN

Diffusion convolutional recurrent neural network (DCRNN) is designed for spatiotemporal prediction (Li et al., 2017). Originally, it was applied to and evaluated on a univariate traffic speed forecasting problem. The main structure employs diffusion graph convolutional networks (DCNNs) and gated recurrent units (GRUs) for processing spatiotemporal graph data. Specifically, DCNN models the spatial dependency analogously to the diffusion process, and are characterized by random walks on the input graph (Atwood & Towsley, 2015). The weighted combination of neighborhoods is formed by the "diffusion process," in which large, well-connected node-communities in graphs are amplified and small-scale node-communities are suppressed. As a special architecture of GCN, DCNN learns the representation of graph structured data and captures the spatial dependency. GRU is a variant of RNN that is targeted to learn sequential data such as time series (Chung et al., 2014). Similar to LSTM, GRUs use gated units to control the information flow in sequence data, however, unlike LSTM, GRUs have no output gate, so fewer parameters need to be trained.

In this study, we modify DCRNN to be applicable to a multivariate study so that it takes multi-source inputs, including water table, contaminant transport and pollutant discharge schedules, while having outputs as multi-step predictions of contamination concentrations. We used DCRNN as our benchmark model with the following parameter settings: the vector dimension of RNN units is 32; the number of RNN layers is 2; and the max diffusion step is 2.

Appendix B: Benchmark Models: ConvLSTM

ConvLSTM is a model that combines both CNN and LSTM techniques in a sequence-to-sequence learning framework; therefore, it is suitable for high-dimensional spatiotemporal inputs (Shi et al., 2015). Unlike LSTM, ConvLSTM layer contains a convolutional operator in LSTM for both the input-to-state and state-to-state transitions so that it preserves not only the advantages of fully connected LSTM (FC-LSTM) in extracting temporal patterns but is also suitable for learning spatial dependencies. Previous studies have shown that ConvLSTM network can capture spatial and temporal correlations simultaneously and consistently outperforms FC-LSTM in applications such as video recognition, travel demand prediction, and precipitation forecasting (Oprea et al., 2020; Shi et al., 2015; Wang et al., 2018).

However, ConvLSTM is pertinent to image-like data and cannot be directly applied to irregular data, such as unevenly distributed observations. In this study, we use masked ConvLSTM, which contains an additional regional mask layer to filter the monitoring locations out of the regions. Thus, the areas with no observation are ignored. The mask filter layer operates before the ConvLSTM unit and is followed by two layers of ConvLSTM in both the encoder and the decoder. The output of the last layer in the decoder is input into a 3D convolutional layer (Conv3D) that transforms the feature representations into targets. The parameters in the ConvLSTM unit are the kernel size, the number of kernels, and the stride size, defined as 3×3, 10 and 1×1, respectively. Similarly, in the output layer as Conv3D, the kernel size is $1 \times 1 \times 10$, the stride size is $1 \times 1 \times 1$, and the number of kernels is 32.

Appendix C: The Design of Hyperparameters

To determine the hyperparameters in the benchmark models, we employed cross-validation strategy for parameter optimization across various configurations, by using k-fold cross-validation ($k = 5$) to split all the samples into a training set and a validation set. For each parameter combination, a model was trained on the training set and evaluated on the validation set. The final configuration was selected based on the parameter set yielding the lowest median RMSE across all possible configurations.

The specific parameters for ConvLSTM, include the following:

1. Output layer Conv3D kernels: $1 \times 1 \times 3$, $1 \times 1 \times 5$, $1 \times 1 \times 10$
2. Number of kernels in ConvLSTM layers: 4, 16, 32

For DCRNN, the parameters are as follows:

1. Hidden states: 16, 32, 64
2. Number of DCRNN layers: 1, 2

The learning rates for aGNN, DCRNN, and ConvLSTM were investigated at 0.01, 0.005, 0.003, and 0.001. From these values, we selected the learning rates that resulted in the best algorithm performance. Consequently, the learning rates chosen were 0.003, 0.003, and 0.005 for aGNN, DCRNN, and ConvLSTM, respectively.

Appendix D: Modeling Accuracy

Figure D1 illustrates that the spatial distribution of RAE exhibits higher values near pollution sources, particularly in the upper stream regions, compared to lower stream areas. This pattern is attributed to the narrow range of contaminant concentration. Even minor prediction deviations can lead to large RAE values. Notably, in scenario C-M3, a significantly larger site result in fewer contaminant impacts at many distant locations, thus yielding high RAEs in those areas. Overall, aGNNs exhibit lower RAE across the entire area compared to the other models.

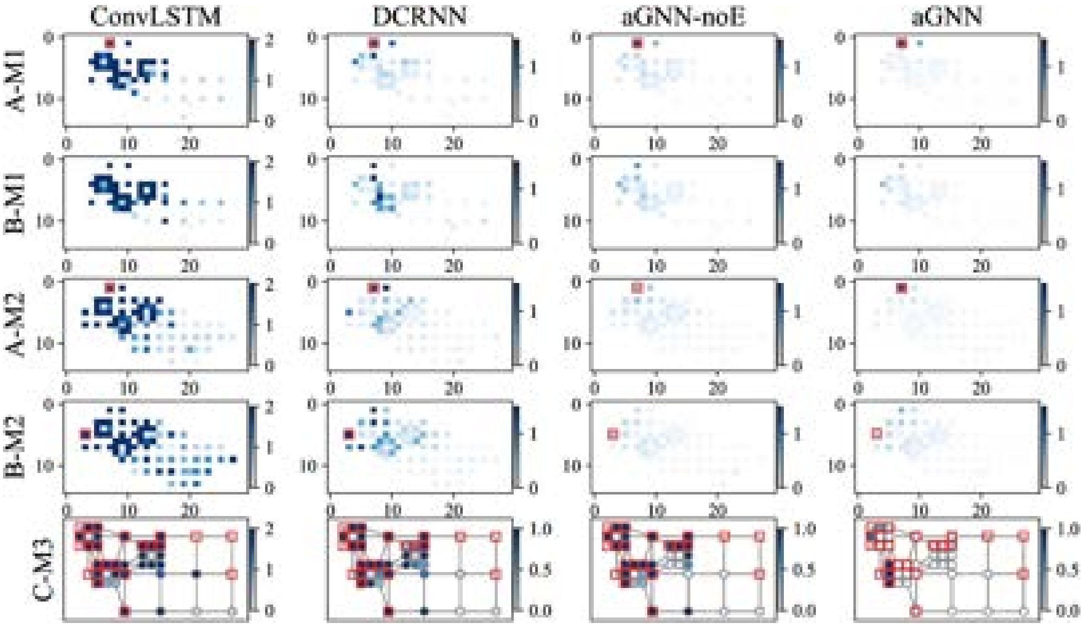

Figure D1. Spatial distribution of modeling accuracies, measured by Relative Absolute Error (RAE), across five cases (A-M1, A-M2, B-M1, B-M2, C-M3) using four algorithms. Red rectangles indicate areas where the true value range is smaller than 0.01.

Data Availability Statement

The data and codes of aGNN used in this study are available at Pang (2024).

Acknowledgments

This research is supported by the National Key Research and Development Program of China (No. 2022YFC3202300), the Fundamental Research Funds for the Central Universities (Nos. B240201159, B240201014), and the National Natural Science Foundation of China (No. 51909118). Additional support was provided by Guangdong Provincial Key Laboratory of Soil and Groundwater Pollution Control (No. 2023B1212060002).

References

Adel, H., & Schütze, H. (2016). Exploring different dimensions of attention for uncertainty detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Vol. 1, Long papers* (pp. 22–34).

Alzahrani, M. K., Shapoval, A., Chen, Z., & Rahman, S. S. (2023). Pore-GNN: A graph neural network-based framework for predicting flow properties of porous media from micro-CT images. *Adv. Geo-Energy Res.*, *10*(1), 39–55. https://doi.org/10.46690/ager.2023.10.05

Andrews, C. B., & Hennet, R. J.-C. (2022). Quest for groundwater quality sustainability – Lessons from 40 years of remediation in the United States. *Sustain. Horizons*, *2*, 100009. https://doi.org/10.1016/j.horiz.2022.100009

Atwood, J., & Towsley, D. (2015). Diffusion-convolutional neural networks. https://doi.org/10.48550/arXiv.1511.02136

Babaeian, E., Paheding, S., Siddique, N., Devabhaktuni, V. K., & Tuller, M. (2022). Short- and mid-term forecasts of actual evapotranspiration with deep learning. *Journal of Hydrology*, *612*, 128078. https://doi.org/10.1016/j.jhydrol.2022.128078

Babakhani, P., Bridge, J., Doong, R., & Phenrat, T. (2017). Parameterization and prediction of nanoparticle transport in porous media: A reanalysis using artificial neural network. *Water Resources Research*, *53*(6), 4564–4585. https://doi.org/10.1002/2016WR020358

Bai, T., & Tahmasebi, P. (2023). Graph neural network for groundwater level forecasting. *Journal of Hydrology*, *616*, 128792. https://doi.org/10.1016/j.jhydrol.2022.128792

Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., et al. (2018). Relational inductive biases, deep learning, and graph networks (pp. 1–40). https://doi.org/10.48550/arXiv.1806.01261

Belkin, M., & Niyogi, P. (2001). Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Proceedings of the 14th international conference on neural information processing systems: Natural and synthetic* (pp. 585–591).

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, *5*(2), 157–166. https://doi.org/10.1109/72.279181

Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., & Vandergheynst, P. (2017). Geometric deep learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, *34*(4), 18–42. https://doi.org/10.1109/MSP.2017.2693418

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 workshop on deep learning* (pp. 1–9).

Fan, W., Ma, Y., Li, Q., He, Y., Zhao, E., Tang, J., & Yin, D. (2019). Graph neural networks for social recommendation. In *The world wide web conference* (pp. 417–426). ACM. https://doi.org/10.1145/3308558.3313488

Feng, S., Li, X., Zeng, F., Hu, Z., Sun, Y., Wang, Z., & Duan, H. (2023). Spatiotemporal deep-learning model with graph convolutional network for well logs prediction. *IEEE Geoscience and Remote Sensing Letters*, *20*, 1–5. https://doi.org/10.1109/LGRS.2023.3317349

Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G., & Yacalis, G. (2018). Could machine learning break the convection parameterization deadlock? *Geophysical Research Letters*, *45*(11), 5742–5751. https://doi.org/10.1029/2018GL078202

Gorelick, S. M., & Zheng, C. (2015). Global change and the groundwater management challenge. *Water Resources Research*, *51*(5), 3031–3051. https://doi.org/10.1002/2014WR016825

Guo, S., Lin, Y., Wan, H., Li, X., & Cong, G. (2021). Learning dynamics and heterogeneity of spatial-temporal graph data for traffic forecasting. *IEEE Transactions on Knowledge and Data Engineering*, *XX*(11), 1–5428. https://doi.org/10.1109/TKDE.2021.3056502

Hakim, W. L., Rezaie, F., Nur, A. S., Panahi, M., Khosravi, K., Lee, C.-W., & Lee, S. (2022). Convolutional neural network (CNN) with metaheuristic optimization algorithms for landslide susceptibility mapping in Icheon, South Korea. *Journal of Environmental Management*, *305*, 114367. https://doi.org/10.1016/j.jenvman.2021.114367

Harbaugh, A. W. (2005). *MODFLOW-2005: The U. S. Geological survey modular ground-water model—the ground-water flow process* (pp. 6–A16). USGS Techniques and Methods. https://doi.org/10.3133/tm6A16

He, T., Wang, N., & Zhang, D. (2021). Theory-guided full convolutional neural network: An efficient surrogate model for inverse problems in subsurface contaminant transport. *Advances in Water Resources*, *157*, 104051. https://doi.org/10.1016/j.advwatres.2021.104051

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Huang, G., Liu, Z., van der Maaten, L., & Weinberger, K. Q. (2016). Densely connected convolutional networks. In *Proceedings of the IEEE on computer vision and pattern recognition*.

Huang, R., Ma, C., Ma, J., Huangfu, X., & He, Q. (2021). Machine learning in natural and engineered water systems. *Water Research*, *205*, 117666. https://doi.org/10.1016/j.watres.2021.117666

Jing, H., He, X., Tian, Y., Lancia, M., Cao, G., Crivellari, A., et al. (2023). Comparison and interpretation of data-driven models for simulating site-specific human-impacted groundwater dynamics in the North China Plain. *Journal of Hydrology*, *616*, 128751. https://doi.org/10.1016/j.jhydrol.2022.128751

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks (pp. 1–14). arXiv preprint arXiv:1609.02907.

Kitanidis, P. K. (2015). Persistent questions of heterogeneity, uncertainty, and scale in subsurface flow and transport. *Water Resources Research*, *51*(8), 5888–5904. https://doi.org/10.1002/2015WR017639

Klemmer, K., Safir, N., & Neill, D. B. (2022). Positional encoder graph neural networks for geographic data. In *KDD 2022 28th ACM SIGKDD conf. knowl*. Discov. Data Min.

Kratzert, F., Klotz, D., Shalev, G., Klambauer, G., Hochreiter, S., & Nearing, G. (2019). Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets. *Hydrology and Earth System Sciences*, *23*(12), 5089–5110. https://doi.org/10.5194/hess-23-5089-2019

Lall, U., Josset, L., & Russo, T. (2020). A snapshot of the world's groundwater challenges. *Annual Review of Environment and Resources*, *45*(1), 171–194. https://doi.org/10.1146/annurev-environ-102017-025800

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, *1*(4), 541–551. https://doi.org/10.1162/neco.1989.1.4.541

Li, R., Yuan, X., Radfar, M., Marendy, P., Ni, W., O'Brien, T. J., & Casillas-Espinosa, P. (2021). Graph signal processing, graph neural network and graph learning on biological data: A systematic review. *IEEE Reviews in Biomedical Engineering*, *16*, 109–135. https://doi.org/10.1109/RBME.2021.3122522

Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2017). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *6th Int. Conf. Learn. Represent. ICLR 2018 - Conf. Track Proc* (pp. 1–16).

Lundberg, S., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems* (pp. 4768–4777). https://doi.org/10.5555/3295222.3295230

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., et al. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, *2*(1), 56–67. https://doi.org/10.1038/s42256-019-0138-9

Luong, M.-T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025.

Markstrom, S. L., Niswonger, R. G., Regan, R. S., Prudic, D. E., & Barlow, P. M. (2005). *GSFLOW — Coupled ground-water and surface-water flow model based on the integration of the precipitation-runoff modeling system (PRMS) and the modular ground-water flow model (MODFLOW-2005)*. U.S. Geological Survey Techniques and Methods 6-D1. https://doi.org/10.3133/tm6D1

Mo, S., Zhu, Y., Zabaras, N., Shi, X., & Wu, J. (2019). Deep convolutional encoder-decoder networks for uncertainty quantification of dynamic multiphase flow in heterogeneous media. *Water Resources Research*, *55*(1), 703–728. https://doi.org/10.1029/2018WR023528

Mohammed, A., & Corzo, G. (2024). Spatiotemporal convolutional long short-term memory for regional streamflow predictions. *Journal of Environmental Management*, *350*, 119585. https://doi.org/10.1016/j.jenvman.2023.119585

Mugunthan, P., Shoemaker, C. A., & Regis, R. G. (2005). Comparison of function approximation, heuristic, and derivative-based methods for automatic calibration of computationally expensive groundwater bioremediation models. *Water Resources Research*, *41*(11), 1–17. https://doi.org/10.1029/2005WR004134

Najah Ahmed, A., Binti Othman, F., Abdulmohsin Afan, H., Khaleel Ibrahim, R., Ming Fai, C., Shabbir Hossain, M., et al. (2019). Machine learning methods for better water quality prediction. *Journal of Hydrology*, *578*, 124084. https://doi.org/10.1016/j.jhydrol.2019.124084

Ni, L., Wang, D., Singh, V. P., Wu, J., Wang, Y., Tao, Y., & Zhang, J. (2020). Streamflow and rainfall forecasting by two long short-term memory-based models. *Journal of Hydrology*, *583*, 124296. https://doi.org/10.1016/j.jhydrol.2019.124296

Oprea, S., Martinez-Gonzalez, P., Garcia-Garcia, A., Castro-Vargas, J. A., Orts-Escolano, S., Garcia-Rodriguez, J., & Argyros, A. (2020). A review on deep learning techniques for video prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*(6), 2806–2826. https://doi.org/10.1109/TPAMI.2020.3045007

Pang, M. (2024). aGNN: Attention-based graph neural network for groundwater contaminant transport: Feb 13, 2024 release (version 2) [Software]. *Zenodo*. https://doi.org/10.5281/zenodo.10652555

Pang, M., Du, E., & Zheng, C. (2023). A data-driven approach to exploring the causal relationships between distributed pumping activities and aquifer drawdown. *Science of the Total Environment*, *870*, 161998. https://doi.org/10.1016/j.scitotenv.2023.161998

Pang, M., & Shoemaker, C. A. (2023). Comparison of parallel optimization algorithms on computationally expensive groundwater remediation designs. *Science of the Total Environment*, *857*, 159544. https://doi.org/10.1016/j.scitotenv.2022.159544

Pang, M., Shoemaker, C. A., & Bindel, D. (2022). Early termination strategies with asynchronous parallel optimization in application to automatic calibration of groundwater PDE models. *Environmental Modelling & Software*, *147*, 105237. https://doi.org/10.1016/j.envsoft.2021.105237

Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). DeepWalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 701–710). ACM. https://doi.org/10.1145/2623330.2623732

Pietrzak, D. (2021). Modeling migration of organic pollutants in groundwater — Review of available software. *Environmental Modelling & Software*, *144*, 105145. https://doi.org/10.1016/j.envsoft.2021.105145

Razavi, S., Tolson, B. A., & Burn, D. H. (2012). Review of surrogate modeling in water resources. *Water Resources Research*, *48*(7), W07401. https://doi.org/10.1029/2011WR011527

Refsgaard, J. C., Storm, B., & Clausen, T. (2010). Système Hydrologique Europeén (SHE): Review and perspectives after 30 years development in distributed physically-based hydrological modelling. *Hydrology Research*, *41*(5), 355–377. https://doi.org/10.2166/nh.2010.009

Reichstein, M., Camps-valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., & Prabhat (2019). Deep learning and process understanding for data-driven Earth system science. *Nature*, *566*(7743), 196–204. https://doi.org/10.1038/s41586-019-0912-1

Sajedi-Hosseini, F., Malekian, A., Choubin, B., Rahmati, O., Cipullo, S., Coulon, F., & Pradhan, B. (2018). A novel machine learning-based approach for the risk assessment of nitrate groundwater contamination. *Science of the Total Environment*, *644*, 954–962. https://doi.org/10.1016/j.scitotenv.2018.07.054

Shapley, L. S. (1953). A value for n-person games. In H. W. T. Kuhn, & W. Albert (Eds.), *Contribution to the theory of games* (pp. 307–318). *Annals of mathematics studies. 28*. Princeton University Press.

Shen, C. (2018). A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resources Research*, *54*(11), 8558–8593. https://doi.org/10.1029/2018WR022643

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W., & Woo, W. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *Adv. neural inf. process. syst. 2015-January* (pp. 802–810).

Soriano, M. A., Siegel, H. G., Johnson, N. P., Gutchess, K. M., Xiong, B., Li, Y., et al. (2021). Assessment of groundwater well vulnerability to contamination through physics-informed machine learning. *Environmental Research Letters*, *16*(8), 084013. https://doi.org/10.1088/1748-9326/ac10e0

Sun, A. Y., Jiang, P., Mudunuru, M. K., & Chen, X. (2021). Explore spatio-temporal learning of large sample hydrology using graph neural networks. *Water Resources Research*, *57*(12), e2021WR030394. https://doi.org/10.1029/2021WR030394

Sun, A. Y., & Scanlon, B. R. (2019). How can big data and machine learning benefit environment and water management: A survey of methods, applications, and future directions. *Environmental Research Letters*, *14*(7), 073001. https://doi.org/10.1088/1748-9326/ab1b7d

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*, *4*, 3104–3112. https://doi.org/10.5555/2969033.2969173

Tong, X., Mohapatra, S., Zhang, J., Tran, N. H., You, L., He, Y., & Gin, K. Y.-H. (2022). Source, fate, transport and modelling of selected emerging contaminants in the aquatic environment: Current status and future perspectives. *Water Research*, *217*, 118418. https://doi.org/10.1016/j.watres.2022.118418

United Nations WATER. (2018). *Groundwater overview: Making the invisible visible*. Delft, Neth Int. Groundw. Resourc. Assess. Cent.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. In *Adv. neural inf. process. syst. 2017-December* (pp. 5999–6009).

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2017). Graph attention networks.

Wang, D., Yang, Y., & Ning, S. (2018). DeepSTCL: A deep spatio-temporal ConvLSTM for travel demand prediction. In *Proc. Int. Jt. Conf. Neural networks 2018-July, 1–8*. https://doi.org/10.1109/IJCNN.2018.8489530

Wu, S., Sun, F., Zhang, W., Xie, X., & Cui, B. (2022). Graph neural networks in recommender systems: A survey. *ACM Computing Surveys*, *55*(5), 1–37. https://doi.org/10.1145/3535101

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2021). A comprehensive survey on graph neural networks. *IEEE Trans. Neural Networks Learn. Syst.*, *32*(1), 4–24. https://doi.org/10.1109/TNNLS.2020.2978386

Wunsch, A., Liesch, T., & Broda, S. (2021). Groundwater level forecasting with artificial neural networks: A comparison of long short-term memory (LSTM), convolutional neural networks (CNNs), and non-linear autoregressive networks with exogenous input (NARX). *Hydrology and Earth System Sciences*, *25*(3), 1671–1687. https://doi.org/10.5194/hess-25-1671-2021

Xia, W., & Shoemaker, C. A. (2021). Improving the speed of global parallel optimization on PDE models with processor affinity scheduling. *Computer-Aided Civil and Infrastructure Engineering*, *37*(3), 1–21. https://doi.org/10.1111/mice.12737

Xiong, R., Zheng, Y., Chen, N., Tian, Q., Liu, W., Han, F., et al. (2022). Predicting dynamic riverine nitrogen export in unmonitored watersheds: Leveraging insights of AI from data-rich regions. *Environmental Science & Technology*, *56*(14), 10530–10542. https://doi.org/10.1021/acs.est.2c02232

Yu, J., Tian, Y., Jing, H., Sun, T., Wang, X., Andrews, C. B., & Zheng, C. (2023). Predicting regional wastewater treatment plant discharges using machine learning and population migration big data. *ACS ES&T Water*, *3*(5), 1314–1328. https://doi.org/10.1021/acsestwater.2c00639

Yu, J., Tian, Y., Wang, X., & Zheng, C. (2021). Using machine learning to reveal spatiotemporal complexity and driving forces of water quality changes in Hong Kong marine water. *Journal of Hydrology*, *603*, 126841. https://doi.org/10.1016/j.jhydrol.2021.126841

Zheng, C., & Bennett, G. D. (2002). *Applied contaminant transport modeling*. Wiley-Interscience.

Zheng, C., & Wang, P. P. (1999). MT3DMS: A modular three-dimensional multispecies transport model for simulation of advection, dispersion and chemical reactions of contaminants in groundwater systems. In *Strategic environmental research and development program*.

Zhong, Z., Sun, A. Y., Yang, Q., & Ouyang, Q. (2019). A deep learning approach to anomaly detection in geological carbon sequestration sites using pressure measurements. *Journal of Hydrology*, *573*, 885–894. https://doi.org/10.1016/j.jhydrol.2019.04.015

Zhou, H., Ren, D., Xia, H., Fan, M., Yang, X., & Huang, H. (2021). AST-GNN: An attention-based spatio-temporal graph neural network for Interaction-aware pedestrian trajectory prediction. *Neurocomputing*, *445*, 298–308. https://doi.org/10.1016/j.neucom.2021.03.024