

**Document Version**

Final published version

**Licence**

CC BY

**Citation (APA)**

Acosta, C. M., Herath, H. M. V. V., Saha, A., Rasnayaka, S., & Marshall, L. (2026). Exploring graph neural networks for flood modeling: Challenges, opportunities, and future prospects. *Environmental Modelling and Software*, 205, Article 107132. <https://doi.org/10.1016/j.envsoft.2026.107132>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

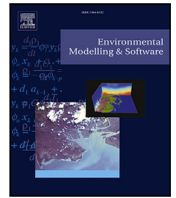
In case the licence states "Dutch Copyright Act (Article 25fa)", this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.  
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

**Sharing and reuse**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.



# Exploring graph neural networks for flood modeling: Challenges, opportunities, and future prospects

Carlo Malapad Acosta<sup>a</sup>, Herath Mudiyanse Viraj Vidura Herath<sup>b</sup>, Abhishek Saha<sup>c,d</sup>,  
Sanka Rasnayaka<sup>a</sup>, Lucy Marshall<sup>b,\*</sup>

<sup>a</sup> Department of Computer Science, School of Computing, National University of Singapore, Singapore

<sup>b</sup> School of Civil Engineering, Faculty of Engineering, The University of Sydney, Sydney, New South Wales, Australia

<sup>c</sup> Delft Institute of Applied Mathematics, Delft University of Technology, Delft, The Netherlands

<sup>d</sup> Hydroinformatics Institute, Singapore

## ARTICLE INFO

### Keywords:

Flood modeling  
Surrogate models  
Machine learning  
Deep learning  
Graph neural networks

## ABSTRACT

Hydrodynamic flood models play a central role in understanding the spatiotemporal evolution of floods for effective disaster management. Current implementations are limited, as physics-based numerical models are constrained by slow inference times, and data-driven models lack interpretability and physical consistency.

Graph Neural Networks (GNNs) provide a physically meaningful framework to represent unstructured spatial domains as graphs, mirroring the node-link structure of numerical models. Unlike other deep learning methods, GNNs require no complex spatial data transformation and can seamlessly integrate physics-based priors.

This review synthesizes forty nine peer-reviewed studies on GNN applications in flood modeling, identifying key gaps in physics incorporation, scalability, and cross-domain generalization. Drawing on advances from related fields where GNNs have excelled, we outline strategies to overcome these limitations. GNNs hold significant potential as next-generation flood models by combining physical realism, computational efficiency, and adaptability for real-time flood prediction and risk management.

## Contents

|        |                                               |    |
|--------|-----------------------------------------------|----|
| 1.     | Introduction .....                            | 2  |
| 2.     | Background .....                              | 3  |
| 2.1.   | Flood modeling .....                          | 3  |
| 2.1.1. | Problem statement .....                       | 3  |
| 2.1.2. | Modeling methods .....                        | 3  |
| 2.2.   | Graph neural networks .....                   | 5  |
| 2.2.1. | Graphs .....                                  | 5  |
| 2.2.2. | Message passing .....                         | 5  |
| 2.2.3. | Common GNN architectures .....                | 6  |
| 2.2.4. | Physics-informed GNNs .....                   | 6  |
| 3.     | Review of current approaches .....            | 6  |
| 3.1.   | Contextualizing GNNs for flood modeling ..... | 6  |
| 3.2.   | Methodology .....                             | 7  |
| 3.3.   | General findings .....                        | 7  |
| 3.4.   | Point-based modeling .....                    | 8  |
| 3.4.1. | River networks .....                          | 8  |
| 3.4.2. | Coastal regions .....                         | 9  |
| 3.4.3. | Urban regions .....                           | 10 |
| 3.4.4. | Storm water drainage systems .....            | 10 |

\* Corresponding author.

E-mail addresses: [e1391136@u.nus.edu](mailto:e1391136@u.nus.edu) (C.M. Acosta), [viraj.herath@sydney.edu.au](mailto:viraj.herath@sydney.edu.au) (H.M.V.V. Herath), [a.saha@tudelft.nl](mailto:a.saha@tudelft.nl) (A. Saha), [sanka@nus.edu.sg](mailto:sanka@nus.edu.sg) (S. Rasnayaka), [lucy.marshall@sydney.edu.au](mailto:lucy.marshall@sydney.edu.au) (L. Marshall).

<https://doi.org/10.1016/j.envsoft.2026.107132>

Received 28 May 2026; Received in revised form 25 July 2026; Accepted 28 July 2026

Available online 4 August 2026

1364-8152/© 2026 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

|        |                                                                                        |    |
|--------|----------------------------------------------------------------------------------------|----|
| 3.5.   | Particle-based modeling .....                                                          | 10 |
| 3.6.   | Mesh-based modeling .....                                                              | 10 |
| 4.     | Analysis of GNNs-based flood models .....                                              | 12 |
| 4.1.   | Advantages and strengths .....                                                         | 12 |
| 4.1.1. | Representational power .....                                                           | 12 |
| 4.1.2. | Predictive performance .....                                                           | 14 |
| 4.1.3. | Computational efficiency .....                                                         | 14 |
| 4.1.4. | Physical grounding .....                                                               | 14 |
| 4.2.   | Research gaps and limitations .....                                                    | 14 |
| 4.2.1. | Hydraulic and hydrodynamic consistency .....                                           | 14 |
| 4.2.2. | Modeling scope .....                                                                   | 15 |
| 4.2.3. | Network representation .....                                                           | 15 |
| 4.2.4. | Predictive transparency .....                                                          | 15 |
| 4.2.5. | Evaluation rigor .....                                                                 | 16 |
| 4.3.   | Critical synthesis .....                                                               | 16 |
| 5.     | Opportunities from related domains .....                                               | 16 |
| 5.1.   | Physics incorporation .....                                                            | 16 |
| 5.2.   | Handling large graphs .....                                                            | 18 |
| 5.3.   | Simultaneous node & edge predictions .....                                             | 18 |
| 5.4.   | Inter-GNN benchmarks .....                                                             | 19 |
| 5.5.   | Generalization & transfer learning .....                                               | 19 |
| 5.6.   | Encode diverse node & edge features .....                                              | 20 |
| 5.7.   | Handling temporal data .....                                                           | 20 |
| 5.8.   | Handling sparse data .....                                                             | 20 |
| 6.     | Conclusions .....                                                                      | 21 |
|        | CReditT authorship contribution statement .....                                        | 21 |
|        | Declaration of Generative AI and AI-assisted technologies in the writing process ..... | 21 |
|        | Declaration of competing interest .....                                                | 21 |
|        | Acknowledgments .....                                                                  | 21 |
|        | Data availability .....                                                                | 21 |
|        | References .....                                                                       | 21 |

## 1. Introduction

Floods are one of the greatest threats to urban populations, causing major economic and structural damage estimated to be more than 3 trillion USD in the past 25 years (United Nations Office for Disaster Risk Reduction (UNDRR) and Centre for Research on the Epidemiology of Disasters (CRED), 2020). In addition, they pose a significant threat to human life with deaths reaching at least 8 million in the twentieth century (Jonkman and Kelman, 2005). This risk is amplified by the extent of its influence, with up to 23% of the world's population (around 1.81 billion people) being exposed to the adverse effects of floods (Rentschler et al., 2022). These numbers are only expected to increase with time, as climate change has increased the sudden occurrence of flooding disasters (Tabari, 2020). To mitigate these negative outcomes, it is necessary to implement a combination of risk assessment, early warning signals, and real-time emergency response, all of which require detailed inundation information. Flood modeling involves simulating flood behavior using computational approaches, either through physics-based numerical simulations or data-driven methods. These models account for environmental conditions (e.g., rainfall, topography, boundary conditions) and hydraulic infrastructure, to predict variables such as water depth, discharge, velocity, and volume as floods propagate. Accurate and reliable flood models provide critical insights for data-driven decision making, supporting effective disaster management and risk mitigation.

Computational flood modeling approaches initially began with numerical methods. They are numerical solvers of partial differential equations (PDEs) that describe flooding behavior over time given various parameters such as rainfall, boundary conditions, channel geometry, and riverbed roughness (Kumar et al., 2023). Despite their accuracy, numerical methods require extensive computational time and resources, which limit their practical use. Some studies have tried to improve performance through GPU-based acceleration techniques (Ming et al., 2020; Vacondio et al., 2014). However, these approaches

come with a high financial cost and can have varying performance depending on hardware specifications. Further simplifications of the underlying equations have also been explored, but were shown to be ineffective in more complex scenarios (Costabile et al., 2017).

Data-driven surrogate models offer a compelling alternative to traditional numerical simulations, providing substantially faster predictions while maintaining high accuracy. These approaches typically employ machine learning (ML) to forecast flood evolution by identifying patterns from large historical datasets (Kumar et al., 2023). As data collection in flood management becomes more mature, the availability of training data increases, further improving model performance. Deep learning (DL), a subset of ML, has gained prominence due to its ability to capture complex non-linear relationships and perform automatic feature extraction (LeCun et al., 2015), making it particularly well-suited for challenging domains such as natural language processing, computer vision, and flood modeling. However, data-driven models are subject to key limitations, including reliance on large volumes of data and weak physical consistency.

Convolutional neural networks (CNNs) have been widely adopted for image processing due to their spatial inductive bias, enabling them to model complex large-scale domains with relatively limited training data. Many research studies have leveraged CNNs to process and predict image-based flood maps (Herath et al., 2025; Hosseiny, 2021; Löwe et al., 2021; Y. Wang et al., 2020). However, CNNs require data to be represented in a structured Euclidean grid, which often does not reflect real-world flood scenarios characterized by irregular land elevations and non-uniform fluid flow.

Graph neural networks (GNNs) were developed to overcome this limitation, as they can directly process unstructured data in the form of graphs without the need for data transformation. They utilize a generalized form of CNN-based convolution to non-Euclidean space to retain the same inductive biases that improve data efficiency (Battaglia et al., 2018). However, this same flexibility entails managing oversmoothing node representations in deeper architectures and memory requirements that rapidly increase with graph size. In the context of flood modeling, a

target catchment can be represented in a flexible manner, enabling multiple representations that preserve dyadic relationships such as flow, slope or distance. Graph convolution then propagates latent representations following these priors, which mirror the physical propagation of water. Despite this structural alignment, GNN techniques remain largely unexplored in flood modeling. Meanwhile, distributed graph training, neighborhood sampling, transfer learning, and heterogeneous graph modeling are mature in traffic forecasting, power systems, molecular chemistry, and computational fluid dynamics, and each presents a direct roadmap for extending current GNN flood models.

This review addresses this gap by consolidating the current approaches and proposing these promising research directions. Several reviews have examined this field with a progressively narrower scope, from broad overviews of flood modeling approaches (Kumar et al., 2023), to ML for flood prediction (Mosavi et al., 2018), to DL specifically (Bentivoglio et al., 2022). However, GNNs in these works, if even mentioned, are only discussed at a high-level, without a deeper discussion on particulars like graph construction and modeling techniques. To the best of our knowledge, this is the first survey dedicated to GNNs for flood modeling applications.

The scientific contributions of this paper are the following:

- (a) A comprehensive review and taxonomy of 49 peer-reviewed studies employing GNNs for flood modeling.
- (b) A critical analysis of the strengths and limitations of current GNN approaches.
- (c) A collation of research opportunities for advancing GNN-based flood modeling, drawing inspiration from successful applications in other domains.

Section 2 presents the theoretical and historical background of both flood modeling and GNNs. Section 3 reviews the current research on GNN applications within the flood modeling domain. Section 4 synthesizes the key advantages and limitations of such GNNs. Finally, Section 5 explores methods developed in other fields that can be adapted to enhance GNN-based flood modeling.

## 2. Background

This section provides a detailed background of the two key concepts tackled in this review, flood modeling and GNNs. Section 2.1 defines the flood modeling problem and reviews current modeling approaches. Section 2.2 introduces GNNs through a discussion of its theoretical foundations.

### 2.1. Flood modeling

#### 2.1.1. Problem statement

Flood models simulate hydrodynamic and hydraulic processes to replicate flooding behavior in various settings including river networks, coastal lines, and urban regions (Bentivoglio et al., 2022). These models use historical hydrodynamic states (e.g., water depth, velocity), hydrological variables (e.g., rainfall, runoff), and catchment properties (e.g., elevation, area, slope, roughness) to predict future flood extent and intensity (Kumar et al., 2023).

The governing equations used in a flood model depend primarily on the granularity of representation required for a target catchment. For spatially-constrained high-resolution flood modeling, three-dimensional (3D) models use the incompressible Reynolds-averaged Navier Stokes (RANS) equations to understand complex local-scale phenomena (e.g., transient flood shorelines, vertical turbulence) (Teng et al., 2017). However, typical flood-prone areas include coastal waters, flood plains and river valleys, which are overland topographies characterized by greater horizontal breadth. For such cases, resolving vertical turbulence or mixing is not necessary because the dominant gradients occur in the horizontal direction. The two-dimensional (2D)

shallow water equations (SWE), derived by depth-averaging the RANS equations, are an acceptable trade-off between computational complexity and accurate physical representation in these scenarios (Costabile et al., 2017). Systems where flow is predominantly one-dimensional (1D), such as along open channels and pipes, can be described by the simpler Saint-Venant (S-V) equations (Teng et al., 2017). They can be interpreted as a cross-sectionally integrated form of the SWE, expressed in terms of wetted cross-sectional area and discharge, thereby greatly simplifying the problem domain. Coupled 1D-2D formulations can also be used, typically in urban systems, by using the 1D S-V equations for flow conveyance structures (Simon et al., 2019) and the 2D SWE for the overland flow (Sañudo et al., 2025). These are then coupled using flux-exchange schemes between the 1D and 2D domains. For example, conservative finite-volume coupling strategies compute flux exchanges between adjacent 1D and 2D control volumes while maintaining conservation of mass and momentum across the interface (Morales-Hernández et al., 2013). These different physical representations naturally lead to different graph abstractions. For example, river or catchment systems can be represented using gauges or subcatchments as nodes, drainage systems using junctions and pipes, and unstructured hydrodynamic meshes using cells and flux-exchange edges. These abstractions underpin the taxonomy of GNN-based flood models reviewed in this paper.

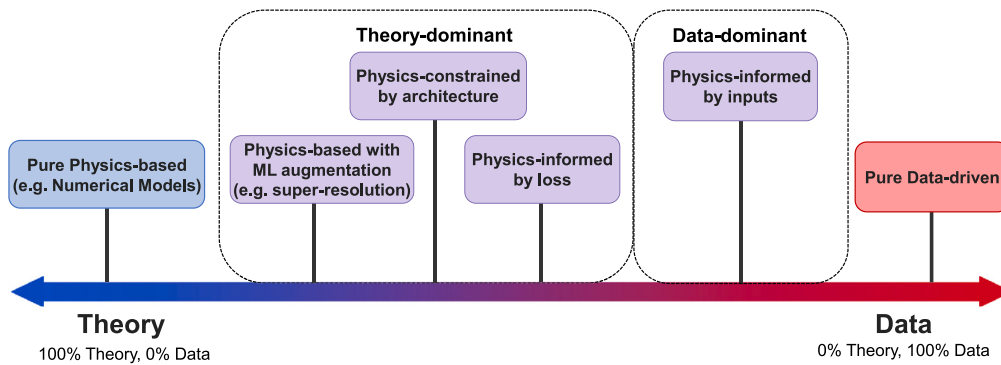
#### 2.1.2. Modeling methods

The two prevalent strategies in flood modeling are numerical methods that solve equations describing the underlying physics and data-driven methods that approximate spatiotemporal relationships within historical data. Data-driven methods can be further classified into traditional ML and DL. Between these two ends of the modeling spectrum, hybrid numerical-ML and physics-informed approaches seek to combine physical grounding with data-driven flexibility, as illustrated in Fig. 1. Physics-informed DL (PIDL) is a subset of DL for physics-based domains, where physical knowledge can be incorporated through input features, model architectures, or loss functions. A comprehensive discussion of each method is beyond the scope of this review. Instead, we briefly highlight the more widely used approaches.

Numerical models provide simplified mathematical representations of real-world flooding processes (Devia et al., 2015). They approximate intractable equations governing flood movement with discretization techniques like the finite difference, finite volume, and finite element schemes (Luo et al., 2022). These mesh-based approaches can employ structured or unstructured, regular or irregular grids (Kang and Kubatko, 2024), and are integrated over time using implicit, explicit, or hybrid semi-implicit/IMEX schemes. Well-known examples include HEC-RAS (Brunner, 1997), Delft3D-FM (Deltares, 2022), and LISFLOOD-FP (Shaw et al., 2021).

Although numerical models are highly accurate due to their direct derivation from physical laws, they require significant processing power (Kumar et al., 2023). Recent advances in numerical algorithms such as subgrid (Stelling, 2012) or dual-grid formulations (Sanders et al., 2025) have been developed, which solve volume conservation and friction forces in high resolution, and other terms in coarser resolutions (Saha et al., 2025). These have generated a substantial improvement in computational efficiency, especially when combined with parallel computations (Noh et al., 2018). Numerical models are also constrained by their reliance on high-fidelity input parameters, which are often unavailable given the absence of precise measurement instruments (Devkota et al., 2020; Wright, 2018). On top of this, each application typically demands extensive calibration and parameterization efforts before effective adoption (Ha et al., 2023).

Data-driven ML models act as surrogate models that learn flood behavior from historical data using statistical techniques. They offer a substantial increase in computational speed by reproducing the outputs of numerical models without requiring explicit knowledge of the underlying physical processes (Mosavi et al., 2018). Although unsupervised



**Fig. 1.** Gradient of approaches in PIDL. The spectrum ranges from purely theory-driven to purely data-driven methods. Physics-informed approaches lie in between these two extremes with theory-dominant methods that embed physics through the architecture or loss function, while data-dominant methods incorporate physical knowledge via inputs.

learning has been used to assess flood susceptibility (Landuyt et al., 2020; Refice et al., 2020), supervised learning is the dominant approach for spatiotemporal regression. Linear regression models (Rezaeianzadeh et al., 2014; Roland and Stuckey, 2008; Yariyan et al., 2020), tree-based models (Chen et al., 2019, 2020) and Support Vector Machines (Hong et al., 2018; Panahi et al., 2021) were all used to predict flood inundation. Despite their success, these conventional ML methods require manual feature engineering accompanied by tedious trial and error (Bentivoglio et al., 2022).

DL models excel at capturing non-linear relationships, making them particularly effective for complex domains such as flood modeling. The expressive power from deep computational layers enables automatic feature extraction, eliminating the need for manual feature engineering (LeCun et al., 2015). Multilayer perceptrons (MLPs), the simplest DL architecture, were the first applied to flood modeling for daily river outflow prediction (Rezaeianzadeh, 2013) and inundation mapping (Amini, 2010). However, their limited predictive power often necessitates integration with techniques such as autoencoder pretraining (Ahmadlou et al., 2021) or transformation functions (Xie et al., 2021). MLPs are also data-intensive due to the curse of dimensionality, which is pronounced in large-scale catchments with numerous hydrological and topographical features.

To improve data efficiency, subsequent architectures incorporated inductive biases tailored to the problem domain. Recurrent neural networks (RNNs) are designed to handle sequential data, enabling them to model temporal flood relationships effectively. Long Short-Term Memory (LSTM) networks demonstrated superior performance over MLPs in flood forecasting (Rahimzad et al., 2021), with further improvements achieved through data augmentation and batch normalization (Fang et al., 2021). On the other hand, CNNs introduce translation invariance and locality biases, making them suitable for learning spatial representations of flood maps. They were successfully applied in urban flood depth prediction (Guo et al., 2020) and multidimensional flood susceptibility mapping (Y. Wang et al., 2020). Hybrid models combining CNNs and RNNs have also been explored to jointly capture spatiotemporal dependencies, such as a Gated Recurrent Unit (GRU) with a CNN for real-time channel flood classification (Dong et al., 2020). Despite these advances, CNNs are constrained to Euclidean data, whereas many real-world catchments are better represented as unstructured spatial domains. Moreover, DL models in general are often criticized as “black boxes”, as their large parameter spaces make interpretability challenging (Devia et al., 2015).

Hybrid approaches that combine numerical and data-driven methods have also been explored, particularly for super-resolution. In these frameworks, a physics-based numerical model produces an initial coarse resolution prediction, which an ML model then downscales to a higher resolution. These hybrid models achieve higher accuracy

than purely data-driven approaches since the temporal dynamics are handled by the numerical model, grounding predictions in physically-derived priors and avoiding the error accumulation typical of autoregressive ML predictions. A model employing Gaussian Processes for the upskilling step has been shown to outperform various purely ML-based models including LSTMs and 1D CNNs (Fraehr et al., 2023, 2024). More recently, DL models including U-Nets (CNN variants that process data at multiple resolutions) (Herath et al., 2025) and latent diffusion models (Neo et al., 2026) have been applied to the same task. Nevertheless, hybrid approaches remain slower than pure ML surrogates, as they still depend on running a numerical model. Because these approaches still rely on coarse-resolution or low-fidelity numerical-model outputs, they are positioned closer to the theory-dominant side of the modeling spectrum, with the ML component primarily acting as an augmentation, correction, or super-resolution step.

In flood modeling, predictive accuracy alone is insufficient. Models must also be interpretable and physically consistent, as sparse real-world observations cause purely data-driven approaches to fit patterns that violate governing laws, producing artifacts such as water accumulating in dry cells or volume being created and destroyed between timesteps. PIDL models combine the strengths of data-driven and numerical methods by embedding physical constraints into DL architectures, enabling them to learn the underlying behavior of physical systems (Sharma et al., 2023). These can be developed through either physics-informed data, architecture or loss. Training with physically accurate data implicitly guides the model to predict values that follow its governing physical laws. Since numerical models generate data from physical equations, DL models trained on such data are implicitly exposed to physically consistent relationships (M. Kim et al., 2024; Qian et al., 2019; Yang et al., 2024). In flood modeling, this is the most widely adopted strategy, with synthetic training datasets commonly generated from hydrodynamic solvers. Physics-informed architectures are designed with model components tailored for physics applications. For instance, Fourier neural operators approximate PDE solutions via convolution-like operations in Fourier space (Li et al., 2021). These have demonstrated efficacy in urban inundation mapping (Sun et al., 2023) and coastal flood simulations (Jiang et al., 2021). Finally, physics-informed loss functions incorporate governing principles directly into the optimization objective, promoting physically consistent predictions. This strategy underpins physics-informed neural networks (PINNs) that approximate a higher-order solution to a PDE whose derivatives are obtained during backpropagation through a neural network (Raissi et al., 2019). Physics-informed loss has been directly applied in flood modeling such as representing the continuity equation as an inequality relating water flow between previous and future timesteps (Donnelly et al., 2024). The physical grounding of PINNs comes at the cost of increased complexity, as losses involving PDE residuals may be stiff and multi-objective weighting is non-trivial.

**Table 1**  
Comparison of different flood modeling approaches.

| Method                  | Advantages                                                                                               | Disadvantages                                                                                                                   |
|-------------------------|----------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------|
| Numerical methods       | High accuracy<br>Based on physical laws<br>Interpretable                                                 | High computational overhead                                                                                                     |
| Conventional ML methods | Enhanced processing speed                                                                                | Manual feature engineering<br>Purely statistical understanding<br>Can only handle low-dimensional spaces                        |
| MLPs                    | Automatic feature selection<br>Simple architecture                                                       | Lack of interpretability<br>No inductive bias                                                                                   |
| RNNs                    | Handles sequential data well<br>Can capture temporal information                                         | No bias for spatial data                                                                                                        |
| CNNs                    | Translation invariance and locality bias<br>Can capture spatial information<br>Better data efficiency    | No bias for temporal data<br>Can only handle Euclidean space                                                                    |
| Hybrid numerical and ML | Physical grounding<br>Simpler target learning task                                                       | Inference speed bottleneck                                                                                                      |
| Physics-informed        | Interpretable ML<br>Improved generalizability                                                            | Higher training time<br>Model/loss complexity                                                                                   |
| GNNs                    | Can handle unstructured spatial data<br>Flexibility in physics integration<br>Increased generalizability | Message passing not optimized for hardware<br>Oversmoothing when increasing layers<br>Difficulty capturing global relationships |

Within the theory–data spectrum illustrated in Fig. 1, GNNs should be viewed as flexible graph-based DL architectures rather than a single modeling category. Depending on the design, they can operate as purely data-driven surrogates, data-dominant models informed through hydrologic or hydraulic input features, theory-dominant models constrained through physics-aware architectures or loss functions, or hybrid numerical-ML augmentation and correction operators. Table 1 summarizes the advantages and limitations of these approaches. In subsequent sections, GNN-based methods will be discussed, highlighting their unique advantages relative to these approaches.

## 2.2. Graph neural networks

### 2.2.1. Graphs

Formally, a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  is defined as a data structure made up of a set of nodes  $\mathcal{V} = \{v_1, v_2, \dots, v_{|\mathcal{V}|}\}$  and edges  $\mathcal{E} = \{e_1, e_2, \dots, e_{|\mathcal{E}|}\}$ . Edges denote the relationship of nodes within the graph and are often represented in terms of an adjacency matrix  $\mathbf{A} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$  where  $A_{ij} = 1$  if  $e_{ij} \in \mathcal{E}$  and  $A_{ij} = 0$  if  $e_{ij} \notin \mathcal{E}$ . Both nodes and edges can have an associated vector that describe its properties denoted as node features  $f_v$  and edge features  $f_e$  respectively. In practice, nodes along with their features are represented by combining these node vectors into a single matrix  $\mathbf{F}_\mathcal{V} = \mathbb{R}^{|\mathcal{V}| \times |f_v|}$ . Edge features are represented as a separate matrix  $\mathbf{F}_\mathcal{E} = \mathbb{R}^{|\mathcal{E}| \times |f_e|}$ .

Graphs can be categorized based on their structural properties.

- *Directed vs. Undirected.* Directed graphs are graphs whose edges have an associated direction to them. This means that the relationship between two connected nodes only goes in the direction given by the edge. For example, an edge  $e_{ij}$  (meaning  $A_{ij} = 1$ ) indicates a direction from the source node  $i$  to the target node  $j$ . The edges of an undirected graph have no associated direction, which means the connection between two nodes goes both ways. The adjacency matrix of an undirected graph is symmetric to reflect this property ( $A = A^T$ ,  $A_{ij} = A_{ji}$ ).
- *Homogeneous vs. Heterogeneous.* Homogeneous graphs only have one type of node and edge and are the type described in the prior definition. Heterogeneous graphs, on the other hand, can have multiple types of nodes, each of which represents a different type of entity (Zhang et al., 2019). There are also multiple types of edge that can connect nodes of the same or different type. Mathematically, this is defined as  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{O}_\mathcal{V}, \mathcal{R}_\mathcal{E})$  where  $\mathcal{O}_\mathcal{V}$  and  $\mathcal{R}_\mathcal{E}$  represent the set of node and edge types, respectively. Edges have

a corresponding mapping function that is used to describe the relationship between two different entities (Luo et al., 2024). For example, the Microsoft Academic Graph dataset is a widely used benchmark that captures the landscape of academic publications through a heterogeneous graph (K. Wang et al., 2020). Consider two types of nodes the graph defines: publications and authors. Publications can be connected to authors through the “has authors” relationship, which is represented by a directed edge from publication to author. Publications can also be connected to other publications through a “citing” relationship represented by a different type of edge.

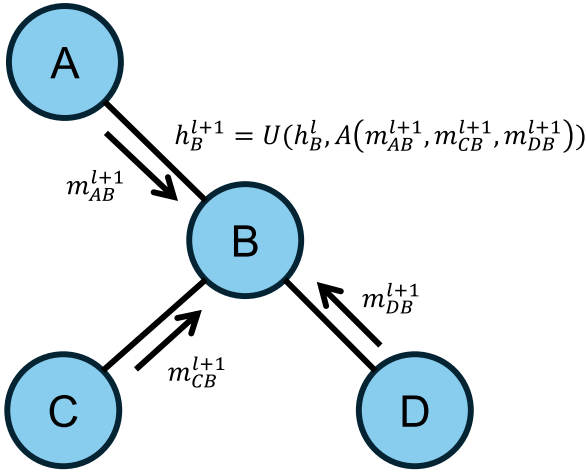
### 2.2.2. Message passing

In this survey, the message passing framework will be used to understand the core mechanics behind graph convolution (Gilmer et al., 2017). Although this uses a spatial interpretation, spectral-based methods can also be understood in terms of this framework. Additionally, popular geometric DL libraries implement graph convolution modules through this framework (Fey and Lenssen, 2019; M. Wang et al., 2020). This discussion will present a generalized adaptation of the framework, informed by the current comprehensive understanding of GNNs.

A message passing operation for a given node collects information from its immediate neighbors  $\mathcal{N}(v_i) = v_j \in \mathcal{V} : e_{ij} \in \mathcal{E}$  and uses it to update its current features or representation. After this is performed, the node would then have a new latent representation which holds information regarding itself and its neighborhood. This process is usually repeated for multiple iterations, with a new latent embedding being generated for all nodes each time. Each layer in a GNN corresponds to one iteration of the message passing operation. After message passing, each node representation incorporates information from its neighboring nodes. As the number of message-passing layers increases, information can propagate over longer graph distances, allowing each node to capture broader spatial context. However, using too many layers may induce oversmoothing, where latent node representations become increasingly similar and lose discriminative information.

Message passing is composed of three main parts: message generation, message aggregation, and node update. In flood modeling, this operation can be interpreted as information exchange between physically connected or spatially related units, such as gauges, junctions, particles, or mesh cells, where messages encode hydraulic or spatial interactions used to update latent flood-state representations. This process can be described by the following equation:

$$h_i^{l+1} = U(h_i^l, A(\{M(h_i^l, h_j^l, e_{ij}); \forall j \in \mathcal{N}(v_i)\}))$$



**Fig. 2.** An example of the message passing operation on node B. Messages  $m_{AB}^{l+1}$ ,  $m_{CB}^{l+1}$  and  $m_{DB}^{l+1}$  are sent from neighboring nodes A, C and D respectively. These messages are aggregated by the aggregation function  $A$ . The node embedding of B is then updated by combining the aggregated messages with the current node embeddings through the update function  $U$ .

where  $h_i^l$  and  $h_i^{l+1}$  is the embedding of node  $i$  for the current and next layer, and  $M$ ,  $A$  and  $U$  are the message, aggregate and update functions, respectively. Fig. 2 gives a visualization of this message passing operation.

In the message generation phase, a message  $m_{ji}$  is generated between a node and each of its neighbors. These messages are generated using the node's features, neighboring node's features, and optionally, the edge features connecting the two nodes. Mathematically, this is represented as

$$m_{j \rightarrow i} = M(h_i, h_j, e_{ij})$$

For message aggregation, all messages from a node's neighbors are combined into a single representation. In the original paper, this aggregation operation was presented as a sum (Gilmer et al., 2017). However, since then, different aggregation techniques have been explored, such as taking the average or max of messages, each with their own implications (Xu et al., 2019). A vital criteria for the aggregation function is permutation invariance to reflect how messages have no inherent order and can be interchanged. This stage can be represented by the following equation:

$$m_i^{l+1} = A(\{m_{j \rightarrow i}^{l+1}; \forall j \in N(v_i)\})$$

Finally, during node update, the node representation is updated using the aggregated value from the messages. This involves a combination between the node's current embedding and the aggregated messages from its neighbors. Oftentimes, this operation is a feedforward network to introduce learnable parameters to the model. It is also important to note that a non-linear activation function should be applied to the network to enable the modeling of complex nonlinear relationships. This can be mathematically expressed as:

$$h_i^{l+1} = U(h_i^l, m_i^{l+1})$$

### 2.2.3. Common GNN architectures

Over time, gradual improvements have been made to extend the message passing operation and enhance the capabilities of GNNs. Table 2 summarizes commonly used GNN architectures and their key contributions. These architectures form the foundation for more complex, domain-specific models in various studies of graph-based DL.

### 2.2.4. Physics-informed GNNs

Combining physics-informed and data-driven methods enables the creation of models that are interpretable, generalizable, and data-efficient. However, designing loss functions or model components grounded in physical laws often requires significant mathematical rigor and domain expertise. Compared to other DL models, GNNs offer a more natural framework to integrate these approaches.

Raissi et al. distinguish between two strategies for applying physics-informed loss: the continuous time model and the discrete time model (Raissi et al., 2019). Continuous time models act as spatiotemporal function approximators that directly learn the latent solution across the entire domain. Overall, they offer flexibility but are computationally and data intensive and difficult to calibrate for complex real-world scenarios (Gao et al., 2022). On the other hand, discrete time models utilize implicit Runge–Kutta schemes to take large, highly accurate time steps, enabling them to efficiently resolve solutions with much less data. This is the approach adopted by efficient numerical solvers for PDEs (e.g., finite-volume, finite-element methods). PINNs incorporate discrete time models by applying convolution operations typically implemented as CNNs (Gao et al., 2022). These suffer from the same limitations of requiring structured Euclidean data which GNNs directly overcome. Incorporating an inductive bias that directly maps to a mesh-like topology allows GNNs to generalize well to unseen cases in physical domains (Thangamuthu et al., 2022). Furthermore, utilizing a flexible graph structure allows for ease of incorporating varying boundary conditions and focus on complex areas of high interest.

Physics-informed architectures can be challenging to construct as they involve integrating complex physics laws into pre-existing ML algorithms. In this regard, GNNs offer a unique advantage as the core message passing operation is a general framework that can be easily modified for specific applications. While not required, this is often done in conjunction with an encode-process-decode structure (Battaglia et al., 2018; Jiakuan You et al., 2021). The node and edge feature matrices pass through an encoder which gives them an identical shape and provides the option of combining them through various matrix operations. In general, a separate encoder is used for node and edge features as they have a different semantic meaning in a graph structure. After processing the graphs through message passing, the latent representations of nodes are then passed through a decoder to generate the final output.

## 3. Review of current approaches

This section explores the application of GNNs in flood modeling by first contextualizing GNNs in this domain (Section 3.1), and introducing the literature selection process (Section 3.2). A high-level review of all the papers is provided in Section 3.3. Following this, studies are further discussed by category, specifically as methods that are either point-based (Section 3.4), particle-based (Section 3.5) or mesh-based (Section 3.6).

### 3.1. Contextualizing GNNs for flood modeling

GNNs offer a compelling framework for modeling flood dynamics, particularly due to their ability to handle unstructured spatial data, encode localized spatial interactions, and integrate physics-informed constraints. These strengths make them well-suited for capturing complex hydraulic and hydrodynamic processes across spatially distributed systems such as river basins, urban drainage networks, and floodplains. Unlike grid-based DL models, GNNs operate directly on irregular graph structures, enabling seamless representation of unstructured computational meshes commonly used in hydrodynamic models. Furthermore, the localized message-passing mechanism of GNNs closely resembles the neighborhood-based flux exchange processes used in numerical solvers, where hydrodynamic states are updated based on interactions between adjacent cells. This inductive bias promotes physically

**Table 2**  
Common GNN architectures and their key contributions.

| Model                                                                | Key contribution                     | Description                                                                                                                             | Relevance to flood modeling                                          |
|----------------------------------------------------------------------|--------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| Graph Convolutional Network (GCN) (Kipf and Welling, 2017)           | Localized spectral convolution       | Localized first-order approximation of spectral graph convolution; aggregates one-hop neighbors with symmetric degree normalization     | Baseline for most point-based flood GNNs                             |
| Graph Attention Network (GAT) (Velićković et al., 2018)              | Attention-weighted messages          | Learns an importance weight per edge, scaling each message                                                                              | Weights hydraulically influential connections                        |
| Graph Sample and aggregate (GraphSAGE) (Hamilton et al., 2018)       | Sampled-neighborhood aggregation     | Message passing over a sampled neighborhood rather than the full graph                                                                  | Enables scaling to operational mesh sizes                            |
| Graph Convolution Recurrent Neural Network (GCRN) (Seo et al., 2016) | Spatiotemporal message passing       | Combines graph convolution with recurrent units, stacked or embedded                                                                    | Captures flood temporal dynamics                                     |
| Graph Network (GN) (Battaglia et al., 2018)                          | Generalized node-edge-global updates | Unifies message passing into node, edge, and global update functions, each independently parameterized                                  | Basis for edge-level flux prediction and particle-based flood models |
| MeshGraphNet (Pfaff et al., 2021)                                    | Mesh-based physical surrogate        | Encode-process-decode architecture designed to mimic mesh-based physical simulations, demonstrating GNNs as effective surrogate models. | Direct template for mesh-based flood surrogates                      |

meaningful spatial learning and can improve transferability across varying spatial configurations and mesh resolutions. Applications employ core GNN architectures which can then serve as foundational building blocks for domain-specific modifications. Such adaptations may include the incorporation of physics-based loss functions, feature engineering with hydrological/hydrodynamic variables, or integration of physical modeling principles with data-driven learning.

The prerequisite for all implementations is the representation of the target catchment as a formal graph structure. Graph construction remains a challenge because different flood systems imply different notions of connectivity. For example, drainage systems are comprised of physical pipes, junctions and surcharge pathways, while coastal applications have more implicit connectivity. Poor graph construction can exclude important hydraulic pathways or introduce non-physical relationships. Thus, graph construction remains effectively ad hoc across the field since choices are made largely by convention rather than comparatively. In general, nodes represent a point or section within the area while edges represent relationships, both explicit and implicit, between these. Each node in the graph is associated with a set of features that characterize the state of that region. These features can be broadly categorized into static and dynamic types. Static node features remain constant throughout the flood event and typically include variables such as cell area, elevation, and Manning's roughness coefficient. Dynamic node features vary over time and capture the evolving state of the system. Examples include rainfall intensity, water level, and water volume at a given timestep. Similarly, the edges of the graph describe the physical and/or implicit relationships between connected nodes. Static edge features may include the distance and slope between two nodes. Dynamic edge features, on the other hand, can represent time-varying quantities such as flow velocity or discharge between adjacent regions. This combination of static and dynamic features allows GNNs to learn both the spatial structure and the temporal evolution of flood processes.

Flood modeling is inherently a spatiotemporal problem, where the dynamic evolution of a flood event is captured by changes in the physical state of a catchment over time. This can be represented as a sequence of graphs  $\{\mathcal{G}^0, \mathcal{G}^1, \dots, \mathcal{G}^T\}$  over  $T$  timesteps, where  $\mathcal{G}_0$  denotes the initial state of the system. Each graph encodes the spatial structure of the domain, along with the relevant static and dynamic features at the nodes and edges. A direct connection with numerical flood solvers arises when the computational domain is discretized into cells or elements: numerical solvers update hydraulic states through

local exchanges across cell interfaces, while GNNs can use the same connectivity as a graph and learn message-passing operations that approximate these local state interactions. The primary objective in GNN-based flood modeling is to predict one or more dependent variables that characterize the temporal evolution of the flood, such as water level, water volume, velocity, or discharge. These target variables may be associated with nodes (e.g., water level in a cell) or edges (e.g., velocity or discharge between cells), depending on the model design and prediction task.

The goal is to train a model  $\mathcal{Y}_\theta$ , parameterized by  $\theta$ , that can accurately predict the target variable(s) at time  $t + 1$ , based on the sequence of input graphs from the previous  $p$  timesteps. This can be formally expressed as:

$$\mathbf{Y}^{t+1} = \mathcal{Y}_\theta(\mathcal{G}_{t-p}, \mathcal{G}_{t-p+1}, \dots, \mathcal{G}_{t-1}, \mathcal{G}_t)$$

where  $\mathbf{Y}^{t+1}$  denotes the predicted values of the target variable(s) for all relevant nodes or edges at time  $t + 1$ .

### 3.2. Methodology

A PRISMA-style approach was adopted to produce the list of publications for this survey. During the identification stage, papers were queried from the Scopus database as long as they contain both GNN-related and hydrology-related keywords. A total of 413 total papers were found in the initial search. Additional rule-based filters were applied ensuring only journal papers published since 2018 until March 2026 and under the subject areas Engineering, Environmental Science, Earth and Planetary Sciences were included, resulting in 193 remaining publications. From this, high-level screening was performed by reviewing each title and abstract. The following criteria were considered for a paper's disqualification: (1) loose use of graph terminology in non-GNN approaches (2) non-flooding applications (e.g., water quality, traffic flow forecasting), and (3) purely spatial prediction tasks. Lastly, a snowball search was performed to find related papers cited in the existing reference repository. A complete list of 49 papers were gathered for an in-depth review. A flowchart illustrating the entire paper screening process can be found in Fig. 3.

### 3.3. General findings

The screened collection of peer-reviewed studies were categorized into six unique application types depending on the construction of their

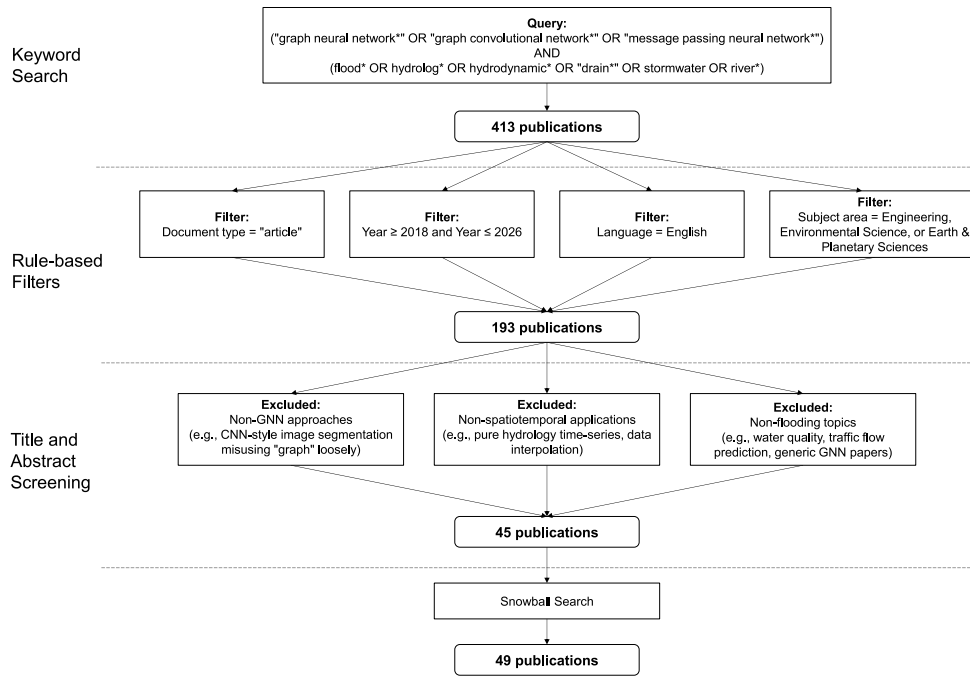


Fig. 3. The flowchart of the search strategy used to find the list of publications for review.

underlying graph structure. Study areas can be represented as either point-based graphs, particle-based graphs or mesh-based graphs. Point-based applications can further be classified based on the established node definition as either river networks, coastal regions, urban regions or storm water drainage systems (SWDS). An enumeration of the list of references under each application type and level of physics incorporation can be found in Table 3. Additionally, Fig. 4 displays the distribution of published papers across the years. Most of the studies were published between 2023 and 2026 following a clear positive trend, which reflects the growing interest in this emerging research area. The number of publications in the first quarter of 2026 is already more than half of the total papers published in 2025. This proves the growing interest and opportunities in this subject area. In the following subsections, each application type is further discussed in detail to identify key patterns in existing approaches. However, as mentioned in the introduction, this paper will mainly focus on mesh-based modeling, which would subsequently receive a deeper analysis.

### 3.4. Point-based modeling

Point-based modeling constructs the graph structure by setting nodes as locations with available time series observations. These are typically associated with physical infrastructure such as gauging stations that measure hydraulic variables including water level and discharge. This type of application facilitates direct representation of real-world measurement systems, but exhibits node sparsity due to practical limitations in sensor deployment. Subcategories under this application type are further identified by the specific entities represented by nodes and the relationships represented by edges.

In point-based flood modeling, the role of GNNs is to learn spatial dependencies among observation locations by propagating information through graph edges that represent hydrological, hydraulic, geographic, or statistical relationships. This allows point observations, such as water level or discharge at gauging stations, to be interpreted not as independent time series but as interacting components of a connected flood system.

#### 3.4.1. River networks

The most common subcategory under this application type is the modeling of river networks, which accounts for more than half of the collected publications. In this context, nodes represent measurement stations along a river that record observations for water level and/or streamflow over time. As the number of stations increases, modeling complexity also increases because the model must accommodate heterogeneous flow regimes across different locations. Data quality challenges may also become more pronounced, since wider monitoring networks are more likely to contain missing, inconsistent, or faulty records. Due to the prevalence of these systems, a number of methodologies have been developed under these conditions.

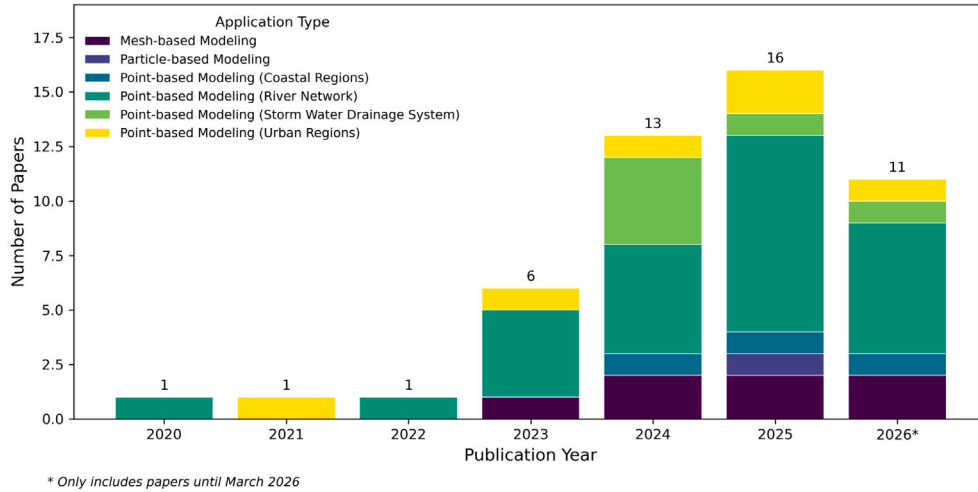
The multifaceted relationships within a river network reveals multiple strategies for graph edge construction. Typically, these are built directly mirroring the river topology, to reflect the upstream and downstream relationships between nodes (Akkala et al., 2025a,b; Charania and Patel, 2026; Sui et al., 2026; Zhang et al., 2025). Other more implicit relationships based on Euclidean distance (Guo et al., 2025; Huang et al., 2025) or Pearson correlation (Kilinc et al., 2025; Yang et al., 2023; Zhao et al., 2020) have also been explored. However, a static graph structure has limitations in capturing evolving flow correlations, especially in dynamic hydrological systems (Xue and Zhu, 2025). Consequently, a more adaptive approach involves dynamically modifying the edge connectivity over time to account for flow-altering events such as reservoir operations, underwater landslides, and land-use evolution (R. Liu et al., 2026; Nguyen et al., 2026; Xue and Zhu, 2025). On top of solely varying edge connectivity, heterogeneous graph representations have also been incorporated to encode diverse hydrologic, hydraulic and geographic interactions through multiple node and edge types (J. Jiang et al., 2024; Luo et al., 2024).

In terms of model architecture, some studies adopt existing GNN architectures, such as GCN (Kipf and Welling, 2017) and GraphSAGE (Hamilton et al., 2018), to measure their effectiveness in the context of flood modeling (Rocha et al., 2025; Victor et al., 2023). Their ability to model spatial relationships has been shown to outperform other DL paradigms. For example, in a regression task predicting future daily

**Table 3**

Classification of papers by application type and physical fidelity.

| Application type                                      | Physical fidelity                   | References                                                                                                                                                                                                                                                                                                                             |
|-------------------------------------------------------|-------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Point-based modeling<br>(River Network)               | Purely data-driven                  | Akkala et al. (2025a), Charania and Patel (2026), Rocha et al. (2025), Deng et al. (2024), Guo et al. (2025), J. Jiang et al. (2024), Kilinc et al. (2025), Y. Li et al. (2024), R. Liu et al. (2026), Roudbari et al. (2024), Victor et al. (2023), Sui et al. (2026), Wang and Zhu (2025), Xue and Zhu (2025) and Yang et al. (2023) |
|                                                       | Physics-informed by inputs          | Akkala et al. (2025b), Huang et al. (2025), Jin et al. (2026), Liu et al. (2023), Luo et al. (2024), Nguyen et al. (2026), Sun et al. (2022), Zhang et al. (2025) and Zhao et al. (2020)                                                                                                                                               |
|                                                       | Physics-constrained by architecture | S. Liu et al. (2026) and Zhou et al. (2025)                                                                                                                                                                                                                                                                                            |
| Point-based modeling<br>(Coastal Regions)             | Purely data-driven                  | W. Jiang et al. (2024)                                                                                                                                                                                                                                                                                                                 |
|                                                       | Physics-informed by inputs          | Zhangke et al. (2026)                                                                                                                                                                                                                                                                                                                  |
|                                                       | Physics-informed by loss            | Li et al. (2025)                                                                                                                                                                                                                                                                                                                       |
| Point-based modeling<br>(Urban Regions)               | Purely data-driven                  | Chaimook et al. (2025), Fang et al. (2025), Farahmand et al. (2023) and Liang et al. (2024)                                                                                                                                                                                                                                            |
|                                                       | Physics-informed by inputs          | Agonafir and Zheng (2026) and A.Y. Sun et al. (2021)                                                                                                                                                                                                                                                                                   |
| Point-based modeling<br>(Storm Water Drainage System) | Physics-informed by inputs          | He et al. (2025) and M. Li et al. (2024)                                                                                                                                                                                                                                                                                               |
|                                                       | Physics-constrained by architecture | Garzón et al. (2024b,a, 2026) and Zhang et al. (2024)                                                                                                                                                                                                                                                                                  |
| Particle-based modeling                               | Physics-informed by inputs          | Shao et al. (2025)                                                                                                                                                                                                                                                                                                                     |
| Mesh-based modeling                                   | Physics-informed by inputs          | Kazadi et al. (2024), Pan et al. (2026) and Taghizadeh et al. (2025a)                                                                                                                                                                                                                                                                  |
|                                                       | Physics-informed by loss            | Acosta et al. (2026) and Taghizadeh et al. (2025b)                                                                                                                                                                                                                                                                                     |
|                                                       | Physics-constrained by architecture | Bentivoglio et al. (2023) and Bentivoglio et al. (2024)                                                                                                                                                                                                                                                                                |

**Fig. 4.** The number of papers per year categorized by application type.

inflow for the Shetrunji River Basin, GCN achieved the highest accuracy (RMSE = 26.859 m<sup>3</sup>/s, Nash–Sutcliffe Efficiency (NSE) = 0.874) compared with MLP (RMSE = 35.901 m<sup>3</sup>/s, NSE = 0.773) and hybrid LSTM–GRU (RMSE = 42.292 m<sup>3</sup>/s, NSE = 0.682) (Charania and Patel, 2026). However, the graph structure only encodes spatial dependencies through edge connectivity and has limited expressiveness in modeling complex temporal dynamics (Akkala et al., 2025b). To address this, a prominent approach is to integrate additional architectural components to explicitly capture temporal relationships. While LSTMs have been the most utilized across models (Akkala et al., 2025a,b; Guo et al., 2025; Kilinc et al., 2025; S. Liu et al., 2026; Luo et al., 2024; Wang and Zhu, 2025; Xue and Zhu, 2025), other approaches have also been considered including MLPs (Y. Li et al., 2024), GRUs (J. Jiang et al., 2024; Zhao et al., 2020), 1D convolutions (Jin et al., 2026; Sui et al., 2026; Sun et al., 2022) and transformers (Huang et al., 2025; Nguyen et al., 2026).

While the majority of GNN models are purely data-driven, others have also applied physics-informed principles in their inference process.

For example, a non-invasive approach to incorporating physics is by enriching input features using hydrological models such as HEC-HMS (Huang et al., 2025), DHDM (Zhang et al., 2025) and NWM (Sun et al., 2022). Only two studies have directly constrained their model architectures to follow physical laws. Zhou et al. incorporates the delay time from water traversing a specific channel through learnable parameters, which constrain the directed adjacency matrix (Zhou et al., 2025). On the other hand, Liu et al. incorporate a gated fusion mechanism that applies a ReLU to tanh activation function sequence to mimic the dominance of upstream flow over downstream flow (S. Liu et al., 2026).

#### 3.4.2. Coastal regions

Characterizing coastal flood dynamics requires monitoring stations situated near drainage outlets. While these systems share similarities with river network modeling, their proximity to the ocean introduces highly dynamic downstream relationships due to the influence of tidal forces and atmospheric conditions (Li et al., 2025).

To address this, GNNs are often combined with temporal mechanisms (e.g., GRU) to capture rapid variations induced by tides and extreme events (W. Jiang et al., 2024; Zhangke et al., 2026). Additionally, additional features specific to coastal hydrology are incorporated as model input. For example, aside from historical water levels, studies incorporated external hurricane-related factors such as zonal/meridional wind stress, surface/sea level pressure, translation speed and heading angle to improve their models' predictive accuracy (W. Jiang et al., 2024; Zhangke et al., 2026). Other studies utilized the incorporation of physics as an effective tool for modeling storm dependencies. Li et al. implemented a state-aware weighted loss function that categorizes water levels into "flood", "typical", or "drought" states and assigns heavier penalties to prediction errors during critical extreme events such as storm surges.

### 3.4.3. Urban regions

In the case of metropolitan areas, nodes are described in terms of established urban constructs such as cities, districts and census tracts. As there is no explicit physical connectivity between these regions, varying edge building strategies have been adopted including full connectivity (Agonafir and Zheng, 2026) and connectivity based on geographic distance (Chaimook et al., 2025; Liang et al., 2024), and/or hydrological similarity (A.Y. Sun et al., 2021).

In this setting, there is a lack of physical sensors at a street-level resolution as these measurements are only available on a broader basin scale (A.Y. Sun et al., 2021). As such, ground truth data must be captured through other means. For example, images of urban floods from social media platforms like Sina Weibo have been used to generate a classification dataset based on flooding intensity (Fang et al., 2025; Liang et al., 2024). Human-sensed data from such platforms have also been used to inform model predictions to simulate real-time situational awareness (Farahmand et al., 2023). Historical records detailing crowd-sourced city complaints and municipal-level flood occurrence have also been used to construct a dataset (Agonafir and Zheng, 2026; Chaimook et al., 2025). The absence of a standardized graph structure from gauge locations has also opened the possibility of arbitrarily defined inference points. For example, Fang et al. divide the urban catchment into a 30 m  $\times$  30 m grid and map point-based measurements to their corresponding grid cells (Fang et al., 2025). This interpolation method can be a starting point for reconciling point-based into mesh-based modeling, enabling point-wise datasets to be used for grid-constrained prediction tasks.

Urban flood behavior is heavily dictated by man-made influence and infrastructure. To capture this, factors such population and building density, sewer backups and catch basin blockages are incorporated as model inputs (Agonafir and Zheng, 2026; Chaimook et al., 2025). However, these structures also affect local hydrodynamics which introduces complexity when incorporating physics constraints. As a result, most publications in this study area remain purely data driven.

### 3.4.4. Storm water drainage systems

While SWDS also operate in the urban setting, these systems constrain their modeling to the underlying drainage network. This gives them a localized perspective of the flood event rather than having to consider city-wide surface runoff.

This problem formulation provides two key benefits. First, it provides an explicit physical mapping of the graph structure by interpreting junctions or manholes as nodes and the connecting pipes as edges. This standardizes dataset construction and facilitates the incorporation of physical properties (e.g., pipe diameter, length) as input features (He et al., 2025). Second, it simplifies flood hydrodynamics to 1D movement along the direction of the pipe. Fluid motion in this scenario can be efficiently simulated through physics-based numerical models, specifically Storm Water Management Model (SWMM). These models can be used to generate large synthetic datasets, which implicitly imbues GNNs with physical knowledge through data (He et al., 2025; M.

Li et al., 2024; Zhang et al., 2024). Additionally, 1D numerical models can comprehensively quantify hydraulic data at each point of the system, whether it be flow along pipes or water level in junctions. As a result, GNN flood models under this application have been designed to predict both node and edge features (Garzón et al., 2024a, 2026; M. Li et al., 2024; Zhang et al., 2024).

In terms of physics incorporation, the simplicity of 1D pipe flow eases the construction of physics-constrained model architectures for this use case. For instance, (Zhang et al., 2024) integrate physical fidelity in model predictions by computing the inflow and outflow at each node based on its adjacent edges, ensuring mass conservation and coupling predictions across spatial scales. On the other hand, Garzon et al. enforce physical constraints by applying a min-max operation that clips the predicted hydraulic heads so they cannot drop below the pipe invert elevation or exceed the street terrain elevation (Garzón et al., 2024b,a, 2026).

## 3.5. Particle-based modeling

Particle-based modeling captures flood behavior at the particle level. Compared to point-based and mesh-based modeling, it operates at a much smaller scale by assigning each particle as a node in the graph. Since particle movement is dynamic and volatile, edges within the graph must be constantly reconstructed at each timestep based on its nearest neighbors (Shao et al., 2025). This reconstruction step is necessary in this context, in contrast to other applications where this technique is only used to augment existing performance (J. Jiang et al., 2024; Nguyen et al., 2026).

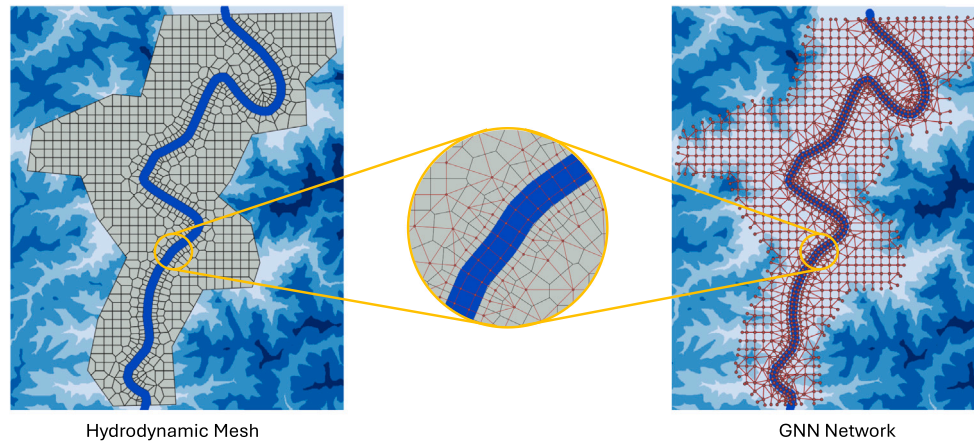
This approach is extensively studied in the field of fluid mechanics, where ML models are designed to be surrogate approximations of numerical Smoothed-Particle Hydrodynamic (SPH) models (Li and Farimani, 2022; Sanchez-Gonzalez et al., 2020). However in the context of flood modeling, there is only one publication from the collected references that uses this modeling schema. Specifically, Shao et al. directly extend the GN model architecture for rapid flood forecasting (Shao et al., 2025). They employ a METIS-based subgraph partitioning strategy that significantly reduces GPU memory usage, the typical bottleneck for high-resolution forecasting.

Nevertheless, particle-level flood simulations are inevitably constrained by prohibitive memory demands from quantifying the mechanics of each particle. This is a limitation it shares with an SPH solver, which also introduces a bottleneck during dataset generation. This limits their scalability for large-scale applications relative to modeling techniques with a higher level of abstraction.

## 3.6. Mesh-based modeling

Mesh-based modeling balances the high computational demands of particle-based methods and the sparse spatial coverage of point-based methods. This survey primarily focuses on this approach because it offers the clearest path toward a direct surrogate replacement of hydrodynamic solvers. Both operate on spatial discretizations and update local states through neighborhood interactions. This alignment enables physics-informed priors such as boundary conditions to be easily embedded during modeling. A shared mesh representation also simplifies dataset generation, as numerical solvers output dense ground truth at every cell and timestep without any need for interpolation.

Numerical methods represent a catchment using a structured or unstructured mesh, which discretizes the domain into a set of cells. To translate this mesh into a graph representation, nodes are typically defined at the centroids of mesh cells, while edges are established between adjacent cell centers, reflecting the physical connectivity of the domain. This graph representation preserves the spatial and topological relationships inherent in the original mesh, enabling the use of GNNs to model flow dynamics across the domain. Mesh characteristics can



**Fig. 5.** Graph-based representation of a 2D hydrodynamic model mesh. Red lines illustrate the underlying graph structure: the centroid of each mesh cell is represented as a graph node, while connections between adjacent cell centers define the graph edges.

**Table 4**  
Selected Mesh-based GNN flood models and their study objectives.

| Model                                    | Study objective                                                                                                                                                                           |
|------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SWE-GNN (Bentivoglio et al., 2023)       | To develop a hydrodynamic-inspired GNN surrogate model for fast and generalizable spatiotemporal flood simulations that mimics finite-volume methods                                      |
| FloodGNN-GRU (Kazadi et al., 2024)       | To develop a spatiotemporal flood prediction model that combines GNNs with GRUs to efficiently emulate physics-based inundation models for fast and accurate short-term flood forecasting |
| mSWE-GNN (Bentivoglio et al., 2024)      | To develop a multi-scale GNN that models flood dynamics across resolutions, handling different flow speeds and generalizing to unseen topographies and boundary conditions                |
| HydroGraphNet (Taghizadeh et al., 2025a) | To develop a physics-informed GNN that integrates mass conservation laws and enhances interpretability for flood forecasting in unstructured mesh domains                                 |
| MFGNN (Taghizadeh et al., 2025b)         | To develop a data efficient GNN with limited reliance on expensive high-fidelity data for large-scale fluvial and pluvial flood modeling                                                  |
| GNN-Transformer (Pan et al., 2026)       | To develop a surrogate model with a GNN module for spatial learning and Transformer module for temporal encoding for short-duration intense-rainfall flood forecasting                    |
| DUALFloodGNN (Acosta et al., 2026)       | To develop a GNN flood model that can simultaneously predict hydrodynamic state at both graph nodes and edges while following mass conservation at both global and local scale            |

strongly affect GNN-based flood-model performance. Finer mesh discretization can better capture local topographic controls, wetting and drying fronts, and spatial gradients in water depth or velocity, but it also increases graph size, memory demand, and computational cost. Mesh quality is also important, as poorly shaped cells, abrupt resolution transitions, or graph connectivity that does not align with hydraulic exchange pathways may reduce physical consistency and degrade predictive performance. A schematic illustration of this transformation is provided in Fig. 5.

To provide an initial overview, a summary of studies for this application type and modeling objectives, is listed chronologically in Table 4. Additionally, Table 5 provides a comparative overview of the GNN architectures used in the selected flood models. It highlights the key DL components integrated into each model, including the type of message passing operation, temporal unit, encoder–decoder structure, and the use of attention mechanisms.

Long-term flood modeling that mimics the capabilities of numerical hydrodynamic models introduces challenges in maintaining temporal consistency and physical plausibility of predictions. This issue arises primarily from the autoregressive nature of ML flood models, where small prediction errors can accumulate and propagate over time, potentially leading to divergence from physically realistic behavior. There are two enhancements to mesh-based flood models that mitigate these

challenges: dedicated temporal learning components and governing physical constraints.

While GNNs naturally consider spatial dependencies through edges that represent physical relationships between mesh cells, flood forecasting also requires modeling complex temporal dynamics, as the input typically consists of a sequence of graphs over time. The integration of additional neural network modules results in hybrid architectures that address these concerns due to their enhanced spatiotemporal learning capabilities. FloodGNN-GRU employs GCRNs in an embedded fashion, replacing traditional MLPs within RNNs with graph convolutional layers (Kazadi et al., 2024). It also incorporates an attention mechanism to highlight the most influential spatial relationships within the graph, thus improving predictive accuracy and interpretability. On the other hand, Pan et al. utilize a GNN-Transformer architecture that ingests sinusoidal positional encodings based on the temporal rainfall sequence (Pan et al., 2026). To enhance spatial robustness, they employed geometric data augmentation, including horizontal flips, vertical flips and rotations, to optimize the network's predictive performance across complex topologies. Notably, their study had the dataset with the largest graph size among the reviewed literature.

An alternative approach which has recently gained prominence is by incorporating physics-informed constraints into GNN architectures. By embedding physical principles directly into the learning process, these

**Table 5**  
Architectural details of Mesh-based GNN flood models.

| Model           | Prediction target | Message passing  | Temporal unit | Encoder-Decoder  | Attention |
|-----------------|-------------------|------------------|---------------|------------------|-----------|
| SWE-GNN         | Node              | Physics-informed | None          | MLP              | No        |
| FloodGNN-GRU    | Node              | GVP <sup>a</sup> | GRU           | GVP              | Yes       |
| mSWE-GNN        | Node              | Physics-informed | None          | MLP              | No        |
| HydroGraphNet   | Node              | Graph Network    | None          | KAN <sup>b</sup> | No        |
| MFGNN           | Node              | Graph Network    | None          | MLP              | No        |
| GNN-Transformer | Node              | MLP              | Transformer   | None             | Yes       |
| DUALFloodGNN    | Node and Edge     | MLP              | None          | MLP              | No        |

<sup>a</sup> Geometric Vector Perceptron.

<sup>b</sup> Kolmogorov–Arnold Network.

models aim to generate physically consistent predictions and improve generalizability to unseen flood events and locations.

SWE-GNN, proposed by Bentivoglio et al. (2023), integrates hydrodynamic constraints derived from the SWE within a finite-volume framework. During the message-passing process, SWE-GNN modulates messages using the difference in latent dynamic features, specifically encoded water levels, between neighboring nodes. This design enforces the physical intuition that flow should occur only when there is a water level gradient. To further ensure that dry cells (i.e., nodes with zero water depth) remain inactive, the authors remove the bias terms from the encoder and decoder MLPs, thereby reducing the risk of spurious non-zero predictions in areas where no flooding is expected.

Building upon their earlier work, the same research group introduced mSWE-GNN (Bentivoglio et al., 2024) to address key limitations of the original SWE-GNN framework. To accommodate time-varying boundary conditions, they incorporated additional boundary nodes that act as ghost cells, effectively simulating inflow dynamics at the domain boundaries. Furthermore, they implemented a multi-scale graph architecture, enabling the model to capture long-range spatial dependencies between distant nodes with fewer message-passing iterations. This enhancement improves computational efficiency while preserving the model's ability to represent complex flood propagation patterns.

A different physics-informed strategy is employed in HydroGraphNet, developed by Taghizadeh et al. (2025a). Rather than embedding physics into the architecture itself, they incorporate it into the loss function. The model simultaneously predicts water depth and volume for all nodes and includes a penalty term to enforce the physical relationship between these two quantities. A forward and backward check with the predicted volume is performed to ensure that the volume changes are within bounds from the previous and next timesteps, respectively.

However, applying physics-informed loss at a global level does not guarantee the same behavior for local flood dynamics. DUALFloodGNN address this gap by explicitly enforcing mass conservation at a node-level (Acosta et al., 2026). To do this, it simultaneously predicts water volume at nodes and flow at edges both constrained by the prior local loss. This study also presents a comparative evaluation of GNN-based flood models, demonstrating that the proposed design outperformed both SWE-GNN and HydroGraphNet in water depth regression.

A significant bottleneck in DL-based approaches is the reliance on large datasets, which is difficult to obtain for higher graph resolutions. To address this, MFGNN introduces a hierarchical learning architecture where the output of a GNN trained on abundant low-resolution data is corrected by a GNN trained on sparse high-resolution data (Taghizadeh et al., 2025b). Low-fidelity spatial predictions are upsampled to a finer mesh using inverse-distance weighting. Additionally, instead of supplying dynamic features as a time series, descriptive statistics are derived from the input data acting as a form of dimensionality reduction.

Table 6 summarizes the strategies used across the various models to incorporate physical principles into the learning framework. Three prominent strategies appear as previously identified: physics-informed inputs, physics-informed loss and physics-constrained architecture. The literature reflects a relatively even distribution of exploration across these three approaches. Of the three, physics-informed input is the most common, as numerical models inherently imbue their outputs

with physical consistency by virtue of being generated from those laws. Table 7 enumerates these input features categorized into node and edge features, which is then further classified as either static or dynamic. Notably, most models treat dynamic features as node-level inputs. Taken together, these strategies form a spectrum of increasing physical directness, ranging from implicit guidance through physically consistent training data to explicit enforcement of governing equations within the architecture or loss function.

#### 4. Analysis of GNNs-based flood models

This section reveals key trends in the covered literature, highlighting overall strengths and limitations in Sections 4.1 and 4.2 respectively. A synthesis of these discussions is featured in Section 4.3, which highlights research opportunities through the lens of flood GNN operationalization.

##### 4.1. Advantages and strengths

GNNs have gained increasing attention in flood modeling due to their ability to process irregular, graph-structured data and capture complex spatiotemporal dependencies. They also offer advantages in computational efficiency, adaptability, and integration of physical knowledge. The key strengths of GNN-based flood models are discussed in this section.

##### 4.1.1. Representational power

The inherent applicability of the graph structure facilitates encoding diverse semantics for deeper contextualization prior to GNN processing. GNNs naturally model data with intrinsic graph structures, avoiding the need to coerce non-grid data into tabular or image formats. This makes it ideal for fields like flood modeling, where irregular terrains and sensor networks are common. Unlike CNN-based models, which operate on regular grids, GNNs can work directly with particle systems (Shao et al., 2025), unstructured meshes (Acosta et al., 2026; Taghizadeh et al., 2025a), multi-scale graphs (Bentivoglio et al., 2022; Taghizadeh et al., 2025b), and heterogeneous graphs (Huang et al., 2025; Luo et al., 2024; Taghizadeh et al., 2025b). This enables seamless coupling with finite-volume or finite-element hydrodynamic models, efficient integration with irregular sensor networks, and built-in scalability to complex geographies. Compared with CNNs, GNNs are particularly advantageous when flood domains are naturally represented as irregular networks, particles, or unstructured meshes rather than raster grids. CNNs remain effective for gridded flood-map prediction, but their fixed local convolution kernels are less suited to representing non-Euclidean connectivity, heterogeneous sensor networks, and mesh-based hydraulic exchange. In contrast, GNNs can encode these relationships directly through graph edges, enabling more flexible representation of upstream–downstream interactions, irregular spatial dependencies, and multi-hop information transfer.

Its compatibility with unstructured data extends to the ability of GNNs to incorporate diverse data sources. Nodes can refer to arbitrarily defined regions rather than fixed spatial divisions, which allows inclusion of additional targeted features like human-sensed data (e.g., social

**Table 6**  
Physics-incorporation strategies used in Mesh-based GNN flood models.

| Model           | Physical Fidelity                   | Strategy                                                                                                                                                                                              |
|-----------------|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SWE-GNN         | Physics-constrained by architecture | Propagation rule mimics SWE mass and momentum fluxes; gradient term between nodes represents hydrodynamic flux; inspired by finite-volume methods to maintain physical consistency                    |
| FloodGNN-GRU    | Physics-informed by inputs          | Physics is indirectly captured via feature representations (e.g., treating velocities as vectors and using spatial adjacency in the graph)                                                            |
| mSWE-GNN        | Physics-constrained by architecture | Hydrodynamics-based propagation mimicking SWE; ghost cells to enforce boundary conditions; rotation-invariance improves physical generalization; optional mass conservation loss (tested in ablation) |
| HydroGraphNet   | Physics-informed by loss            | Mass conservation laws embedded into the loss function; physical consistency checks using ReLU-penalized physics-based loss terms                                                                     |
| MFGNN           | Physics-informed by inputs          | Nine statistical descriptors explaining key features of the input hyetograph and hydrograph are extracted                                                                                             |
| GNN-Transformer | Physics-informed by inputs          | Rainfall time series used for positional encoding; generated input from custom-made physics-based hydrodynamic model                                                                                  |
| DUALFloodGNN    | Physics-informed by loss            | Mass conservation laws embedded into the loss function; global constrains given boundary condition; local constrains given the flux at each node                                                      |

**Table 7**  
Graph structure details of Mesh-based GNN flood models.

| Model           | Node features                                                                                                                                                                                                                                                                                                                                               | Edge features                                                                                          | Graph type        | Graph size                                                 | Ground truth                   |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|-------------------|------------------------------------------------------------|--------------------------------|
| SWE-GNN         | <b>Static:</b> cell area, elevation, slope, Manning's n<br><b>Dynamic:</b> water depth, unit discharge                                                                                                                                                                                                                                                      | <b>Static:</b> normal vector, side length<br><b>Dynamic:</b> flux                                      | Structured grid   | ≈5000                                                      | Delft3D-FM                     |
| FloodGNN-GRU    | <b>Static:</b> elevation, Manning's n, distance to river<br><b>Dynamic:</b> rainfall, water depth, velocity                                                                                                                                                                                                                                                 | None                                                                                                   | Structured grid   | ≈10,000                                                    | LISFLOOD-FP                    |
| mSWE-GNN        | <b>Static:</b> cell area, elevation, Manning's n<br><b>Dynamic:</b> water depth, unit discharge                                                                                                                                                                                                                                                             | <b>Static:</b> side length<br><b>Dynamic:</b> flux                                                     | Structured grid   | ≈11,000                                                    | Delft3D-FM                     |
| HydroGraphNet   | <b>Static:</b> position, cell area, elevation, slope, aspect, curvature, Manning's n, flow accumulation, infiltration<br><b>Dynamic:</b> Water depth, water volume, rainfall, inflow                                                                                                                                                                        | <b>Static:</b> Euclidean distance, relative position                                                   | Unstructured mesh | 4787                                                       | HEC-RAS                        |
| MFGNN           | <b>Static:</b> position, elevation, slope, aspect, curvature, cell area, Manning's n, and infiltration rate<br><b>Dynamic:</b> peak time, median flow time, pre/post peak flow ratio, first half volume fraction, first third volume fraction, peak/total flow ratio, duration to peak ratio, peak to average flow ratio, total flow volume, event duration | <b>Static:</b> Euclidean distance, relative distance                                                   | Unstructured mesh | <b>Low-fidelity:</b> 2400<br><b>High-fidelity:</b> 158,800 | HEC-RAS                        |
| GNN-Transformer | <b>Static:</b> position, elevation, land use type, slope, aspect, curvature, and Manning's n<br><b>Dynamic:</b> water depth, velocity<br><b>Global:</b> rainfall positional encoding                                                                                                                                                                        | <b>Static:</b> feature-based weights                                                                   | Structured grid   | 1,435,271                                                  | Constructed hydrodynamic model |
| DUALFloodGNN    | <b>Static:</b> position, cell area, elevation, aspect, curvature, Manning's n, flow accumulation, infiltration<br><b>Dynamic:</b> Water volume, rainfall, inflow                                                                                                                                                                                            | <b>Static:</b> Euclidean distance, relative position, slope, face length<br><b>Dynamic:</b> Water flow | Unstructured mesh | 1129                                                       | HEC-RAS                        |

media, emergency calls) (Farahmand et al., 2023; Liang et al., 2024; Fang et al., 2025). On top of this, the graph structure itself can be modified to accommodate more information such as using heterographs to consolidate various sensor sources (J. Jiang et al., 2024) and embedding semantic relationships through a knowledge graph (Jin et al., 2026).

The richer representation provided by flexible graph input of GNN models enables spatiotemporal modeling within a unified framework. This is contrast to other DL approaches that separate spatial and temporal processing (e.g., CNN for space, RNN for time). For example,

flood GNN models additionally consider similarity of hydrologic properties and correlation in historic flood events when determining the weights of the edges within the graph (Farahmand et al., 2023; Zhao et al., 2020; Xue and Zhu, 2025). This enables capturing both spatial dependencies (e.g., upstream–downstream interactions) and temporal dynamics within a single architecture. While hybrid designs combining GNN layers with temporal units (e.g., LSTMs, GRUs, Transformers) remain common for performance gains, fully integrated spatiotemporal GNNs are increasingly effective (Bentivoglio et al., 2024; Taghizadeh et al., 2025b).

#### 4.1.2. Predictive performance

Throughout the reviewed literature, GNN-based models perform strongly on both flood depth regression and inundation extent classification. For continuous variables such as water depth, studies report low RMSE and MAE alongside high NSE values. NSE is a standard hydrological modeling metric that provides a normalized comparison between the model residual variance and the variance of the reference observations (Nash and Sutcliffe, 1970). For binary flood extent, studies report high Critical Success Index (CSI), defined as the ratio of correctly predicted flooded cells to the union of predicted and observed flooded cells (Donaldson et al., 1975). The reported scores are computed for validation and test events withheld from training. However, these evaluations typically measure agreement with hydrodynamic reference simulations rather than direct predictive accuracy against observed flood events. These highlight their ability to capture spatial dependencies and flow dynamics at a level comparable to physics-based solvers in complex riverine and urban systems. Furthermore, several studies have explored zero-shot transfer of trained models, though with mixed success (Bentivoglio et al., 2024; Garzón et al., 2026; Sun et al., 2022). While these early results are inconclusive, they point to a promising direction for developing generalizable GNN-based predictive models.

In most benchmark studies, flood-specific GNN architectures outperform other ML approaches, including LSTMs, GRUs (J. Jiang et al., 2024; Zhangke et al., 2026), and CNNs (Kazadi et al., 2024; Taghizadeh et al., 2025a), as well as early GNN baselines (Bentivoglio et al., 2023; A.Y. Sun et al., 2021). This demonstrates synergistic gains of incorporating temporal modules or physics-informed techniques for autoregressive spatiotemporal use cases. In addition, performance improvement can also be attributed to the robustness of the message-passing operation where GNNs can model both local dependencies (immediate neighbors) and global structures (multi-hop interactions). Therefore, GNNs have shown better performances in capturing spatial correlations and long-range effects compared to CNNs and RNNs in flood modeling.

#### 4.1.3. Computational efficiency

GNN-based flood models achieve substantial inference speedups relative to traditional hydrodynamic solvers. For instance, SWE-GNN (Bentivoglio et al., 2023) and mSWE-GNN (Bentivoglio et al., 2024) report 100–700× faster inference than high-resolution solvers, while FloodGNN-GRU (Kazadi et al., 2024) achieves up to 1000×. These accelerations, on top of their high predictive performance, make near-real-time flood modeling and rapid scenario testing feasible, especially in probabilistic and operational forecasting contexts. However, this benefit primarily applies to computation at inference time and does not necessarily carry over to the complete training pipeline, including dataset generation, training epochs and hyperparameter tuning.

With regards to model size, GNNs are parameter-efficient because they reuse weights in graph convolution and exploit relationship bias within the data (Battaglia et al., 2018). This reduces the total number of parameters compared to dense models like MLPs and RNNs. In contrast with diffusion models, Variational Auto Encoders (VAEs), and CNNs, GNN models typically require less number of layers for effective training, further reducing their parameter count. These traits reduce training time, resource demand, and data requirements.

The computational efficiency of GNNs also extends dealing with larger domains or fine-grained resolutions as GNNs can inherently scale with graph size. To handle such networks, sampling neighborhoods within the graph can be performed without losing inductive learning capabilities, as shown in GraphSAGE (Hamilton et al., 2018). This, combined with traditional scaling techniques, can be used to train on larger datasets incrementally.

#### 4.1.4. Physical grounding

The flexibility provided by the message passing operation provides a natural framework for embedding physical laws. This renders GNNs highly customizable to integrating physics and domain knowledge through hydrodynamic-driven architectures (e.g., SWE-inspired fluxes Bentivoglio et al., 2023, source-to-sink relationships S. Liu et al., 2026), or loss terms enforcing conservation principles (e.g., mass conservation Acosta et al., 2026; Taghizadeh et al., 2025a; Zhang et al., 2024). This nature improves physical fidelity, reduces reliance on huge datasets, and provides rationale behind predictions.

Interpretability is a key component of flood models to provide insight behind model predictions. To this end, graph structures make it easier to interpret influence paths to see which nodes or features affect the prediction the most. They offer an intuitive high-level understanding through graph visualization and attention edge weights (Chaimook et al., 2025). For more thorough analysis, tools such as SHAP analysis (Guo et al., 2025; He et al., 2025; Victor et al., 2023) have been used to identify the impact of features. This supports trust and understanding in scientific and decision-making applications, like flood modeling.

#### 4.2. Research gaps and limitations

While GNN-based flood models demonstrate significant potential, several research gaps and limitations remain before they can match the robustness, generality, and physical fidelity of established hydrodynamic modeling frameworks. Key limitations identified in the literature are outlined in this section.

##### 4.2.1. Hydraulic and hydrodynamic consistency

Data-driven GNNs have yet to adopt concepts from physics-based modeling to achieve the physical realism of numerical models. For example, despite progress in physics-informed GNN design, most current models only partially integrate the conservation laws from the SWE.

- **Mass Conservation.** Existing approaches enforce global mass balance either explicitly via loss terms (Taghizadeh et al., 2025a) or implicitly through message passing (Bentivoglio et al., 2023). However, satisfying global conservation does not guarantee local node-level conservation, which can still be violated in predictions. Among the reviewed studies, only the recent DUALFlood-GNN framework explicitly enforces local mass conservation at the individual cell level (Acosta et al., 2026).
- **Momentum Conservation.** No reviewed GNN-based flood model explicitly enforces the full momentum balance from the SWE. Models like SWE-GNN (Bentivoglio et al., 2023) and mSWE-GNN (Bentivoglio et al., 2024) approximate momentum flux exchange through hydrodynamic-gradient-inspired message passing, but these are indirect and do not guarantee strict momentum conservation. This gap limits physical fidelity, especially for modeling complex behaviors such as rapid flows, backwater effects, or flow reversals.
- **Energy Loss.** Empirical friction laws (e.g., Manning's, Chézy) and local structure-related losses are standard in 1D/2D hydrodynamic models but absent from current flood GNNs. Some models include Manning's coefficient as a static feature, yet no explicit friction slope or head-loss terms are present in their computations. This omission limits the ability of GNN models to realistically model flow attenuation and backwater effects, especially in urban catchments with complex infrastructure. Future models could mimic hydrodynamic solvers more closely by integrating explicit energy loss terms into message passing or loss functions and by representing hydraulic structures (e.g., culverts, gates) as dedicated graph components with head-loss dynamics.

Opportunities also exist to integrate methodologies that have proven effective independently, but not yet in combination. Current improvements rely on either physics-based architectures or hybrid DL modules (e.g., GRUs, LSTMs). However, explicitly integrating these approaches within a single architecture remains comparatively underexplored. Combining temporal modules with physics-informed designs could improve both predictive accuracy and physical consistency, though training complexity may increase. In terms of spatial dimension, all reviewed GNN-based flood models represent either purely 1D networks (e.g., rivers or drainage systems) or purely 2D domains (e.g., floodplains and urban grids). In contrast, more realistic numerical flood models often couple 1D elements, such as levees, culverts, and pipes, with 2D domains, particularly in urban flood simulations. This highlights an opportunity to develop 1D–2D coupled GNN surrogates capable of supporting rapid, large-scale flood modeling.

At large scale, the main limitation of graph-based flood models is that memory and computational demands increase with the number of nodes and edges, making operational-resolution flood domains difficult to represent directly. Further, current GNNs have yet to mimic realistic hydrodynamic model resolutions. Operational hydrodynamic models often require 50,000+ nodes, but most flood GNN networks are much smaller due to memory constraints. Neighborhood-sampling methods such as GraphSAGE (Hamilton et al., 2018) can reduce memory load, but may weaken long-range dependency capture. Scaling to realistic resolutions without sacrificing accuracy remains an open challenge. As the graph size increases, hardware constraints emerge due to the corresponding increase in memory requirements. Distributed GNN training, which no reviewed work has implemented, can be used to partition graphs for parallel processing across multiple workers (Shao et al., 2023). Leveraging such frameworks could even enable national-scale or continental-scale flood modeling.

#### 4.2.2. Modeling scope

There remains significant potential to increase the problem scope of current surrogate modeling research to address more complex challenges such as longer lead times. GNNs, like other DL models, face challenges in long-term flood forecasting due to error accumulation for extended timeframes. For autoregressive models, this is caused by the use of the previous model output as input for the next timestep, with noise gradually increasing after each prediction. For non-autoregressive sequence-to-sequence models, errors can arise because the dependence between the input sequence and more distant future timesteps weakens with lead time, causing predictions to drift toward the mean behavior of the target sequence. In rapidly evolving flood scenarios, where predictions must be generated at high temporal frequency, model stability over long lead times becomes essential to ensure reliable high-resolution forecasts. While recent approaches have introduced physics-informed constraints to improve stability, their effectiveness across complex, real-world flood settings remains limited. For instance, SWE-GNN (Bentivoglio et al., 2022) demonstrated relatively stable roll-outs of up to 240 steps, but its evaluation was restricted to simplified constant single-inflow scenarios. Future research directions include the integration of more advanced physics-informed techniques and the exploration of alternative architectures to extend prediction horizons and improve robustness. For example, graph-transformer-style architectures can introduce self-attention across the temporal axis for each node, enabling global connections between highly related timesteps and supporting longer-range temporal dependency learning.

Another insufficiently examined topic is generalization across different catchment and flood types. Section 3 illustrated that studies could be cleanly segregated according to their application type since they typically deal with only a single flood type (e.g., riverine, pluvial, or coastal). Additionally, only a limited number of studies consider evaluation across multiple catchments (Bentivoglio et al., 2023, 2024; Garzón et al., 2026; S. Liu et al., 2026). Broader cross-catchment, multi-flood-type testing is needed to assess transferability and guide model selection for diverse environments.

GNN flood models can also be evaluated for effectiveness in the real world. However, real-world flood datasets are often sparse and may contain faulty or missing measurements (Wright, 2018). These factors directly affect GNN performance, given that, as a DL architecture, GNNs are inherently data-intensive. While simple interpolation methods have been explored (Farahmand et al., 2023; Zhao et al., 2020), these approaches cannot fully capture the complex spatiotemporal dynamics of flood propagation. Integrating GNNs with more advanced data assimilation or reconstruction techniques could enhance model robustness and predictive accuracy in data-scarce environments. Of all the reviewed studies, only Sui et al. directly tackles this problem (Sui et al., 2026) by dynamically generating edges from streamflow data and using spatiotemporal message passing to bypass missing values.

#### 4.2.3. Network representation

The representational power of the graph structure leaves much opportunity to embed context to inform future-state prediction. Most of the reviewed models rely primarily on hydrometeorological and catchment variables such as DEM, rainfall, and land use. However, it is possible to incorporate a wider range of features, including infiltration, soil moisture, infrastructure, urban texture (e.g., building height and density), and climate indices. Additionally, graph-based architectures can naturally integrate such information from GIS datasets. Additional relevant information enhances flood prediction accuracy with minimal cost due to the ability of DL models to automatically extract core features.

Most GNN-based flood models focus exclusively on node-level predictions. While this approach is physically meaningful for variables such as water depth, level, and volume, even models that predict discharges and velocities often do so at the node level. This requires additional mapping and reinterpretation of these quantities, which can introduce uncertainties. However, in hydrodynamics, velocities, discharges, and fluxes are naturally associated with cell interfaces (edges). Explicit edge prediction, as is common in other physics-informed GNN domains, could yield more physically consistent outputs and simplify the integration of conservation laws. Acosta et al. pilot this approach through their DUALFloodGNN model which simultaneously predicts water volume at nodes and water flow at edges (Acosta et al., 2026).

#### 4.2.4. Predictive transparency

As mentioned previously, interpretability is important in flood modeling to gain a deeper understanding of the core principles that govern this physical process. Although some studies have addressed model interpretability using techniques such as SHAP analysis (Guo et al., 2025; He et al., 2025; Victor et al., 2023), integrated gradients (A.Y. Sun et al., 2021) and Kolmogorov–Arnold Network (KAN) encoders (Taghizadeh et al., 2025a), the integration of explainable AI (XAI) tools in GNN-based flood modeling can be further improved. For example, GNNExplainer (Ying et al., 2019), a model-agnostic method specifically designed for GNNs, has not yet been applied to interpret flood models. The use of multiple tools in conjunction also warrants further exploration as this can be used to validate XAI accuracy and discover the most meaningful approaches. Moreover, the insights obtained from existing interpretability studies have not been systematically collated or used to guide practical model improvements. Broader adoption of XAI could help identify spurious correlations, enhance robustness, and inform targeted model refinement.

Current approaches that utilize uncertainty quantification are only limited to a few through Monte Carlo dropout (Nguyen et al., 2026), bootstrap sampling (Charania and Patel, 2026) and Bayesian inference (Liu et al., 2023). Techniques such as Bayesian GNNs (Elinas et al., 2020; Stadler et al., 2021; Zhang et al., 2018) or deep ensembles (K. Sun et al., 2021; Wei et al., 2023) could further enhance this capability, essential for operational decision-making. Without calibrated uncertainty estimates, deterministic GNN predictions may appear overconfident or underestimate flood risk, particularly for extreme

events or domains that differ from the training data. Future work should therefore assess uncertainty calibration using probabilistic metrics or prediction-interval coverage, and develop models that provide calibrated uncertainty bounds rather than only point predictions.

#### 4.2.5. Evaluation rigor

Evaluation practice within the domain of GNN flood modeling can be further improved to provide greater insight to future direction. For example, benchmark comparisons are typically made against traditional ML methods or generic GNN baselines rather than specialized flood GNNs from related literature. This limits insight into architectural trade-offs and makes it difficult to identify the most effective designs. Public benchmark datasets for flood GNN evaluation would enable more rigorous, standardized comparisons. While this comparison may be constrained by a model's supported datasets, the taxonomy provided in Section 3 provides a catalog of benchmark GNNs for specific application types.

Proper ablation studies must also be established as a research standard in this domain. Of all the studies reviewed in this survey, only a handful conduct an ablation analysis of their proposed model (Bentivoglio et al., 2024; Chaimook et al., 2025; Huang et al., 2025; Kazadi et al., 2024; J. Jiang et al., 2024; Zhou et al., 2025). This common omission impedes a clear understanding of the primary drivers of model performance, which is especially pronounced for emerging physics-informed approaches.

#### 4.3. Critical synthesis

Given the discussions from the previous sections, GNNs have already demonstrated their feasibility and efficiency as hydrodynamic-solver surrogates. They achieve strong predictive accuracy in localized catchments, reporting low RMSE and high NSE for depth regression and high CSI for inundation extent, at a level comparable to the physics-based solvers (Charania and Patel, 2026; Huang et al., 2025; Kilinc et al., 2025; Taghizadeh et al., 2025a). These are delivered with significant inference speedups (Bentivoglio et al., 2023, 2024; Kazadi et al., 2024), all within a representational framework flexible enough to ingest unstructured meshes, heterogeneous sensor data, and physical constraints. The central challenge now lies in translating these capabilities into real-world operationalization. There are three main prerequisites that this rests upon: raising the fidelity of the training signal, tailoring the model to real-world conditions, and establishing the foundations for iterative improvement.

The first requirement is raising the fidelity of the training signal, where the approach is primarily dependent on the utilized data source. For data that comes from observed events, the sparse and faulty nature of real flood measurements must be addressed. Current approaches have only addressed this through simple interpolation methods (Farahmand et al., 2023; Zhao et al., 2020) or reconstruction from implicit streamflow relationships (Sui et al., 2026). Capturing the complex spatiotemporal dynamics of real events also calls for richer temporal modeling. Hybrid architectures coupling GNNs with recurrent units have been extensively utilized, but other components such as Transformers and 1D convolutions remain comparatively underexplored and warrant further investigation. For synthetic data derived from numerical models, the GNN surrogate must inherit more of the solver's physics than it currently does, particularly if they are intended to replace or substantially accelerate time-critical components of numerical modeling workflows. Mass conservation is the most enforced (Bentivoglio et al., 2023, 2024; Taghizadeh et al., 2025a), but must further be refined to influence local hydrodynamics as well (Acosta et al., 2026). Momentum is at best approximated through the gradient-inspired message passing (Bentivoglio et al., 2023, 2024) rather than conserved. Energy loss is absent entirely, with Manning's coefficient appearing as a static feature but no explicit friction-slope constraints. How to

embed these higher-order constraints remains an open question, not a demonstrated capability.

The second requirement is tailoring the model to real-world conditions along two dimensions. Scalability is the more immediate as operational domains routinely demand 50,000+ nodes, whereas most reviewed models only operate on a few thousand. While there are established methods to scale GNN input (to be discussed in Section 5.2), these remain unimplemented in the flood modeling domain. Generalization is the more complex axis. Models are typically demonstrated on single catchments and flood types, and the transfer learning remains limited to a handful of studies. Current models generalize reliably only when source and target domains share similar input distributions (Garzón et al., 2024b). Zero-shot transfer degrades sharply across structurally distinct catchments, though fine-tuning can restore near-baseline accuracy (Bentivoglio et al., 2024; Fang et al., 2025; Garzón et al., 2026). Robustness across varying flood categories, boundary conditions, and mesh resolutions remains unrealized.

The third requirement is the foundations for iterative improvement. Explainability has already gained traction with SHAP and integrated gradients being utilized (Guo et al., 2025; He et al., 2025; Victor et al., 2023), but GNN-specific tools such as GNNExplainer (Ying et al., 2019) remain unapplied. Furthermore, ablation studies have only been conducted in a handful of reviewed works (Bentivoglio et al., 2024; Chaimook et al., 2025; Huang et al., 2025; Kazadi et al., 2024; J. Jiang et al., 2024; Zhou et al., 2025). Limited insight into a model's predictions leaves practitioners without a systematic account of why models work. Uncertainty quantification is limited to a few models (Charania and Patel, 2026; Liu et al., 2023; Nguyen et al., 2026). This is indispensable once predictions inform real decisions. Lastly, evaluation practice offers no standard datasets or protocols, with comparisons drawn against generic baselines rather than specialized flood GNNs.

Taken together, GNN flood models are ready today for rapid scenario screening, but operational forecasting awaits closing this operationalization gap. The limitations are matters of engineering and evaluation practice rather than theoretical limits, which makes the outlook favorable and the priorities clear.

### 5. Opportunities from related domains

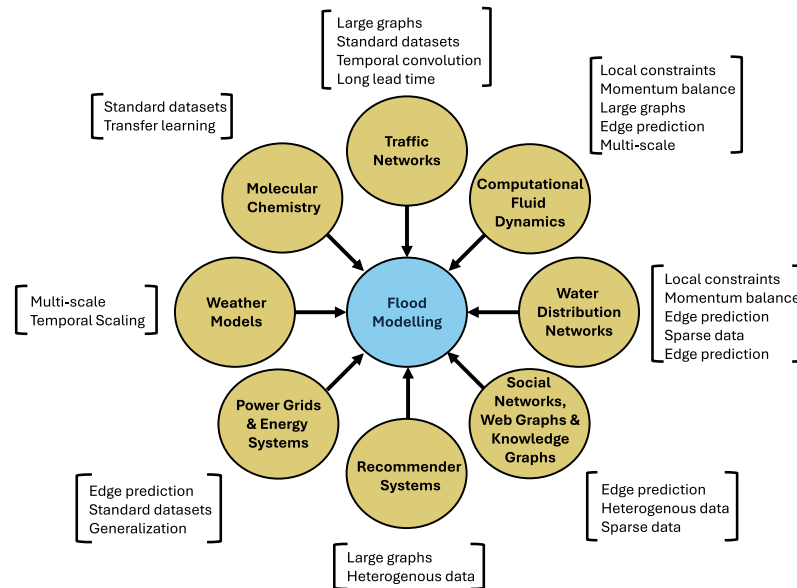
Although the use of GNNs in flood modeling is still in its early stages, these models are well established and extensively explored in related domains. GNNs have excelled in domains that share key characteristics with flood modeling: networked data (spatial or relational), temporal evolution (dynamic graphs), and complex, multimodal inputs. Success in these domains supports the suitability of GNNs for flood modeling and points to numerous transferable techniques, such as dynamic graph learning, multi-relational edge modeling, and distributed graph training, that could be adapted to overcome current limitations. The challenge is not in theoretical feasibility but in adapting and implementing existing GNN capabilities, including scalability, physics integration, and rigorous benchmarking, for flood mapping applications. Table 8 summarizes relevant fields and highlights the strengths of GNNs within each domain, while Fig. 6 illustrates specific techniques from these disciplines that could be adapted for flood modeling. The figure highlights eight source fields, each paired with associated techniques that offer potential improvements when integrated into GNN-based flood models.

#### 5.1. Physics incorporation

Water distribution systems (WDS) and computational fluid dynamics (CFD) are closely related to flood modeling, as all describe the movement of water under physical laws, albeit in different contexts. Physics-informed GNN techniques from these domains can be adapted to floods, since they are governed by similar underlying principles.

**Table 8**  
Fields with established GNN applications and reasons for their effectiveness.

| Field                                          | Why GNNs Excel                                                                                                                                             |
|------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Traffic networks                               | Naturally represented as spatiotemporal graphs (nodes = intersections, edges = roads), enabling joint modeling of network structure and temporal dynamics. |
| Power grids & Energy systems                   | Graph structure (nodes = substations, edges = transmission lines) supports modeling of complex dependencies and cascading failures.                        |
| Molecular chemistry                            | Molecules are intrinsic graphs (atoms and bonds), allowing GNNs to capture local chemical interactions and bond-specific effects.                          |
| Social networks, Web graphs & Knowledge graphs | Learn node and edge embeddings from complex, dynamic, and heterogeneous interaction data.                                                                  |
| Recommender systems                            | Model user–item interaction graphs to capture both local preferences and global patterns.                                                                  |
| Computational fluid dynamics                   | Handle unstructured meshes or particle systems, capturing spatiotemporal relationships and incorporating physics-informed constraints.                     |
| Water distribution networks                    | Naturally modeled as graphs (nodes = junctions, edges = pipes), enabling integration of hydraulic physics into learning.                                   |
| Weather models                                 | Represent large-scale atmospheric states as spatiotemporal graphs, enabling automatic extraction of diverse spatial and temporal features.                 |



**Fig. 6.** Techniques for enhancing flood modeling with GNNs, inspired by diverse application domains.

Current gaps in flood GNNs, such as enforcing local constraints and momentum conservation, could be addressed by directly solving governing equations, using predictor–corrector schemes, or integrating simplified physical formulations.

- **Direct solvers.** Most physics-informed flood GNNs only guide learning implicitly through hydraulic principles, without directly embedding governing equations. In CFD, success has been achieved by integrating the Navier–Stokes (NS) equations into the architecture. The GNN then acts as a function approximator for the PDE solution, mimicking numerical solvers. The outputs of GNN layers can be interpreted as higher-order derivatives of the input states. These derivatives are then integrated (e.g., via a numerical integration scheme) to predict future states. Some approaches embed a coarse PDE solver into the network (Belbute-Peres et al., 2020) or approximate the differential operator with a neural nonlinear solver (Horie and Mitsume, 2023). Others apply the NS equations directly as a physics-informed loss, enforcing momentum conservation and local constraints at the cell level (He

et al., 2022; T. Li et al., 2024). Raissi et al. (2019) demonstrate how to derive such loss functions for NS, which could be adapted for the SWE. This approach increases implementation complexity, as it requires direct familiarity with the governing PDEs of the system under consideration.

- **Predictor–corrector.** The predictor–corrector approach, used in both CFD and WDS, generates an initial prediction that is subsequently adjusted to satisfy physical laws. For example, the Fluid Graph Network (FGN) (Li and Farimani, 2022) predicts particle acceleration based on initial state and external force, then applies two corrections: momentum from particle collisions and pressure from fluid compression. In WDS, Ashraf et al. (2024) correct a GCN's predictions using a custom physics-informed module inspired by message passing, enforcing flow conservation and pressure–flow relationships. Message-passing-based correction can enforce local constraints by decomposing the problem into simpler intermediary tasks. However, this approach also increases data requirements, as ground truth must be additionally collected

for each sub model. This also decreases inference speed given that more architectural components lead to more processing time.

- **Simplified equations.** WDS models often use the Hazen–Williams equation, an empirical formula for pipe flow at given temperatures (Christensen et al., 2000), which is straightforward to integrate into architectures or loss functions (Hajgató et al., 2021; Xing and Sela, 2022). Flood GNNs could adopt a similar strategy, for instance by using the 1D S-V equations for channel flow instead of the full SWE, especially in targeted applications. Such reduced formulations could simplify integration in direct solver methods. This approach has already been adopted in SWDS, as it shares the same structural composition with WDS, but remain largely unexplored in other branches of flood modeling. However, relying on a simplified equation may limit the model’s general applicability, as such equations are typically valid only in specific scenarios.
- **Hybrid super-resolution.** Hybrid numerical–ML super-resolution is another promising direction for physics incorporation in GNN-based flood modeling. Existing non-GNN hybrid approaches have shown that coarse-resolution hydrodynamic simulations can be corrected or upscaled to approximate high-resolution inundation behavior (Fraehr et al., 2023, 2024; Herath et al., 2025; Neo et al., 2026). Although GNN-based super-resolution flood mapping was not identified within the reviewed GNN flood-modeling literature, related work in CFD has demonstrated mesh-based super-resolution of fluid flows using multiscale GNNs (Barwey et al., 2025). This suggests a promising opportunity for flood modeling, where GNNs could act as correction operators that map coarse hydrodynamic states to fine-scale water depth, velocity, or discharge fields while preserving hydraulic connectivity and physical consistency on unstructured or multi-resolution meshes. However, hybrid super-resolution approaches still require a coarse or low-fidelity hydrodynamic simulation to be run before ML-based correction, meaning their inference speed is constrained by the underlying physics-based model. They may also inherit biases from the coarse model and require paired coarse- and fine-resolution simulation data for training.

Overall, these examples suggest that physical constraints can be integrated into GNN-based flood models at different levels of directness, ranging from soft guidance through physics-informed inputs or losses to stronger enforcement through solver-inspired architectures, predictor–corrector modules, simplified governing equations, or hybrid super-resolution workflows. For flood mapping specifically, these strategies could improve the physical consistency of predicted inundation extent, water depth, and velocity fields by enforcing local mass and momentum relationships between neighboring cells, drainage links, or mesh elements, thereby reducing non-physical artifacts in mapped flood extents. However, the appropriate level of constraint depends on the target application, since stronger physics integration can improve consistency but may also increase implementation complexity, data requirements, and computational cost.

## 5.2. Handling large graphs

Most current flood GNN models operate on graphs with only a few thousand nodes, yet other domains routinely scale to hundreds of millions or even billions of nodes. While no theoretical barrier prevents similar scaling for flood modeling, practical deployment requires established techniques such as sampling and distributed processing, adapted to the domain’s spatiotemporal characteristics.

- **Sampling.** Sampling subgraphs can reduce memory and compute requirements while preserving accuracy and generalizability. Rather than strictly improving performance, sampling methods extend the capabilities of GNNs, enabling them to train on larger

graphs and deeper architectures that would otherwise exceed available computational resources. GraphSAGE (Hamilton et al., 2018) pioneered neighborhood sampling, but deeper GNNs can face the “neighborhood explosion” problem (Chiang et al., 2019) due to exponential growth in sampled neighbors. To mitigate this, sampling can be decoupled from the forward pass and instead be done as a pre-processing step, using methods such as clustering-based partitioning (Cluster-GCN Chiang et al., 2019) or bias-corrected sampling (GraphSAINT Zeng et al., 2020).

Sampling strategies can also be domain-aware. In fluid dynamics, Peng et al. (2022) sample non-uniformly, prioritizing turbulent regions requiring higher resolution. Traffic forecasting studies have explored multi-level k-way partitioning (Mallick et al., 2020) and grouping by overall cumulative traffic count (Wang et al., 2018). In recommendation systems, PinSage (Ying et al., 2018) uses short random walks for importance sampling, which also informs message aggregation. For flood modeling, sampling could similarly prioritize flood prone areas or regions with high water level variability. One example of domain-aware sampling in flood modeling is provided by Sun et al. who sample a graph by selecting pour points (the lowest drainage outlets) corresponding to sub-basins, rather than using the full high-resolution river network (Sun et al., 2022). This approach may serve as a starting point for defining further sampling techniques, as a standard has not yet been established within the flood modeling domain.

- **Distributed training.** Distributed training has been applied in various DL applications to facilitate high-speed training on massive datasets. Translating this to the context of GNNs is not as straightforward due to embedding dependence on neighboring nodes and workload imbalance from naive graph partition (Shao et al., 2023). Public frameworks such as AliGraph (Zhu et al., 2019), DGL (M. Wang et al., 2020), and PyTorch Geometric (Fey and Lenssen, 2019) provide tooling for distributed graph training, reducing implementation complexity. PinSage leverages these principles to train on graphs with over 3 billion nodes (Ying et al., 2018), offering a blueprint for adapting large-scale training techniques to flood modeling. PinSage also demonstrated that such scalability can be achieved without sacrificing model quality: it trained on 7.5 billion examples over a graph with 3 billion nodes and 18 billion edges, achieved 67% hit-rate and 0.59 Mean Reciprocal Rank (MRR), and improved over the best baseline by 150% in hit-rate and 60% in MRR (Ying et al., 2018). Its efficiency was further supported by subgraph-based training, which reduced runtime by a factor of six with negligible performance loss, and a MapReduce inference pipeline that generated embeddings for 3 billion items in less than 24 h (Ying et al., 2018).

Graph partitioning presents a particular challenge in flood modeling because it is coupled with the physics of the domain. Partitions should remove as few edges as possible to limit inter-worker communication, while avoiding active flow regions where removing an edge could disrupt mass exchange constraints. This suggests partitioning along natural hydraulic boundaries, such as sub-basin divides or catchment outlets, where flow coupling is weakest. Boundary consistency can then be maintained by exchanging halo nodes between workers, analogous to ghost cells used to represent boundary conditions in numerical solvers. Overlapping or sliding-window-style subgraph sampling may also be useful, because edges cut in one partition can be included in adjacent samples or represented through halo nodes. This would help preserve hydraulic exchange pathways across partition boundaries while still reducing memory demand.

## 5.3. Simultaneous node & edge predictions

In many domains where node and edge properties are tightly coupled, GNNs have been designed to update both node and edge embeddings simultaneously. Li et al. describe fluid motion at a particle-level

with both node-focused networks (for advection and pressure) and edge-focused networks (for collision) (Li and Farimani, 2022). Node-focused networks primarily update and predict quantities stored at graph nodes, using edges mainly as pathways for message passing. In contrast, edge-focused networks explicitly compute and update edge embeddings, enabling predictions for edge-level quantities such as flux, velocity, or discharge. These are connected in series and perform message passing to update node and edge embeddings respectively. In WDS, Ashraf et al. (2024) update edge embeddings directly from the messages exchanged between connected nodes at each message passing step.

Power grid studies have leveraged similar coupling through dual representation learning, using diffusion-based feature propagation (Lin and Sun, 2022) and joint multi-headed attention (Zhao et al., 2025). In knowledge graphs, a GCN encoder within an autoencoder architecture enables the same model to support both node classification and link prediction (Schlichtkrull et al., 2017). These strategies can be adapted for multi-task regression in flood modeling, allowing a single model to output predictions for both nodes and edges as initially explored by Acosta et al. (2026). Simultaneously predicting node and edge hydrodynamic features must be implemented with a mass conservation constraint, improving physical fidelity and capturing water transfer dynamics. Without this, the node and edge prediction heads can produce misaligned outputs, resulting in unstable and diverging rollouts.

#### 5.4. Inter-GNN benchmarks

There is currently no widely adopted standardized benchmark specifically for evaluating flood GNN models. Studies often use custom datasets with varying features, which makes it difficult to consolidate the evaluation of models. For physics-informed approaches, this challenge is greater because some architectures or loss functions require specialized hydrodynamic measurements that are not widely used.

Previous meta-analyses have established general criteria and benchmarks for flood mapping methods (El Moumen et al., 2026). However, recent advancements in GNNs provide new opportunities to address limitations identified in traditional approaches. More recently, the flood modeling community has begun to address benchmark development through datasets such as FloodCastBench for large-scale 2D hydrodynamic flood modeling and forecasting (Xu et al., 2025), and UrbanFloodBench for coupled 1D-2D urban flood surrogate modeling (Lim et al., 2026a,b). These datasets represent important steps toward standardized flood-model benchmarking. However, they remain recent efforts, and, to the best of our knowledge, the GNN-based flood modeling studies reviewed in this paper have not yet adopted them for systematic inter-GNN comparison. Therefore, there remains a need for GNN-oriented benchmark protocols that define not only shared datasets, but also standard graph features, target variables, data splits, evaluation metrics, computational-efficiency measures, and reproducibility requirements.

Other domains have benefited greatly from common benchmark datasets and task definitions, such as METR-LA for traffic prediction (Jagadish et al., 2014), MoleculeNet (Wu et al., 2018) and PubChem (S. Kim et al., 2024) for molecular chemistry, PowerGraph (Varbella et al., 2024) for power grid systems, and AirfRANS (Bonnet et al., 2023) for CFD. These examples show how shared datasets and evaluation protocols can support reproducible model comparison, and provide useful guidance for developing equivalent inter-GNN benchmarks for flood modeling.

The value of an equivalent benchmark for GNN-based flood modeling depends on the criteria along which models are compared. Several such criteria can be inherited from existing flood-mapping guidelines. El Moumen et al. (2026) identify five axes that govern predictive accuracy: the applied method, topographic type, spatial extent, reference dataset size, and the selection of conditioning factors. We propose that a benchmark be defined along four criteria that build on these

recommendations. First, a standardized minimum set of node and edge features, grounded in prevalent conditioning factors such as distance from the river network, elevation, and slope (El Moumen et al., 2026). This standardization extends to the target dynamic features, which could be water level or depth for nodes and velocity for edges. Second, high-fidelity reference ground truth for each section of the catchment, whether from hydrodynamic solvers or gauged observations. Third, fixed training, validation, and out-of-distribution test splits that assess generalizability across diverse flood scenarios, spanning flood types (e.g., riverine, urban, coastal compound, dike breach), spatial scales (from regional to national), and temporal resolutions, rather than random splits. This mirrors the scaffold splits used in MoleculeNet (Wu et al., 2018) to test generalization deliberately. Fourth, a defined suite of evaluation metrics should include predictive accuracy, physical consistency, temporal stability, computational efficiency, and reproducibility. Computational efficiency should include inference time, memory use, and speed-up relative to numerical solvers, while reproducibility should include clear reporting of training time, data-generation requirements, model inputs, and code/data availability. AirfRANS (Bonnet et al., 2023) demonstrates that physically meaningful evaluation requires more than aggregate field error. Aside from established performance metrics (e.g., RMSE, MAE, NSE and CSI), criteria could be extended with two requirements specific to dynamic modeling: physical consistency, quantified through mass conservation error, and temporal stability, quantified through error growth over long rollouts.

#### 5.5. Generalization & transfer learning

A key advantage of GNNs over many other DL frameworks is their ability to operate on unstructured graph data with arbitrary numbers of nodes and edges. This opens the possibility of testing on graphs with a different structure from the training set. In the power grid domain, for example, GNNs trained on small networks have demonstrated promising performance on much larger, unseen test systems without retraining (Nauck et al., 2022, 2023). Bentivoglio et al. explored a similar idea for flood modeling but tested only on a small set of cases with limited variation in graph size (Bentivoglio et al., 2023, 2024). Demonstrating that flood GNNs can naturally generalize to higher spatial resolutions would greatly reduce data requirements and training time, as smaller graphs could suffice for training.

When generalization is insufficient, transfer learning can bridge the gap. Transfer learning reuses a pretrained model for a related task, reducing data needs and training time (Zhuang et al., 2021). The application of this technique in flood modeling has only been adopted by a few studies and still remains underdeveloped. For instance, generating a large-scale dataset in flood modeling is constrained by the processing speed of numerical simulators. In molecular chemistry, data scarcity for specific molecules is often addressed by pretraining on related but more abundant datasets. Lee et al. (2021) first trained models on short oligomers, which can easily be synthetically generated, before fine-tuning on longer, more complex molecules. Xiong et al. (2024) found that pretraining on a different but correlated molecular property also improves accuracy.

The same principle can be applied to flood modeling: pretraining a base GNN model on one flood type, catchment, or boundary condition and fine-tuning for another could improve performance and reduce overhead. How well such transfer succeeds, however, depends on the degree of similarity between the source and target domains, spanning graph construction, the rainfall and boundary-condition distributions, topographic likeness, hydraulic structures, and roughness parameter ranges. More fundamentally, it depends on whether the model has captured genuine physical relationships rather than site-specific correlations, which governs how far its learned behavior extends beyond the training domain. Such a model could serve as the foundation for cross-regional, multi-flood-type benchmarking, supporting broader claims of generalizability as discussed in Section 5.4. Buterez et al. (2024) provide a taxonomy of transfer learning strategies for GNNs that could guide such efforts.

### 5.6. Encode diverse node & edge features

Flood events are comprised of a rich set of static and dynamic features which can all be taken into account for model predictions. Current GNN flood models have begun exploring this through heterogeneous message passing (J. Jiang et al., 2024; Luo et al., 2024) and multi-scale graph architectures (Bentivoglio et al., 2024), but related domains offer more mature techniques that could be adapted.

- **Heterogeneous graphs for richer relationships.** Heterographs can represent multiple node and edge types, enabling models to capture diverse relationships within a single graph. In flood modeling, heterographs have been used to distinguish between rainfall and runoff nodes (Luo et al., 2024) and to integrate different data sources (J. Jiang et al., 2024). Other domains demonstrate the full potential of this approach. For example, CompGCN for knowledge graphs utilize entity-relation composition operators to encode different node and edge types, learning separate weights for each relationship type (Vashishth et al., 2020). HetSAGE, a recommender system model, defines multiple edge types between a single node type and applies type-specific attention during aggregation (Zheng et al., 2021). While such flexibility can boost expressiveness, it also risks overparameterization, making regularization essential to avoid overfitting (Schlichtkrull et al., 2017). In flood modeling, heterogeneous GNNs could represent more complex structures, such as river networks, flood defenses, and infrastructure interdependencies, particularly useful for urban flooding scenarios.
- **Multi-scale graphs for long-range dependencies.** Message passing in standard GNNs propagates information locally, so capturing long-range dependencies requires many layers, increasing computational cost and training instability. Multi-scale graphs address this by enabling information flow over large distances with fewer layers. In flood modeling, mSWE-GNN applies a U-Net-inspired design, using mean pooling for downsampling and learned parameters for upsampling (Bentivoglio et al., 2024), an approach adapted from CFD modeling (Lino et al., 2022). Additional techniques from fluid dynamics include top- $k$  pooling (Barwey et al., 2023) and smoothing layers (Chen et al., 2021) to enhance stability and efficiency. GraphCast, a GNN-based weather model, takes a different route. It first projects the input graph onto a fixed icosahedral grid, enabling uniform multi-resolution message passing before combining outputs in the decoder (Lam et al., 2023). This uniform encoding also facilitates generalization across different graph structures, a potential advantage for flood modeling across diverse regions.

These two techniques are not mutually exclusive. Combining heterogeneous and multi-scale graph representations could yield richer and more informative representations of flood events by capturing both diverse node/edge types and interactions across multiple spatial scales. One caveat is that while rich node and edge features can improve physical representation, they can also increase overfitting risk, especially when training events or catchments are limited. Feature selection, regularization, cross-catchment testing, and ablation studies are therefore necessary to ensure generalization.

### 5.7. Handling temporal data

Flood evolution is inherently dynamic, requiring models to capture a variety of temporal dependencies on top of spatial relationships. Flash floods develop over a few hours, driven by intense localized rainfall causing sharp, rapidly evolving peaks, while floods in large catchments can persist for days, with slowly varying inundation from upstream inflow and drainage. Most GNN-based flood models incorporate recurrent units such as GRUs or LSTMs to process time series data (J. Jiang et al.,

2024; Kazadi et al., 2024; Luo et al., 2024; Zhao et al., 2020). While recurrent units are effective at addressing vanishing or exploding gradients, they substantially increase parameter counts and inference time (LeCun et al., 2015). Transformer-based temporal modules have also begun to appear in flood-related GNN models (Chaimook et al., 2025; Li et al., 2025; R. Liu et al., 2026), but their flood-specific adaptation remains comparatively limited. Exploring alternative temporal learning strategies can help balance accuracy with computational efficiency.

One promising direction is convolution-based temporal modeling, first popularized in traffic forecasting by Yu et al. through a simple temporal convolution within STGCN (Yu et al., 2018). This approach proved both accurate and efficient, inspiring extensions that capture multiple time resolutions (Guo et al., 2019) and expand the receptive field via dilated convolutions (Wu et al., 2019). Although some flood GNNs adopt CNN-based temporal learning (Farahmand et al., 2023; Jin et al., 2026; Sui et al., 2026; Sun et al., 2022), these implementations have not been tailored to the specific temporal patterns of flood propagation. For example, convolution can be further adapted to the targeted flood prediction task. Short temporal kernels are well suited to the rapid onset of flash floods, whereas dilated convolutions can extend the receptive field to capture the multi-day evolution of prolonged inundation.

Transformation techniques such as temporal-aware normalization are another underexplored technique in the context of flood modeling. In weather prediction, Keisler et al. normalize changes in physical variables to unit variance within fixed time windows (e.g., three hours) before computing the loss (Keisler, 2022). This improves both training stability and interpretability by making errors directly comparable across spatial and temporal contexts. Applying a similar normalization in flood modeling could enforce temporal consistency in predictions, helping models maintain physically plausible dynamics across different locations and flood stages.

### 5.8. Handling sparse data

Effective handling of data sparsity demands advanced imputation approaches especially for complex spatiotemporal data. Flood modeling studies rely on synthetically generated datasets from numerical models to avoid the issue of missing data. The issue is far more prevalent in real-world sensor data, which has thus far been only insufficiently addressed through simple linear interpolation methods (Farahmand et al., 2023; Zhao et al., 2020). In several disciplines, self-supervised learning has proven effective for completing missing data within graph structures.

Multiple physics-based autoencoder models have been developed to estimate node states in WDS which adhere to natural physical laws (Ashraf et al., 2022; Hajgató et al., 2021; Xing and Sela, 2022). For example, Xing and Sela (2022) train a GNN as an unrolled iterative solver that predicts nodal-head estimates from graphs, masked to expose only a few observed nodes to mimic data sparsity. Rather than supervising the masked nodes against held-out ground truth, the label term is applied only at the observed nodes while the unobserved nodes are constrained through a physics-informed residual, penalizing violations of mass conservation at junctions and head boundary conditions at sources. Once trained, the model can reconstruct the complete field of nodal heads grounded on underlying physical laws, thereby populating the hydraulic state at all unmonitored locations.

This technique has also been explored in other fields. R-GCN, a GNN-based autoencoder, was created to complete missing information in a knowledge graph through application-specific regularization techniques (Schlichtkrull et al., 2017). In the context of flood modeling, self-supervised GNN approaches remain largely unexplored for enhancing real-world data sources. Adopting such methods could significantly reduce the heavy reliance on synthetic data generated by numerical models.

## 6. Conclusions

GNNs offer a compelling framework for flood modeling, aligning naturally with the input mesh of numerical solvers while combining computational efficiency with the capacity to embed physical priors. By representing unstructured domains directly as graphs, they avoid the data transformations that constrain grid-based DL. Yet the field remains early-stage, and the reviewed literature shows that this promise is still largely unrealized.

This survey reviews 49 works on GNN-based flood modeling and identifies salient research gaps. First, physical fidelity remains partial as existing models enforce conservation and momentum constraints only loosely and omit energy-loss terms standard in hydrodynamic solvers. Model scale also lags considerably behind operational requirements, as flood GNNs typically operate on graphs several orders of magnitude smaller than realistic domains. Further evaluations are necessary through standardized benchmarks that assess cross-catchment generalization and sparse real-world data. Lastly, XAI techniques and uncertainty quantification are seldom applied, leaving model behavior insufficiently understood.

Many of these challenges have already been addressed in related domains where GNNs are more mature. Physics-informed methods from CFD and WDS offer a route to enforcing local and momentum constraints, while sampling and distributed training address the problem of scale. Transfer learning supports generalization across settings, self-supervised reconstruction mitigates the effects of sparse data, and heterogeneous designs enable richer feature representations. What flood modeling lacks, therefore, is not theoretical feasibility but the systematic adaptation of these techniques, supported by shared benchmarks that make progress measurable.

This review aims to make those gaps legible and point toward the cross-disciplinary work that can close them. Addressing these gaps would enable GNNs to mature from early prototypes into robust, scalable, and physically consistent flood models that ultimately contribute to safer and more resilient communities.

## CRedit authorship contribution statement

**Carlo Malapad Acosta:** Writing – review & editing, Writing – original draft, Visualization, Conceptualization. **Herath Mudiyansele Viraj Vidura Herath:** Writing – review & editing, Writing – original draft, Visualization, Supervision, Conceptualization. **Abhishek Saha:** Writing – review & editing, Writing – original draft, Conceptualization. **Sanka Rasnayaka:** Writing – review & editing, Visualization, Supervision, Conceptualization. **Lucy Marshall:** Writing – review & editing, Supervision, Conceptualization.

## Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used Google Gemini and ChatGPT in order to improve readability and grammatical structure. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

This work was supported by The University of Sydney – National University of Singapore Ignition Grants 2025 Round.

## Data availability

No data was used for the research described in the article.

## References

- Acosta, C.M., Herath, H.M.V.V., Lim, J.Y., Saha, A., Rasnayaka, S., Marshall, L., 2026. DUALFloodGNN: Physics-informed graph neural network for operational flood modeling. *arXiv:2512.23964* URL: <https://arxiv.org/abs/2512.23964>.
- Agonafir, C., Zheng, T., 2026. Structured exploration of machine learning model complexity for spatio-temporal forecasting of urban flooding. *Urban Water J.* 23 (3), 380–407. <http://dx.doi.org/10.1080/1573062X.2025.2589078>, <https://doi.org/10.1080/1573062X.2025.2589078>.
- Ahmadlou, M., Al-Fugara, A., Al-Shabeeb, A.R., Arora, A., Al-Adamat, R., Pham, Q.B., Al-Ansari, N., Linh, N.T.T., Sajedi, H., 2021. Flood susceptibility mapping and assessment using a novel deep learning model combining multilayer perceptron and autoencoder neural networks. *J. Flood Risk Manag.* 14 (1), e12683. <http://dx.doi.org/10.1111/jfr3.12683>, URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/jfr3.12683>.
- Akkala, A., Boubrahimi, S.F., Hamdi, S.M., Hosseinzadeh, P., Nassar, A., 2025a. Improved streamflow forecasting through SWE-augmented spatio-temporal graph neural networks. *Hydrology* 12 (10), <http://dx.doi.org/10.3390/hydrology12100268>, URL: <https://www.mdpi.com/2306-5338/12/10/268>.
- Akkala, A., Boubrahimi, S.F., Hamdi, S.M., Hosseinzadeh, P., Nassar, A., 2025b. Spatio-temporal graph neural networks for streamflow prediction in the upper colorado basin. *Hydrology* 12 (3), <http://dx.doi.org/10.3390/hydrology12030060>, URL: <https://www.mdpi.com/2306-5338/12/3/60>.
- Amini, J., 2010. A method for generating floodplain maps using IKONOS images and DEMs. *Int. J. Remote Sens.* 31 (9), 2441–2456. <http://dx.doi.org/10.1080/01431160902929230>, <https://doi.org/10.1080/01431160902929230>.
- Ashraf, I., Hermes, L., Artelt, A., Hammer, B., 2022. Spatial graph convolution neural networks for water distribution systems. *arXiv:2211.09587* URL: <https://arxiv.org/abs/2211.09587>.
- Ashraf, I., Strother, J., Hermes, L., Hammer, B., 2024. Physics-informed graph neural networks for water distribution systems. *Proc. the AAAI Conf. Artif. Intell.* 38 (20), 21905–21913. <http://dx.doi.org/10.1609/aaai.v38i20.30192>, URL: <https://ojs.aaai.org/index.php/AAAI/article/view/30192>.
- Barwey, S., Pal, P., Patel, S., Balin, R., Lusch, B., Vishwanath, V., Maulik, R., Balakrishnan, R., 2025. Mesh-based super-resolution of fluid flows with multiscale graph neural networks. *Comput. Methods Appl. Mech. Engrg.* 439, 118072. <http://dx.doi.org/10.1016/j.cma.2025.118072>.
- Barwey, S., Shankar, V., Viswanathan, V., Maulik, R., 2023. Multiscale graph neural network autoencoders for interpretable scientific machine learning. *J. Comput. Phys.* 495, 112537. <http://dx.doi.org/10.1016/j.jcp.2023.112537>, URL: <https://www.sciencedirect.com/science/article/pii/S0021999123006320>.
- Battaglia, P.W., Hamrick, J.B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinin, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., Gulcehre, C., Song, F., Ballard, A., Gilmer, J., Dahl, G., Vaswani, A., Allen, K., Nash, C., Langston, V., Dyer, C., Heess, N., Wierstra, D., Kohli, P., Botvinick, M., Vinyals, O., Li, Y., Pascanu, R., 2018. Relational inductive biases, deep learning, and graph networks. *arXiv:1806.01261* URL: <https://arxiv.org/abs/1806.01261>.
- Belbute-Peres, F.d., Economon, T.D., Kolter, J.Z., 2020. Combining differentiable PDE solvers and graph neural networks for fluid flow prediction. *arXiv:2007.04439* URL: <https://arxiv.org/abs/2007.04439>.
- Bentivoglio, R., Isufi, E., Jonkman, S.N., Taormina, R., 2022. Deep learning methods for flood mapping: a review of existing applications and future research directions. *Hydrol. Earth Syst. Sci.* 26 (16), 4345–4378. <http://dx.doi.org/10.5194/hess-26-4345-2022>, URL: <https://hess.copernicus.org/articles/26/4345/2022/>.
- Bentivoglio, R., Isufi, E., Jonkman, S.N., Taormina, R., 2023. Rapid spatio-temporal flood modelling via hydraulics-based graph neural networks. *Hydrol. Earth Syst. Sci.* 27 (23), 4227–4246. <http://dx.doi.org/10.5194/hess-27-4227-2023>, URL: <https://hess.copernicus.org/articles/27/4227/2023/>.
- Bentivoglio, R., Isufi, E., Jonkman, S.N., Taormina, R., 2024. Multi-scale hydraulic graph neural networks for flood modelling. *EGU sphere* 2024, 1–28. <http://dx.doi.org/10.5194/egusphere-2024-2621>, URL: <https://egusphere.copernicus.org/preprints/2024/egusphere-2024-2621/>.
- Bonnet, F., Mazari, A.J., Cinnella, P., Gallinari, P., 2023. Airfrans: High fidelity computational fluid dynamics dataset for approximating Reynolds-averaged Navier-Stokes solutions. *arXiv:2212.07564* URL: <https://arxiv.org/abs/2212.07564>.
- Brunner, G.W., 1997. HEC-ras river analysis system. *Hydraulic reference manual. Version 1.0.*
- Buterez, D., Janet, J.P., Kiddle, S.J., Oglic, D., Lió, P., 2024. Transfer learning with graph neural networks for improved molecular property prediction in the multi-fidelity setting. *Nat. Commun.* 15 (1), 1517. <http://dx.doi.org/10.1038/s41467-024-45566-8>, URL: <https://doi.org/10.1038/s41467-024-45566-8>.
- Chaimook, P., Khamsemanan, N., Nattee, C., Sharp, A., 2025. Spatiotemporal flood hazard classification in bangkok using graph convolutional network and temporal fusion transformer. *IEEE Access* <http://dx.doi.org/10.1109/ACCESS.2025.3597328>.

- Charania, K.M., Patel, J.N., 2026. Advancing hydrological forecasting in semi-arid river basins through spatiotemporal deep learning frameworks. *Eng. Res. Express* 8 (2), 025107. <http://dx.doi.org/10.1088/2631-8695/ae3273>.
- Chen, J., Hachem, E., Viquerat, J., 2021. Graph neural networks for laminar flow prediction around random two-dimensional shapes. *Phys. Fluids* 33 (12), 123607. <http://dx.doi.org/10.1063/5.0064108>, [https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0064108/15868523/123607\\_1\\_online.pdf](https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0064108/15868523/123607_1_online.pdf).
- Chen, W., Hong, H., Li, S., Shahabi, H., Wang, Y., Wang, X., Ahmad, B.B., 2019. Flood susceptibility modelling using novel hybrid approach of reduced-error pruning trees with bagging and random subspace ensembles. *J. Hydrol.* 575, 864–873. <http://dx.doi.org/10.1016/j.jhydrol.2019.05.089>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169419305347>.
- Chen, W., Li, Y., Xue, W., Shahabi, H., Li, S., Hong, H., Wang, X., Bian, H., Zhang, S., Pradhan, B., Ahmad, B.B., 2020. Modeling flood susceptibility using data-driven approaches of naïve Bayes tree, alternating decision tree, and random forest methods. *Sci. Total Environ.* 701, 134979. <http://dx.doi.org/10.1016/j.scitotenv.2019.134979>, URL: <https://www.sciencedirect.com/science/article/pii/S004896971934971X>.
- Chiang, W.-L., Liu, X., Si, S., Li, Y., Bengio, S., Hsieh, C.-J., 2019. Cluster-GCN: An efficient algorithm for training deep and large graph convolutional networks. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. In: KDD 2019, ACM, pp. 257–266. <http://dx.doi.org/10.1145/3292500.3330925>.
- Christensen, B.A., Locher, F.A., Swamee, P.K., 2000. Limitations and proper use of the hazen-williams equation. *J. Hydraul. Eng.* 126 (2), 167–170. [http://dx.doi.org/10.1061/\(ASCE\)0733-9429\(2000\)126:2\(167\)](http://dx.doi.org/10.1061/(ASCE)0733-9429(2000)126:2(167)), URL: <https://ascelibrary.org/doi/abs/10.1061/%28ASCE%290733-9429%282000%29126%3A2%28167%29>.
- Costabile, P., Costanzo, C., Macchione, F., 2017. Performances and limitations of the diffusive approximation of the 2-d shallow water equations for flood simulation in urban and rural areas. *Appl. Numer. Math.* 116, 141–156. <http://dx.doi.org/10.1016/j.apnum.2016.07.003>, URL: <https://www.sciencedirect.com/science/article/pii/S0168927416301106> New Trends in Numerical Analysis: Theory, Methods, Algorithms and Applications (NETNA 2015).
- Deltares, 2022. Delft3D-FM user manual. [https://content.oss.deltares.nl/delft3d/manuals/D-Flow\\_FM\\_User\\_Manual.pdf](https://content.oss.deltares.nl/delft3d/manuals/D-Flow_FM_User_Manual.pdf) (last access: 18 February 2023).
- Deng, L., Zhang, X., Slater, L.J., Liu, H., Tao, S., 2024. Integrating euclidean and non-euclidean spatial information for deep learning-based spatiotemporal hydrological simulation. *J. Hydrol.* 638, 131438. <http://dx.doi.org/10.1016/j.jhydrol.2024.131438>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169424008333>.
- Devia, G.K., Ganasri, B., Dwarakish, G., 2015. A review on hydrological models. *Aquat. Procedia* 4, 1001–1007. <http://dx.doi.org/10.1016/j.aqpro.2015.02.126>, URL: <https://www.sciencedirect.com/science/article/pii/S2214241X15001273> INTERNATIONAL CONFERENCE ON WATER RESOURCES, COASTAL AND OCEAN ENGINEERING (ICWRCOE'15).
- Devkota, R., Bhattarai, U., Devkota, L., Maraseni, T.N., 2020. Assessing the past and adapting to future floods: a hydro-social analysis. *Clim. Change* 163 (2), 1065–1082. <http://dx.doi.org/10.1007/s10584-020-02909-w>.
- Donaldson, R., Dyer, R.M., Kraus, M.J., 1975. An objective evaluator of techniques for predicting severe weather events. In: *Preprints, Ninth Conf. on Severe Local Storms, Norman, OK, Amer. Meteor. Soc. Vol. 321326*.
- Dong, S., Yu, T., Farahmand, H., Mostafavi, A., 2020. A hybrid deep learning model for predictive flood warning and situation awareness using channel network sensors data. *arXiv:2006.09201* URL: <https://arxiv.org/abs/2006.09201>.
- Donnelly, J., Daneshkhan, A., Abolfathi, S., 2024. Physics-informed neural networks as surrogate models of hydrodynamic simulators. *Sci. Total Environ.* 912, 168814. <http://dx.doi.org/10.1016/j.scitotenv.2023.168814>, URL: <https://www.sciencedirect.com/science/article/pii/S0048969723074430>.
- El Moumen, R., Hssaisoune, M., Ait-Brahim, Y., Sbahi, S., Garnes, E.P., Lahssaine, I.A., Bouchaou, L., 2026. Riverine flood mapping methods and criteria: A meta-analysis review and synthesized guidelines. *Nat. Hazards Res.* <http://dx.doi.org/10.1016/j.nhres.2026.03.004>, URL: <https://www.sciencedirect.com/science/article/pii/S266659212600017X>.
- Elinas, P., Bonilla, E.V., Tiao, L., 2020. Variational inference for graph convolutional networks in the absence of graph data and adversarial settings. *arXiv:1906.01852* URL: <https://arxiv.org/abs/1906.01852>.
- Fang, X., Li, J., Wang, M., Chen, A., Shao, S., Liu, Q., 2025. A universal urban flood risk model based on particle-swarm-optimization-enhanced spiking graph convolutional networks. *Sustainability* 17 (22), <http://dx.doi.org/10.3390/su17229973>, URL: <https://www.mdpi.com/2071-1050/17/22/9973>.
- Fang, Z., Wang, Y., Peng, L., Hong, H., 2021. Predicting flood susceptibility using LSTM neural networks. *J. Hydrol.* 594, 125734. <http://dx.doi.org/10.1016/j.jhydrol.2020.125734>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169420311951>.
- Farahmand, H., Xu, Y., Mostafavi, A., 2023. A spatial-temporal graph deep learning model for urban flood nowcasting leveraging heterogeneous community features. *Sci. Rep.* 13 (1), 6768. <http://dx.doi.org/10.1038/s41598-023-32548-x>, URL: <https://doi.org/10.1038/s41598-023-32548-x>.
- Fey, M., Lenssen, J.E., 2019. Fast graph representation learning with PyTorch Geometric. In: *ICLR Workshop on Representation Learning on Graphs and Manifolds*.
- Fraehr, N., Wang, Q.J., Wu, W., Nathan, R., 2023. Development of a fast and accurate hybrid model for floodplain inundation simulations. *Water Resour. Res.* 59 (6), e2022WR033836. <http://dx.doi.org/10.1029/2022WR033836>, URL: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022WR033836> e2022WR033836 e2022WR033836.
- Fraehr, N., Wang, Q.J., Wu, W., Nathan, R., 2024. Assessment of surrogate models for flood inundation: The physics-guided LSG model vs. state-of-the-art machine learning models. *Water Res.* 252, 121202. <http://dx.doi.org/10.1016/j.watres.2024.121202>, URL: <https://www.sciencedirect.com/science/article/pii/S0043135424001027>.
- Gao, H., Zahr, M.J., Wang, J.-X., 2022. Physics-informed graph neural Galerkin networks: A unified framework for solving PDE-governed forward and inverse problems. *Comput. Methods Appl. Mech. Engrg.* 390, 114502. <http://dx.doi.org/10.1016/j.cma.2021.114502>, URL: <https://www.sciencedirect.com/science/article/pii/S0045782521007076>.
- Garzón, A., Kapelan, Z., Langeveld, J., Taormina, R., 2024a. Accelerating urban drainage simulations: A data-efficient GNN metamodel for SWMM flowrates. *Eng. Proc.* 69 (1), <http://dx.doi.org/10.3390/engproc2024069137>, URL: <https://www.mdpi.com/2673-4591/69/1/137>.
- Garzón, A., Kapelan, Z., Langeveld, J., Taormina, R., 2024b. Transferable and data efficient metamodeling of storm water system nodal depths using auto-regressive graph neural networks. *Water Res.* 266, 122396. <http://dx.doi.org/10.1016/j.watres.2024.122396>, URL: <https://www.sciencedirect.com/science/article/pii/S0043135424012958>.
- Garzón, A., Kapelan, Z., Langeveld, J., Taormina, R., 2026. Evaluation of graph neural networks for urban drainage metamodeling: Key components and transferability analysis. *Water Res.* 290, 125079. <http://dx.doi.org/10.1016/j.watres.2025.125079>, URL: <https://www.sciencedirect.com/science/article/pii/S0043135425019827>.
- Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E., 2017. Neural message passing for quantum chemistry. *arXiv:1704.01212* URL: <https://arxiv.org/abs/1704.01212>.
- Guo, W.-D., Chen, W.-B., Chang, C.-H., 2025. A spatiotemporal watershed-scale machine-learning model for hourly and daily flood-water level prediction: the case of the tidal beigang river, Taiwan. *Nat. Hazards* 121 (8), 9563–9611. <http://dx.doi.org/10.1007/s11069-025-07187-2>.
- Guo, Z., Leita, J.P., Simoes, N.E., Moosavi, V., 2020. Data-driven flood emulation: Speeding up urban flood predictions by deep convolutional neural networks. *arXiv:2004.08340* URL: <https://arxiv.org/abs/2004.08340>.
- Guo, S., Lin, Y., Feng, N., Song, C., Wan, H., 2019. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In: *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*. In: AAAI'19/IAAI'19/EAAI'19, AAAI Press, <http://dx.doi.org/10.1609/aaai.v33i01.3301922>.
- Ha, C.-Y., Kim, B.-J., Lee, J.-N., Kim, B.-H., 2023. Parameter optimization of coupled 1D–2D hydrodynamic model for urban flood inundation. *Water* 15 (16), 2946. <http://dx.doi.org/10.3390/w15162946>.
- Hajgató, G., Gyires-Tóth, B., Paál, G., 2021. Reconstructing nodal pressures in water distribution systems with graph neural networks. *arXiv:2104.13619* URL: <https://arxiv.org/abs/2104.13619>.
- Hamilton, W.L., Ying, R., Leskovec, J., 2018. Inductive representation learning on large graphs. *arXiv:1706.02216* URL: <https://arxiv.org/abs/1706.02216>.
- He, L., Nan, J., Ye, X., Chen, L., Ji, S., Chen, Z., Xiao, Q., 2025. A graph neural network using physical attributes to improve the system-wide nodal water-level prediction in sparsely monitored urban drainage systems. *J. Hydrol.* 663, 134306. <http://dx.doi.org/10.1016/j.jhydrol.2025.134306>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169425016464>.
- He, X., Wang, Y., Li, J., 2022. Flow completion network: Inferring the fluid dynamics from incomplete flow information using graph neural networks. *Phys. Fluids* 34 (8), <http://dx.doi.org/10.1063/5.0097688>.
- Herath, H.M.V.V., Marshall, L., Saha, A., Rasnayaka, S., Senewiratne, S., 2025. Subgrid informed neural networks for high-resolution flood mapping. *J. Hydrol.* 660, <http://dx.doi.org/10.1016/j.jhydrol.2025.133329>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169425006675>.
- Hong, H., Tsangaratos, P., Ilia, I., Liu, J., Zhu, A.-X., Chen, W., 2018. Application of fuzzy weight of evidence and data mining techniques in construction of flood susceptibility map of poyang county, China. *Sci. Total Environ.* 625, 575–588. <http://dx.doi.org/10.1016/j.scitotenv.2017.12.256>, URL: <https://www.sciencedirect.com/science/article/pii/S0048969717336902>.
- Horie, M., Mitsune, N., 2023. Physics-embedded neural networks: Graph neural PDE solvers with mixed boundary conditions. *arXiv:2205.11912* URL: <https://arxiv.org/abs/2205.11912>.
- Hosseini, H., 2021. A deep learning model for predicting river flood depth and extent. *Environ. Model. Softw.* 145, 105186. <http://dx.doi.org/10.1016/j.envsoft.2021.105186>, URL: <https://www.sciencedirect.com/science/article/pii/S1364815221002280>.

- Huang, R., Long, Y., Zia, T., 2025. A physical-enhanced spatio-temporal graph convolutional network for river flow prediction. *Appl. Sci.* 15 (16), <https://doi.org/10.3390/app15169054>, URL: <https://www.mdpi.com/2076-3417/15/16/9054>.
- Jagadeish, H.V., Gehrke, J., Labrinidis, A., Papakonstantinou, Y., Patel, J.M., Ramakrishnan, R., Shahabi, C., 2014. Big data and its technical challenges. *Commun. ACM* 57 (7), 86–94. <https://doi.org/10.1145/2611567>.
- Jiang, J., Chen, C., Zhou, Y., Berretti, S., Liu, L., Pei, Q., Zhou, J., Wan, S., 2024. Heterogeneous dynamic graph convolutional networks for enhanced spatiotemporal flood forecasting by remote sensing. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 17, 3108–3122. <https://doi.org/10.1109/JSTARS.2023.3349162>.
- Jiang, P., Meinert, N., Jordão, H., Weisser, C., Holgate, S., Lavin, A., Lütjens, B., Newman, D., Wainwright, H., Walker, C., Barnard, P., 2021. Digital twin earth – coasts: Developing a fast and physics-informed surrogate model for coastal floods via neural operators. *arXiv:2110.07100* URL: <https://arxiv.org/abs/2110.07100>.
- Jiang, W., Zhang, J., Li, Y., Zhang, D., Hu, G., Gao, H., Duan, Z., 2024. Advancing storm surge forecasting from scarce observation data: A causal-inference based spatio-temporal graph neural network approach. *Coast. Eng.* 190, 104512. <https://doi.org/10.1016/j.coastaleng.2024.104512>, URL: <https://www.sciencedirect.com/science/article/pii/S0378383924000607>.
- Jiaxuan You, Ying, R., Leskovec, J., 2021. Design space for graph neural networks. *arXiv:2011.08843* URL: <https://arxiv.org/abs/2011.08843>.
- Jin, X., Qin, M., Duan, H., 2026. Spatio-temporal graph convolutional network incorporating knowledge graph embeddings for hydrological time series prediction. *IEEE Access* <https://doi.org/10.1109/ACCESS.2026.3656563>.
- Jonkman, S.N., Kelman, I., 2005. An analysis of the causes and circumstances of flood disaster deaths. *Disasters* 29 (1), 75–97. <https://doi.org/10.1111/j.0361-3666.2005.00275.x>, URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.0361-3666.2005.00275.x>.
- Kang, Y., Kubatko, E.J., 2024. An automatic mesh generator for coupled 1D–2D hydrodynamic models. *Geosci. Model. Dev.* 17 (4), 1603–1625. <https://doi.org/10.5194/gmd-17-1603-2024>, URL: <https://gmd.copernicus.org/articles/17/1603/2024/>.
- Kazadi, A., Doss-Gollin, J., Sebastian, A., Silva, A., 2024. FloodGNN-GRU: a spatio-temporal graph neural network for flood prediction. *Environ. Data Sci.* 3, e21. <https://doi.org/10.1017/eds.2024.19>.
- Keisler, R., 2022. Forecasting global weather with graph neural networks. *arXiv:2202.07575* URL: <https://arxiv.org/abs/2202.07575>.
- Kilinc, H.C., Apak, S., Ozkan, F., Yurtsever, A., 2025. AH-GCAN-Istm: Adaptive hybrid-graph convolution attention network with LSTM and genetic optimization for hydrological time series forecasting. *Water Resour. Manag.* 39 (15), 8293–8312. <https://doi.org/10.1007/s11269-025-04343-y>.
- Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., Li, Q., Shoemaker, B.A., Thiessen, P.A., Yu, B., Zaslavsky, L., Zhang, J., Bolton, E.E., 2024. PubChem 2025 update. *Nucleic Acids Res.* 53 (D1), D1516–D1525. <https://doi.org/10.1093/nar/gkae1059>, <https://academic.oup.com/nar/article-pdf/53/D1/D1516/60743708/gkae1059.pdf>.
- Kim, M., He, Y., Luo, J., 2024. Physics-informed geostatistical approach (PIGA) for real-time fluvial flood modeling. *J. Hydrol.* 642, 131878. <https://doi.org/10.1016/j.jhydrol.2024.131878>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169424012745>.
- Kipf, T.N., Welling, M., 2017. Semi-supervised classification with graph convolutional networks. *arXiv:1609.02907* URL: <https://arxiv.org/abs/1609.02907>.
- Kumar, V., Sharma, K.V., Caloiero, T., Mehta, D.J., Singh, K., 2023. Comprehensive overview of flood modeling approaches: A review of recent advances. *Hydrology* 10 (7), <https://doi.org/10.3390/hydrology10070141>, URL: <https://www.mdpi.com/2306-5338/10/7/141>.
- Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirsberger, P., Fortunato, M., Alet, F., Ravuri, S., Ewalds, T., Eaton-Rosen, Z., Hu, W., Merose, A., Hoyer, S., Holland, G., Vinyals, O., Stott, J., Pritzel, A., Mohamed, S., Battaglia, P., 2023. Learning skillful medium-range global weather forecasting. *Science* 382 (6677), 1416–1421. <https://doi.org/10.1126/science.adi2336>, URL: <https://www.science.org/doi/abs/10.1126/science.adi2336>.
- Landuyt, L., Verhoest, N.E.C., Van Coillie, F.M.B., 2020. Flood mapping in vegetated areas using an unsupervised clustering approach on sentinel-1 and -2 imagery. *Remote Sens.* 12 (21), <https://doi.org/10.3390/rs12213611>, URL: <https://www.mdpi.com/2072-4292/12/21/3611>.
- LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature* 521 (7553), 436–444. <https://doi.org/10.1038/nature14539>.
- Lee, C.-K., Lu, C., Yu, Y., Sun, Q., Hsieh, C.-Y., Zhang, S., Liu, Q., Shi, L., 2021. Transfer learning with graph neural networks for optoelectronic properties of conjugated oligomers. *J. Chem. Phys.* 154 (2), <https://doi.org/10.1063/5.0037863>.
- Li, Z., Farimani, A.B., 2022. Graph neural network-accelerated Lagrangian fluid simulation. *Comput. Graph.*
- Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., Anandkumar, A., 2021. Fourier neural operator for parametric partial differential equations. *arXiv:2010.08895* URL: <https://arxiv.org/abs/2010.08895>.
- Li, M., Shi, X., Lu, Z., Kapelan, Z., 2024. Predicting the urban stormwater drainage system state using the graph-WaveNet. *Sustain. Cities Soc.* 115, 105877. <https://doi.org/10.1016/j.scs.2024.105877>, URL: <https://www.sciencedirect.com/science/article/pii/S2210670724007017>.
- Li, Y., Su, M., Duan, Z., Liu, H., 2024. A new integrated prediction method of river level based on spatiotemporal correlation. *Stoch. Environ. Res. Risk Assess.* 38 (3), 1121–1143. <https://doi.org/10.1007/s00477-023-02617-8>.
- Li, Z., Zhang, W., Liu, Z., Li, S., Hao, J., Loo, C.K., 2025. Water-state-aware spatiotemporal graph transformer model for water-level prediction. *J. Mar. Sci. Eng.* 13 (11), <https://doi.org/10.3390/jmse13112187>, URL: <https://www.mdpi.com/2077-1312/13/11/2187>.
- Li, T., Zou, S., Chang, X., Zhang, L., Deng, X., 2024. Predicting unsteady incompressible fluid dynamics with finite volume informed neural network. *Phys. Fluids* 36 (4), <https://doi.org/10.1063/5.0197425>, URL: <https://doi.org/10.1063/5.0197425>.
- Liang, Z., Fang, X., Liang, Z., Xiong, J., Deng, F., Nyamasviswa, T.E., 2024. Graph spiking neural network for advanced urban flood risk assessment. *IScience* 27 (11), 111037. <https://doi.org/10.1016/j.isci.2024.111037>.
- Lim, J.Y., Herath, H.M.V.V., Rasnayaka, S., Marshall, L., Zou, H., Saha, A., Hossain, J., Tong, J.Y., Lee, I.Z., Tsukada, Y., Gandhi, S., Matsumaro, K., Raj, S., Mazur, A., Khlopontnykh, A., Cheshenko, D., 2026a. UrbanFloodBench: Bridging AI and hydrology through benchmarking of coupled 1D–2D urban flood surrogate models. <https://doi.org/10.2139/ssrn.6900753>, URL: <https://ssrn.com/abstract=6900753> SSRN preprint.
- Lim, J.Y., Herath, H.M.V.V., Rasnayaka, S., Marshall, L., Zou, H., Saha, A., Hossain, J., Tong, J.Y., Lee, I.Z., Tsukada, Y., Gandhi, S., Matsumaro, K., Raj, S., Mazur, A., Khlopontnykh, A., Cheshenko, D., 2026b. UrbanFloodBench: Dataset repository. <https://doi.org/10.25910/96sa-yk38>, Dataset.
- Lin, H., Sun, Y., 2022. Eleggnet: Electrical-model-guided graph neural networks for power distribution system state estimation. In: *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*. pp. 5292–5298. <https://doi.org/10.1109/GLOBECOM48099.2022.10000808>.
- Lino, M., Fotiadis, S., Bharath, A.A., Cantwell, C.D., 2022. Multi-scale rotation-equivariant graph neural networks for unsteady Eulerian fluid dynamics. *Phys. Fluids* 34 (8), 087110. <https://doi.org/10.1063/5.0097679>, [https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0097679/19818588/087110\\_1\\_online.pdf](https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0097679/19818588/087110_1_online.pdf).
- Liu, S., Li, Y., Zhang, Y., Chen, Y., Ma, J., 2026. An ensemble graph and series decomposition model for water level prediction at both real and virtual stations in the yangtze river. *IEEE Trans. Instrum. Meas.* <https://doi.org/10.1109/TIM.2026.3654723>.
- Liu, G., Ouyang, S., Qin, H., Liu, S., Shen, Q., Qu, Y., Zheng, Z., Sun, H., Zhou, J., 2023. Assessing spatial connectivity effects on daily streamflow forecasting using Bayesian-based graph neural network. *Sci. Total Environ.* 855, 158968. <https://doi.org/10.1016/j.scitotenv.2022.158968>, URL: <https://www.sciencedirect.com/science/article/pii/S0048969722060673>.
- Liu, R., Wang, L., Zeng, L., Wang, D., Li, X., 2026. A hybrid AE-SDGC-autoformer model for short-term runoff forecasting and sustainable water resource management. *Sustainability* 18 (4), <https://doi.org/10.3390/su18042096>, URL: <https://www.mdpi.com/2071-1050/18/4/2096>.
- Löwe, R., Böhm, J., Jensen, D.G., Leandro, J., Rasmussen, S.H., 2021. U-FLOOD – topographic deep learning for predicting urban pluvial flood water depth. *J. Hydrol.* 603, 126898. <https://doi.org/10.1016/j.jhydrol.2021.126898>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169421009483>.
- Luo, P., Luo, M., Li, F., Qi, X., Huo, A., Wang, Z., He, B., Takara, K., Nover, D., Wang, Y., 2022. Urban flood numerical simulation: Research, methods and future perspectives. *Environ. Model. Softw.* 156, 105478. <https://doi.org/10.1016/j.envsoft.2022.105478>, URL: <https://www.sciencedirect.com/science/article/pii/S1364815222001827>.
- Luo, Y., Zhou, Y., Chen, H., Xiong, L., Guo, S., Chang, F.-J., 2024. Exploring a spatiotemporal hetero graph-based long short-term memory model for multi-step-ahead flood forecasting. *J. Hydrol.* 633, 130937. <https://doi.org/10.1016/j.jhydrol.2024.130937>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169424003317>.
- Mallick, T., Balaprakash, P., Rask, E., Macfarlane, J., 2020. Graph-partitioning-based diffusion convolutional recurrent neural network for large-scale traffic forecasting. *Transp. Res. Rec.: J. Transp. Res. Board* 2674 (9), 473–488. <https://doi.org/10.1177/0361198120930010>, URL: <https://doi.org/10.1177/0361198120930010>.
- Ming, X., Liang, Q., Xia, X., Li, D., Fowler, H.J., 2020. Real-time flood forecasting based on a high-performance 2-d hydrodynamic model and numerical weather predictions. *Water Resour. Res.* 56 (7), e2019WR025583. <https://doi.org/10.1029/2019WR025583>, URL: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2019WR025583> e2019WR025583 2019WR025583.
- Morales-Hernández, M., García-Navarro, P., Burgette, J., Brufau, P., 2013. A conservative strategy to couple 1D and 2D models for shallow water flow simulation. *Comput. Fluids* 81, 26–44. <https://doi.org/10.1016/j.compfluid.2013.04.001>, ISBN: 9781138026742.
- Mosavi, A., Ozturk, P., Chau, K.-w., 2018. Flood prediction using machine learning models: Literature review. *Water* 10 (11), <https://doi.org/10.3390/w10111536>, URL: <https://www.mdpi.com/2073-4441/10/11/1536>.
- Nash, J.E., Sutcliffe, J.V., 1970. River flow forecasting through conceptual models part I: A discussion of principles. *J. Hydrol.* 10 (3), 282–290. [https://doi.org/10.1016/0022-1694\(70\)90255-6](https://doi.org/10.1016/0022-1694(70)90255-6).
- Nauck, C., Lindner, M., Schürholt, K., Hellmann, F., 2023. Toward dynamic stability assessment of power grid topologies using graph neural networks. *Chaos: An Interdiscip. J. Nonlinear Sci.* 33 (10), <https://doi.org/10.1063/5.0160915>.

- Nauck, C., Lindner, M., Schürholt, K., Zhang, H., Schultz, P., Kurths, J., Isenhardt, I., Hellmann, F., 2022. Predicting basin stability of power grids using graph neural networks. *New J. Phys.* 24 (4), 043041. <http://dx.doi.org/10.1088/1367-2630/ac54c9>, URL: <http://dx.doi.org/10.1088/1367-2630/ac54c9>.
- Neo, S.H., Seneviratne, S., Herath, H.M.V.V., Saha, A., Rasnayaka, S., Marshall, L.A., 2026. Flood-LDM: Generalizable latent diffusion models for rapid and accurate zero-shot high-resolution flood mapping. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. WACV*, pp. 8063–8072.
- Nguyen, D.H., Vo, N.D., Doan, T.N.C., Truong, T.T.U., Vu, H.C., 2026. Hybrid GNN and temporal encoder models for flood forecasting in the vu gia-thu bon river basin. *River* 5 (1), 3–21. <http://dx.doi.org/10.1002/rvr2.70046>, URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/rvr2.70046>.
- Noh, S.J., Lee, J.-H., Lee, S., Kawaike, K., Seo, D.-J., 2018. Hyper-resolution 1D-2D urban flood modelling using LIDAR data and hybrid parallelization. *Environ. Model. Softw.* 103, 131–145. <http://dx.doi.org/10.1016/j.envsoft.2018.02.008>, URL: <http://linkinghub.elsevier.com/retrieve/pii/S1364815217308198> Publisher: Elsevier Ltd.
- Pan, X., Hou, J., Li, D., Wang, Y., Li, X., Sun, J., Fan, C., Yang, Y., 2026. A study on rapid dynamic flood forecasting in small watersheds using a GNN-transformer approach integrated with spatial physical information. *Water Resour. Manag.* 40 (4), 173. <http://dx.doi.org/10.1007/s11269-025-04360-x>.
- Panahi, M., Dodangeh, E., Rezaie, F., Khosravi, K., Van Le, H., Lee, M.-J., Lee, S., Thai Pham, B., 2021. Flood spatial prediction modeling using a hybrid of meta-optimization and support vector regression modeling. *CATENA* 199, 105114. <http://dx.doi.org/10.1016/j.catena.2020.105114>, URL: <https://www.sciencedirect.com/science/article/pii/S0341816220306640>.
- Peng, J.-Z., Wang, Y.-Z., Chen, S., Chen, Z.-H., Wu, W.-T., Aubry, N., 2022. Grid adaptive reduced-order model of fluid flow based on graph convolutional neural network. *Phys. Fluids* 34 (8), 087121. <http://dx.doi.org/10.1063/5.0100236>, [https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0100236/16589586/087121\\_1\\_online.pdf](https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/5.0100236/16589586/087121_1_online.pdf).
- Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., Battaglia, P.W., 2021. Learning mesh-based simulation with graph networks. *arXiv:2010.03409* URL: <https://arxiv.org/abs/2010.03409>.
- Qian, K., Mohamed, A., Claudel, C., 2019. Physics informed data driven model for flood prediction: Application of deep learning in prediction of urban flood development. *arXiv:1908.10312* URL: <https://arxiv.org/abs/1908.10312>.
- Rahimzad, M., Moghaddam Nia, A., Zolfonoon, H., et al., 2021. Performance comparison of an LSTM-based deep learning model versus conventional machine learning algorithms for streamflow forecasting. *Water Resour. Manag.* 35 (12), 4167–4187. <http://dx.doi.org/10.1007/s11269-021-02937-w>.
- Raissi, M., Perdikaris, P., Karniadakis, G., 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* 378, 686–707. <http://dx.doi.org/10.1016/j.jcp.2018.10.045>, URL: <https://www.sciencedirect.com/science/article/pii/S0021999118307125>.
- Refice, A., Zingaro, M., D'Addabbo, A., Chini, M., 2020. Integrating C- and L-band SAR imagery for detailed flood monitoring of Remote Vegetated Areas. *Water* 12 (10), <http://dx.doi.org/10.3390/w12102745>, URL: <https://www.mdpi.com/2073-4441/12/10/2745>.
- Rentschler, J., Salhab, M., Jafino, B.A., 2022. Flood exposure and poverty in 188 countries. *Nat. Commun.* 13 (1), 3527. <http://dx.doi.org/10.1038/s41467-022-30727-4>.
- Rezaeianzadeh, M., 2013. Assessment of a conceptual hydrological model and artificial neural networks for daily outflows forecasting. *Int. J. Environ. Sci. Technol.* (Tehran) 10 (6), 1181–1192. <http://dx.doi.org/10.1007/s13762-013-0209-0>.
- Rezaeianzadeh, M., Tabari, H., Arabi Yazdi, A., Isik, S., Kalin, L., 2014. Flood flow forecasting using ANN, ANFIS and regression models. *Neural Comput. Appl.* 25 (1), 25–37. <http://dx.doi.org/10.1007/s00521-013-1443-6>.
- Rocha, P.A.C., Santos, V.O., Scott, J., Thé, J.V.G., Gharabaghi, B., 2025. Application of graph neural networks to forecast urban flood events: the case study of the 2013 flood of the bow river, calgary, Canada. *Int. J. River Basin Manag.* 23 (3), 469–486. <http://dx.doi.org/10.1080/15715124.2024.2329243>, <https://doi.org/10.1080/15715124.2024.2329243>, [arXiv:https://doi.org/10.1080/15715124.2024.2329243](https://arxiv.org/abs/https://doi.org/10.1080/15715124.2024.2329243).
- Roland, M., Stuckey, M., 2008. Regression Equations for Estimating Flood Flows at Selected Recurrence Intervals for Ungaged Streams in Pennsylvania. *Scientific Investigations Report 2008–5102*, U.S. Geological Survey, p. 57, URL: <https://pubs.usgs.gov/sir/2008/5102/>.
- Roudbari, N.S., Puneekar, S.R., Patterson, Z., Eicker, U., Poullis, C., 2024. From data to action in flood forecasting leveraging graph neural networks and digital twin visualization. *Sci. Rep.* 14 (1), 18571. <http://dx.doi.org/10.1038/s41598-024-68857-y>.
- Saha, A., Stelling, G., Vuik, C., 2025. Modified horton's infiltration model integrated with a sub-grid shallow water equation solver for high-resolution flooding simulation. *J. Hydrol.* 660, 133326. <http://dx.doi.org/10.1016/j.jhydrol.2025.133326>.
- Sanchez-Gonzalez, A., Godwin, J., Pfaff, T., Ying, R., Leskovec, J., Battaglia, P.W., 2020. Learning to simulate complex physics with graph networks. *arXiv:2002.09405* URL: <https://arxiv.org/abs/2002.09405>.
- Sanders, B.F., Schubert, J.E., Martin, E.-M.H., Wang, S., Sukop, M.C., Mach, K.J., 2025. A fast flood inundation model with groundwater interactions and hydraulic structures. *Adv. Water Resour.* 105057. <http://dx.doi.org/10.1016/j.advwatres.2025.105057>.
- Sañudo, E., García-Feal, O., Hagen, L., Cea, L., Puertas, J., Montalvo, C., Alvarado-Vicencio, R., Hofmann, J., 2025. IberSWMM+: A high-performance computing solver for 2D-1D pluvial flood modelling in urban environments. *J. Hydrol.* 651, 132603. <http://dx.doi.org/10.1016/j.jhydrol.2024.132603>, URL: <https://linkinghub.elsevier.com/retrieve/pii/S0022169424019991>.
- Schlichtkrull, M., Kipf, T.N., Bloem, P., van den Berg, R., Titov, I., Welling, M., 2017. Modeling relational data with graph convolutional networks. *arXiv:1703.06103* URL: <https://arxiv.org/abs/1703.06103>.
- Seo, Y., Defferrard, M., Vandergheynst, P., Bresson, X., 2016. Structured sequence modeling with graph convolutional recurrent networks. *arXiv:1612.07659* URL: <https://arxiv.org/abs/1612.07659>.
- Shao, Y., Li, H., Gu, X., Yin, H., Li, Y., Miao, X., Zhang, W., Cui, B., Chen, L., 2023. Distributed graph neural network training: A survey. *arXiv:2211.00216* URL: <https://arxiv.org/abs/2211.00216>.
- Shao, X., Zhang, X., Gao, Y., Li, B., 2025. A method for fast prediction of flood evolution process based on graph neural network. *Comput. Graph.*
- Sharma, P., Chung, W.T., Akoush, B., Ihme, M., 2023. A review of physics-informed machine learning in fluid mechanics. *Energies* 16 (5), <http://dx.doi.org/10.3390/en16052343>, URL: <https://www.mdpi.com/1996-1073/16/5/2343>.
- Shaw, J., Kesserwani, G., Neal, J., Bates, P., Sharifin, M.K., 2021. LISFLOOD-FP 8.0: the new discontinuous Galerkin shallow-water solver for multi-core CPUs and GPUs. *Geosci. Model. Dev.* 14 (6), 3577–3602. <http://dx.doi.org/10.5194/gmd-14-3577-2021>, URL: <https://gmd.copernicus.org/articles/14/3577/2021/>.
- Simon, C.A., Langendoen, E.J., Abad, J.D., Mendoza, A., 2019. On the governing equations for horizontal and vertical coupling of one- and two-dimensional open channel flow models. *J. Hydraul. Res.* <http://dx.doi.org/10.1080/00221686.2019.1671507>, Publisher: Taylor and Francis Ltd.
- Stadler, M., Charpentier, B., Geisler, S., Zügner, D., Günnemann, S., 2021. Graph posterior network: Bayesian predictive uncertainty for node classification. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (Eds.), *Advances in Neural Information Processing Systems*. 34, Curran Associates, Inc., pp. 18033–18048, URL: [https://proceedings.neurips.cc/paper\\_files/paper/2021/file/95b431e51fc53692913da5263c214162-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2021/file/95b431e51fc53692913da5263c214162-Paper.pdf).
- Stelling, G.S., 2012. Quadtree flood simulations with sub-grid digital elevation models. *Proc. Inst. Civ. Eng. - Water Manag.* 165 (10), 567–580. <http://dx.doi.org/10.1680/wama.12.00018>.
- Sui, T., Wang, C., Zhuang, Y., Yang, S., Fang, J., Huang, Q., 2026. Hydrotgnn: Filling missing streamflow data with graph neural network. *Environ. Model. Softw.* 199, 106937. <http://dx.doi.org/10.1016/j.envsoft.2026.106937>, URL: <https://www.sciencedirect.com/science/article/pii/S1364815226000848>.
- Sun, A.Y., Jiang, P., Mudunuru, M.K., Chen, X., 2021. Explore spatio-temporal learning of large sample hydrology using graph neural networks. *Water Resour. Res.* 57 (12), e2021WR030394. <http://dx.doi.org/10.1029/2021WR030394>, [arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2021WR030394](https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2021WR030394), URL: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2021WR030394> e2021WR030394 2021WR030394.
- Sun, A.Y., Jiang, P., Yang, Z.-L., Xie, Y., Chen, X., 2022. A graph neural network (GNN) approach to basin-scale river network learning: the role of physics-based connectivity and data fusion. *Hydrol. Earth Syst. Sci.* 26 (19), 5163–5184. <http://dx.doi.org/10.5194/hess-26-5163-2022>, URL: <https://hess.copernicus.org/articles/26/5163/2022/>.
- Sun, A.Y., Li, Z., Lee, W., Huang, Q., Scanlon, B.R., Dawson, C., 2023. Rapid flood inundation forecast using Fourier neural operator. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. pp. 3733–3739.
- Sun, K., Zhu, Z., Lin, Z., 2021. AdaGCN: AdaBoosting graph convolutional networks into deep models. *arXiv:1908.05081* URL: <https://arxiv.org/abs/1908.05081>.
- Tabari, H., 2020. Climate change impact on flood and extreme precipitation increases with water availability. *Sci. Rep.* 10 (1), 13768. <http://dx.doi.org/10.1038/s41598-020-70816-2>.
- Taghizadeh, M., Zandsalimi, Z., Nabian, M.A., Shafiee-Jood, M., Alemazkoor, N., 2025a. Interpretable physics-informed graph neural networks for flood forecasting. *Computer-Aided Civ. Infrastruct. Eng.* 1–21. <http://dx.doi.org/10.1111/mice.13484>.
- Taghizadeh, M., Zandsalimi, Z., Shafiee-Jood, M., Alemazkoor, N., 2025b. Multi-fidelity graph neural networks for efficient and accurate flood hazard mapping. *Environ. Model. Softw.* 193, 106654. <http://dx.doi.org/10.1016/j.envsoft.2025.106654>, URL: <https://www.sciencedirect.com/science/article/pii/S136481522500338X>.
- Teng, J., Jakeman, A., Vaze, J., Croke, B., Dutta, D., Kim, S., 2017. Flood inundation modelling: A review of methods, recent advances and uncertainty analysis. *Environ. Model. Softw.* 90, 201–216. <http://dx.doi.org/10.1016/j.envsoft.2017.01.006>, URL: <https://www.sciencedirect.com/science/article/pii/S1364815216310040>.
- Thangamuthu, A., Kumar, G., Bishnoi, S., Bhattoo, R., Krishnan, N.M.A., Ranu, S., 2022. Unravelling the performance of physics-informed graph neural networks for dynamical systems. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (Eds.), *Advances in Neural Information Processing Systems*. 35, Curran Associates, Inc., pp. 3691–3702, URL: [https://proceedings.neurips.cc/paper\\_files/paper/2022/file/17b598fda495256bef6785c2b76c3217-Paper-Datasets\\_and\\_Benchmarks.pdf](https://proceedings.neurips.cc/paper_files/paper/2022/file/17b598fda495256bef6785c2b76c3217-Paper-Datasets_and_Benchmarks.pdf).

- United Nations Office for Disaster Risk Reduction (UNDRR), Centre for Research on the Epidemiology of Disasters (CRED), 2020. The Human Cost of Disasters: An Overview of the Last 20 Years (2000–2019). URL: <https://www.undrr.org/publication/human-cost-disasters-overview-last-20-years-2000-2019>.
- Vacondio, R., Dal Palù, A., Mignosa, P., 2014. GPU-enhanced finite volume shallow water solver for fast flood simulations. *Environ. Model. Softw.* 57, 60–75. <http://dx.doi.org/10.1016/j.envsoft.2014.02.003>, URL: <https://www.sciencedirect.com/science/article/pii/S136481521400053X>.
- Varbella, A., Amara, K., Gjorgiev, B., El-Assady, M., Sansavini, G., 2024. PowerGraph: A power grid benchmark dataset for graph neural networks. In: Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., Zhang, C. (Eds.), *Advances in Neural Information Processing Systems*. 37, Curran Associates, Inc., pp. 110784–110804, URL: [https://proceedings.neurips.cc/paper\\_files/paper/2024/file/c7caf017cbbca1f4b368ffdc7bb8f319-Paper-Datasets\\_and\\_Benchmarks\\_Track.pdf](https://proceedings.neurips.cc/paper_files/paper/2024/file/c7caf017cbbca1f4b368ffdc7bb8f319-Paper-Datasets_and_Benchmarks_Track.pdf).
- Vashishth, S., Sanyal, S., Nitin, V., Talukdar, P., 2020. Composition-based multi-relational graph convolutional networks. *arXiv:1911.03082* URL: <https://arxiv.org/abs/1911.03082>.
- Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y., 2018. Graph attention networks. *arXiv:1710.10903* URL: <https://arxiv.org/abs/1710.10903>.
- Victor, O.S., Costa Rocha, P.A., Scott, J., Thé, J.V.G., Gharabaghi, B., 2023. A new graph-based deep learning model to predict flooding with validation on a case study on the humber river. *Water* 15 (10), 1827. <http://dx.doi.org/10.3390/w15101827>, URL: <https://doi.org/10.3390/w15101827>.
- Wang, Y., Fang, Z., Hong, H., Peng, L., 2020. Flood susceptibility mapping using convolutional neural network frameworks. *J. Hydrol.* 582, 124482. <http://dx.doi.org/10.1016/j.jhydrol.2019.124482>, URL: <https://www.sciencedirect.com/science/article/pii/S002216941931217X>.
- Wang, B., Luo, X., Zhang, F., Yuan, B., Bertozzi, A.L., Brantingham, P.J., 2018. Graph-based deep modeling and real time forecasting of sparse spatio-temporal data. *arXiv:1804.00684* URL: <https://arxiv.org/abs/1804.00684>.
- Wang, K., Shen, Z., Huang, C., Wu, C.-H., Dong, Y., Kanakia, A., 2020. Microsoft academic graph: When experts are not enough. *Quant. Sci. Stud.* 1 (1), 396–413. <http://dx.doi.org/10.1162/qss.a.00021>, *arXiv:https://direct.mit.edu/qss/article-pdf/1/1/396/1760880/qss.a.00021.pdf*.
- Wang, M., Zheng, D., Ye, Z., Gan, Q., Li, M., Song, X., Zhou, J., Ma, C., Yu, L., Gai, Y., Xiao, T., He, T., Karypis, G., Li, J., Zhang, Z., 2020. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv:1909.01315* URL: <https://arxiv.org/abs/1909.01315>.
- Wang, Q., Zhu, S., 2025. Multi-station joint runoff forecasting using graph neural network coupled with spatial connectivity of hydrological stations. *Math. Geosci.* 57 (4), 759–797. <http://dx.doi.org/10.1007/s11004-024-10167-0>.
- Wei, W., Qiao, M., Jadav, D., 2023. GNN-ensemble: Towards random decision graph neural networks. *arXiv:2303.11376* URL: <https://arxiv.org/abs/2303.11376>.
- Wright, D.B., 2018. Rainfall information for global flood modeling. In: *Global Flood Hazard*. American Geophysical Union (AGU), pp. 17–42. <http://dx.doi.org/10.1002/9781119217886.ch2>, URL: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/9781119217886.ch2>.
- Wu, Z., Pan, S., Long, G., Jiang, J., Zhang, C., 2019. Graph WaveNet for deep spatial-temporal graph modeling. *arXiv:1906.00121* URL: <https://arxiv.org/abs/1906.00121>.
- Wu, Z., Ramsundar, B., Feinberg, E.N., Gomes, J., Geniesse, C., Pappu, A.S., Leswing, K., Pande, V., 2018. MoleculeNet: A benchmark for molecular machine learning. *arXiv:1703.00564* URL: <https://arxiv.org/abs/1703.00564>.
- Xie, S., Wu, W., Mooser, S., Wang, Q., Nathan, R., Huang, Y., 2021. Artificial neural network based hybrid modeling approach for flood inundation modeling. *J. Hydrol.* 592, 125605. <http://dx.doi.org/10.1016/j.jhydrol.2020.125605>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169420310660>.
- Xing, L., Sela, L., 2022. Graph neural networks for state estimation in water distribution systems: Application of supervised and semisupervised learning. *J. Water Resour. Plan. Manag.* 148 (5), 04022018. [http://dx.doi.org/10.1061/\(ASCE\)WR.1943-5452.0001550](http://dx.doi.org/10.1061/(ASCE)WR.1943-5452.0001550), URL: <https://ascelibrary.org/doi/abs/10.1061/%28ASCE%29WR.1943-5452.0001550>.
- Xiong, J., Cui, R., Li, Z., Zhang, W., Zhang, R., Fu, Z., Liu, X., Li, Z., Chen, K., Zheng, M., 2024. Transfer learning enhanced graph neural network for aldehyde oxidase metabolism prediction and its experimental application. *Acta Pharm. Sin.* B 14 (2), 623–634. <http://dx.doi.org/10.1016/j.apsb.2023.10.008>, URL: <https://www.sciencedirect.com/science/article/pii/S2211383523004021>.
- Xu, K., Hu, W., Leskovec, J., Jegelka, S., 2019. How powerful are graph neural networks?. *arXiv:1810.00826* URL: <https://arxiv.org/abs/1810.00826>.
- Xu, Q., Shi, Y., Zhao, J., Zhu, X.X., 2025. FloodCastBench: A large-scale dataset and foundation models for flood modeling and forecasting. *Sci. Data* 12 (1), 431. <http://dx.doi.org/10.1038/s41597-025-04725-2>.
- Xue, L., Zhu, Y., 2025. Dynamic hydrological flow prediction with self-iterative spatiotemporal graph neural network: Modeling long- and short-period topological dynamics. *J. Hydrol.* 663, 134122. <http://dx.doi.org/10.1016/j.jhydrol.2025.134122>, URL: <https://www.sciencedirect.com/science/article/pii/S002216942501460X>.
- Yang, F., Ding, W., Zhao, J., Song, L., Yang, D., Li, X., 2024. Rapid urban flood inundation forecasting using a physics-informed deep learning approach. *J. Hydrol.* 643, 131998. <http://dx.doi.org/10.1016/j.jhydrol.2024.131998>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169424013945>.
- Yang, S., Zhang, Y., Zhang, Z., 2023. Runoff prediction based on dynamic spatiotemporal graph neural network. *Water* 15 (13), <http://dx.doi.org/10.3390/w15132463>, URL: <https://www.mdpi.com/2073-4441/15/13/2463>.
- Yariyan, P., Avand, M., Abbaspour, R.A., Haghighi, A.T., Costache, R., Ghorbanzadeh, O., Janizadeh, S., Blaschke, T., 2020. Flood susceptibility mapping using an improved analytic network process with statistical models. *Geomatics, Nat. Hazards Risk* 11 (1), 2282–2314. <http://dx.doi.org/10.1080/19475705.2020.1836036>, URL: <https://doi.org/10.1080/19475705.2020.1836036>.
- Ying, R., Bourgeois, D., You, J., Zitnik, M., Leskovec, J., 2019. GNNExplainer: Generating explanations for graph neural networks. *arXiv:1903.03894* URL: <https://arxiv.org/abs/1903.03894>.
- Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W.L., Leskovec, J., 2018. Graph convolutional neural networks for web-scale recommender systems. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. In: KDD 2018, ACM, pp. 974–983. <http://dx.doi.org/10.1145/3219819.3219890>, URL: <http://dx.doi.org/10.1145/3219819.3219890>.
- Yu, B., Yin, H., Zhu, Z., 2018. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. In: IJCAI-2018, International Joint Conferences on Artificial Intelligence Organization, pp. 3634–3640. <http://dx.doi.org/10.24963/ijcai.2018/505>, URL: <http://dx.doi.org/10.24963/ijcai.2018/505>.
- Zeng, H., Zhou, H., Srivastava, A., Kannan, R., Prasanna, V., 2020. GraphSAINT: Graph sampling based inductive learning method. *arXiv:1907.04931* URL: <https://arxiv.org/abs/1907.04931>.
- Zhang, Y., Pal, S., Coates, M., Üstebay, D., 2018. Bayesian graph convolutional neural networks for semi-supervised classification. *arXiv:1811.11103* URL: <https://arxiv.org/abs/1811.11103>.
- Zhang, C., Song, D., Huang, C., Swami, A., Chawla, N.V., 2019. Heterogeneous graph neural network. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '19, Association for Computing Machinery, New York, NY, USA, pp. 793–803. <http://dx.doi.org/10.1145/3292500.3330961>.
- Zhang, Z., Tian, W., Lu, C., Liao, Z., Yuan, Z., 2024. Graph neural network-based surrogate modelling for real-time hydraulic prediction of urban drainage networks. *Water Res.* 263, 122142. <http://dx.doi.org/10.1016/j.watres.2024.122142>, URL: <https://www.sciencedirect.com/science/article/pii/S0043135424010418>.
- Zhang, S., Zhang, D., Huang, G., Wan, J., Kang, Y., Jiang, D., Xia, B., Zhao, Z., Liu, R., 2025. A novel framework for multi-step water level predicting by spatial-temporal deep learning models based on integrated physical models. *J. Hydrol.* 661, 133683. <http://dx.doi.org/10.1016/j.jhydrol.2025.133683>, URL: <https://www.sciencedirect.com/science/article/pii/S0022169425010212>.
- Zhangke, H., Dantong, L., Ning, J., Xianwu, S., Wei, Z., Weiping, W., 2026. Prediction of the typhoon-induced tidal level at multiple stations using dynamic graph neural network. *Ocean Eng.* 349, 124180. <http://dx.doi.org/10.1016/j.oceaneng.2026.124180>, URL: <https://www.sciencedirect.com/science/article/pii/S0029801826000144>.
- Zhao, C., Li, X., Cai, Y., 2025. A power grid topology detection method based on edge graph attention neural network. *Electr. Power Syst. Res.* 239, 111219. <http://dx.doi.org/10.1016/j.epsr.2024.111219>, URL: <https://www.sciencedirect.com/science/article/pii/S0378779624011052>.
- Zhao, Q., Zhu, Y., Shu, K., Wan, D., Yu, Y., Zhou, X., Liu, H., 2020. Joint spatial and temporal modeling for hydrological prediction. *IEEE Access* 8, 78492–78503. <http://dx.doi.org/10.1109/ACCESS.2020.2990181>.
- Zheng, L., Zuo, Z., Wang, W., Dong, C., Momma, M., Sun, Y., 2021. Heterogeneous graph neural networks with neighbor-SIM attention mechanism for substitute product recommendation. URL: <https://www.amazon.science/publications/heterogeneous-graph-neural-networks-with-neighbor-sim-attention-mechanism-for-substitute-product-recommendation>.
- Zhou, Y., Duan, Y., Yao, H., Li, X., Li, S., 2025. Incorporating hydrological constraints with deep learning for streamflow prediction. *Expert Syst. Appl.* 259, 125379. <http://dx.doi.org/10.1016/j.eswa.2024.125379>, URL: <https://www.sciencedirect.com/science/article/pii/S0957417424022462>.
- Zhu, R., Zhao, K., Yang, H., Lin, W., Zhou, C., Ai, B., Li, Y., Zhou, J., 2019. AliGraph: a comprehensive graph neural network platform. *Proc. the VLDB Endow.* 12 (12), 2094–2105. <http://dx.doi.org/10.14778/3352063.3352127>.
- Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., He, Q., 2021. A comprehensive survey on transfer learning. *Proc. IEEE* 109 (1), 43–76. <http://dx.doi.org/10.1109/JPROC.2020.3004555>.