Improving large-basin streamflow simulation using a modular, differentiable, learnable graph model for routing

Tadd Bindas[1], Wen-Ping Tsai[2], Jiangtao Liu[1], Farshid Rahmani[1], Dapeng Feng[1], Yuchen Bian[3], Kathryn Lawson[1], and Chaopeng Shen[1]

[1]Pennsylvania State University
[2]National Cheng Kung University
[3]Amazon Search

November 24, 2022

Abstract

Recently, runoff simulations in small, headwater basins have been improved by methodological advances such as deep learning (DL). Hydrologic routing modules are typically needed to simulate flows in stem rivers downstream of large, heterogeneous basins, but obtaining suitable parameterization for them has previously been difficult. It is unclear if downstream daily discharge contains enough information to constrain spatially-distributed parameterization. Building on recent advances in differentiable modeling principles, here we propose a differentiable, learnable physics-based routing model. It mimics the classical Muskingum-Cunge routing model but embeds a neural network (NN) to provide parameterizations for Manning's roughness coefficient (n) and channel geometries. The embedded NN, which uses (imperfect) DL-simulated runoffs as the forcing data and reach-scale attributes as inputs, was trained solely on downstream hydrographs. Our synthetic experiments show that while channel geometries cannot be identified, we can learn a parameterization scheme for n that captures the overall spatial pattern. Training on short real-world data showed that we could obtain highly accurate routing results for both the training and inner, untrained gages. For larger basins, our results are better than a DL model assuming homogeneity or the sum of runoff from subbasins. The parameterization learned from a short training period gave high performance in other periods, despite significant bias in runoff. This is the first time an interpretable, physics-based model is learned on the river network to infer spatially-distributed parameters. The trained n parameterization can be coupled to traditional runoff models and ported to traditional programming environments.

Improving Large-Basin Streamflow Simulation Using a Modular, Differentiable, Learnable Graph Model for Routing

Tadd Bindas[1], Wen-Ping Tsai[2], Jiangtao Liu[1], Farshid Rahmani[1], Dapeng Feng[1], Yuchen Bian[3], Kathryn Lawson[1], Chaopeng Shen*[,1]

[1] Civil and Environmental Engineering, The Pennsylvania State University, PA
[2] Hydraulic and Ocean Engineering, National Cheng Kung University, Tainan City
[3] Amazon Search, Palo Alto, CA

* Corresponding author: Chaopeng Shen, cshen@engr.psu.edu

Abstract

Recently, runoff simulations in small, headwater basins have been improved by methodological advances such as deep learning (DL). Hydrologic routing modules are typically needed to simulate flows in stem rivers downstream of large, heterogeneous basins, but obtaining suitable parameterization for them has previously been difficult. It is unclear if daily downstream discharge contains enough information to constrain spatially-distributed parameterization. We propose a differentiable, learnable physics-based routing model based on recent advances in differentiable modeling principles. It mimics the classical Muskingum-Cunge routing model but embeds a neural network (NN) to provide parameterizations for Manning's roughness *n* and channel geometries. The embedded NN, which uses (imperfect) DL-simulated runoffs as the forcing data and reach-scale attributes as inputs, was trained solely on downstream hydrographs. Our synthetic experiments show that while channel geometries cannot be identified, we can learn a parameterization scheme for *n* that captures the overall spatial pattern. Training on short real-world data showed that we could obtain highly accurate routing results for the training and inner, untrained gages. Our results for larger basins (>2,000 km^2) are better than a DL model assuming homogeneity or the sum of runoff from subbasins. The *n* parameterization learned from a short training period gave a high performance in other periods, despite significant bias in the runoff. This is the first time an interpretable, physics-based model is learned on the river network to infer spatially-distributed parameters. The trained *n* parameterization can be coupled to traditional runoff models and ported to traditional programming environments.

Main points:

1. A differentiable routing model can learn routing parameterization from discharge to support long-term flow simulation in large rivers.
2. Our synthetic case retrieved the assumed roughness coefficients while the real case produced estimates consistent with our understanding.
3. For basins >2,000 km^2, our framework outperforms deep learning models that assume homogeneity, despite bias in the runoff forcings.

1. Introduction

Riverine floods are intrinsically linked with stream channel characteristics and pose a major risk to human safety and infrastructure (Douben, 2006; François et al., 2019; IPCC, 2012; Koks & Thissen, 2016). Riverine floods along large stem rivers occur when the peak flow rate exceeds the stem river conveyance capacity. The timing of flood convergence and thus peak flood rates are influenced by the channel's geometries and flow resistance properties (Candela et al., 2005; Kalyanapu et al., 2009). In recent years, we witnessed many deadly riverine floods, e.g., in the Mississippi River, USA (Rice, 2019), India (France-Presse, 2022), while such disasters are expected to rise significantly under projected future climates (Dottori et al., 2018; Prein et al., 2017; Winsemius et al., 2016). The ability to better account for flood convergence and streamflow processes is urgently needed to help us better inform society of stem river flood magnitudes and timing, which can save lives and mitigate damages. Besides their importance on floods, river channel characteristics and flow velocity also have major implications for aquatic ecosystems (Ghanem et al., 1996; Leclerc et al., 1995; Papaioannou et al., 2020).

In hydrologic modeling, routing describes how the stream network receives and conveys runoff from basins while accounting for mass balances and the speed of flood wave propagation (Mays, 2010). Some routing modules are based on the principles of continuity and assume constitutive discharge-flow area or discharge-flow velocity relationships. For example, the widely-applied Muskingum-Cunge (MC) (Cunge, 1969) routing method is a center-in-space center-in-time finite difference solution to the continuity equation, assuming a prismatic flood wave as the constitutive relationship. In some other cases, the momentum equation is solved in conjunction with the continuity equation (Ji et al., 2019) with a range of simplifying assumptions, e.g., ignoring inertia (Shen & Phanikumar, 2010), ignoring both inertia and pressure gradient (only slope remaining) (Mizukami et al., 2016), with sometimes additional formulations to handle effects scale, e.g., Li et al. (2013). These models have parameters that need to be determined from lookup tables or calibration.

While routing parameters often rank among the important ones for discharge simulation (Khorashadi Zadeh et al., 2017; L. Liu et al., 2022), it has been difficult to parameterize them at large scales, especially in a way to both sensibly represent basin-internal spatial heterogeneity and adapt to discharge data. Using traditional roughness values tabulated for various land covers (Arcement & Schneider, 1989) requires in-situ scouting, e.g., to determine if channels have pools, weeds, grass, etc., which is currently impractical for large-scale applications. Without scouting, available land cover data are only available for the floodplain and not for the main channel, which contains the water for the majority of the time and can have distinctly different characteristics from the floodplain. Many calibration exercises, e.g. (Khorashadi Zadeh et al., 2017; L. Liu et al., 2022; Mizukami et al., 2016), used only one set of parameters for an entire basin, neglecting fine-scale spatial heterogeneity in river-reach characteristics. Some studies have employed Manning's roughness, *n* (a coefficient representing a channel's resistance to flow), as a linear function of river depth or other characteristics (Getirana et al., 2012; H.-Y. Li et al., 2022), but it is unclear if these relationships could optimally absorb information from available data. We may be able to find more fine-grained relationships given recent progress in differentiable programming, to be discussed below.

While the accuracy of basin rainfall-runoff models has improved substantially in recent years with machine learning (ML) (Adnan et al., 2021; Feng et al., 2020; Kratzert et al., 2019; Sun et al., 2022; Xiang et al., 2020), process-based models, or models with ML components (Feng, Beck, et al., 2022; Feng, Liu, et al., 2022), the routing modules have not similarly benefited. Neural networks (NNs) like long short-term memory (LSTM), GraphWaveNet (Sun et al., 2021) or convolutional networks (Duan et al., 2020), while very generic, have demonstrated their prowess in learning hydrologic dynamics from big data. They are applicable not only to streamflow hydrology but also variables across the entire hydrologic cycle (Shen et al., 2021; Shen & Lawson, 2021) such as soil moisture (Fang et al., 2017, 2019; J. Liu et al., 2022; O & Orth, 2021), groundwater (Wunsch et al., 2022), snow (Meyal et al., 2020), longwave radiation (Zhu et al., 2021), and water quality parameters (He et al., 2022; Hrnjica et al., 2021; Lin et al., 2022; Rahmani, Lawson, et al., 2021; Zhi et al., 2021). However, these approaches are mostly suitable for relatively homogeneous headwater basins; spatial heterogeneities in forcings and basin characteristics are generally not captured well, and large basins often turn out to have poorer performance for LSTM models.

A recent development in integrating ML with physical understanding is differentiable, learnable process-based models, which can approach the performance of LSTM models but also provide interpretable fluxes and states (Feng, Liu, et al., 2022). By connecting deep networks to reimplemented process-based models (or their neural network surrogates), Tsai et al. (2021) obtained an NN-based parameterization pipeline that infers physical parameters for process-based models. The keyword is "differentiable" (as in differentiable programming), which means that the system allows gradient-tracking along all calculation steps such that gradient-based training of neural networks can be enabled. This critically enables the hybrid framework to learn complex and potentially unknown functions from big data while keeping physical formulations. Feng, Beck, et al. (2022) further found that this type of differentiable model can extrapolate better than purely data-driven LSTM.

Nevertheless, it is unclear if differentiable computing is applicable to the highly-complex river graph. The river network forms a hierarchical graph, which is not unlike the graph networks for applications like social recommendations (Fan et al., 2019), but with a predefined spatial topology (due to a fixed river network) and a converging cascade. A complex river graph can have many nodes, which, when coupled with many time steps, could potentially lead to a training issue known as the vanishing gradient. It is unclear if such an issue would prevent a differentiable model from learning. It is also unclear if downstream discharge data alone has enough information to train a parameterization scheme, and the length of the training period required.

In this work, we created a novel differentiable modeling framework to perform routing and to learn a parameterization scheme for routing flows on the river network. Such a physically-based routing method has never been trained together with neural networks. An NN-based parameterization scheme for Manning's *n* and river bathymetry shape (*q*) is coupled to Muskingum-Cunge routing and is applied throughout the river network. We designed synthetic and real data experiments to answer the following research questions:

1. *Does a downstream hydrograph have enough information to identify n and q parameterization schemes?*
2. *Can a parameterization scheme for routing produce reliable results for long-term simulations for both trained and untrained gages?*
3. *What lengths of training periods are required to train a reliable parameterization scheme*?

Because our framework is built on physical principles and estimates widely-used *n*, it can be easily ported to work with other models. For example, the trained NN and the weights can be loaded into Fortran or C programs to support traditional hydrologic models or routing schemes, e.g. (H. Li et al., 2013; Mizukami et al., 2016). It does not have to be limited to a machine learning platform.

2. Data and Methods

2.1. Overview

As an overview, we used a previously pretrained LSTM model to produce daily runoff estimates for Level-10 Hydrologic Cataloging Unit (HUC10) watersheds (Figure 1a) which were then disaggregated to hourly time steps and routed throughout the river network using the proposed differentiable routing model (Figure 1b). This model can also be perceived to as a physics-guided graph neural network (GNN) from the ML perspective. We embedded an NN as a parameterization scheme for the routing model and trained the whole model on the downstream hydrograph.

In the following, we sequentially describe the pretrained LSTM, the creation of the river graph, the neural network used to approximate Manning's n, and our synthetic experiments. We first ran synthetic experiments to verify if such a framework could recover assumed n and q parameterizations. We then trained the framework on real-world discharge data and compared the results to some alternatives, including an LSTM assuming the entire basin as homogeneous, a summation of runoff inputs, and routing with a spatially-constant n value of 0.02. We then tested the conditions needed to obtain reliable routing parameters for untrained time periods using several models with short training periods.

2.2. Pretrained LSTM

A model based on the long short-term memory (LSTM) algorithm (Hochreiter & Schmidhuber, 1997) was used to estimate runoff inputs in the Muskingum-Cunge equation and provide a benchmark for the routing model. This LSTM model was similar to those from previous streamflow and water quality (Feng et al., 2020; Ouyang et al., 2021; Rahmani, Lawson, et al., 2021; Rahmani, Shen, et al., 2021). To briefly summarize, the LSTM model used a combination of basin-averaged attributes, daily meteorological forcings, and observations as inputs, and outputs daily basin discharge. Meteorological forcings (total annual precipitation, downward long-wave radiation flux, downward short-wave radiation flux, pressure, temperature) were obtained from the NASA NLDAS-2 Forcing Data set (Xia et al., 2009, 2012). We selected 29 basin attributes (Table A1) similar to those chosen in previous LSTM studies (Ouyang et al., 2021). Consistent with Ouyang et al. (2021), we focused on training the LSTM on 3213 gages selected from the USGS Geospatial Attributes of Gages for Evaluating Streamflow II (GAGES-II) dataset (Falcone, 2011) with input data between 1990/01/01 - 1999/12/31. We developed the workflow to obtain forcing data and inputs seamlessly for any small basin in the CONUS. In this case we extracted data from HUC8 subbasins and HUC10 watersheds to gather inputs to train our LSTM model and predict discharge, respectively.

The LSTM model was trained in the same way as in our previous work, on >3000 natural and human-disturbed basins (Ouyang et al., 2021) across the conterminous United States (CONUS) to generate accurate and seamless predictions. When evaluated on the gaging stations in the study domain, the model obtained a domain-wide median daily NSE of 0.7849 for eight gauging stations. After training during the period of 1990/01/01 - 1999/12/31, a forward run was conducted from 2000/01/01-2009/12/31 to predict discharge for the 17 HUC10 watersheds in the study domain, using HUC10-averaged attributes for each HUC10 basins:

$$Q' = LSTM(x_{HUC10}, A_{HUC10}) \tag{1}$$

where Q' [m^3/s] is the daily runoff for the HUC10 basin, and x_{HUC10} and A_{HUC10} are HUC10-averaged atmospheric forcings and static attribute variables, respectively. This LSTM was only used in an inference mode to enable a modular model design and was not further tuned while training our routing model (Figure 1b). We first carefully shifted the LSTM-produced runoff outputs by 5 hours to account for the time zone differences between the forcing data (recorded using UTC) and USGS streamflow (recorded in UTC-5). Then, we applied an additional shift to avoid a double routing issue as implicitly, the LSTM-estimated runoff has already considered in-channel flow at the subbasin scale as it is trained on subbasin-outlet hydrographs. To keep things simple for this initial exploration, we pushed the LSTM-produced hydrograph back by τ hours as an anti-routing procedure to avoid routing the streamflow twice. τ was a hyperparameter, for which we used the value of $\tau = 9$ in all of our routing models. This value was calculated through various trials to maximize NSE and minimize τ while retaining meaningful parameter values that fit with literature constraints. More complicated procedures could be employed in the future, but this simple approach appeared to work decently here.

2.3. River Graph and Discharge Interpolation

We constructed a river network (or graph) for our area of interest, the Juniata River Basin (JRB) (Figure 2), by obtaining nodes (junction points, of which there were 544) and edges (river reaches, of which there were 582) from the National Hydrography Dataset (NHDplus v2) (HorizonSystems, 2016; Moore & Dewald, 2016) which describes both the topology and some attributes of the river reaches. To reduce computational demand, a subset of river reaches was selected from all the available river reaches based on applying a stream density threshold (total stream length/watershed area). We selected rivers with the longest length until a specific stream density was reached (0.2 km/km^2). Next, we calculated slope and sinuosity by overlaying NHD v2.0 with 10-m resolution digital elevation data (USGS ScienceBase-Catalog, 2022). We then discretized the selected rivers using a uniform step size of ~2,000 m to ensure the stability of the Muskingum-Cunge equation. Previous work describes the bulk of the extraction procedure that prepares input data for a physically-based surface-subsurface processes model (Ji et al., 2019; Shen et al., 2013, 2014, 2016; Shen & Phanikumar, 2010). Along with the river graph, we computed a mass transfer matrix to determine the fraction of each HUC10 watershed that flowed laterally into its corresponding river segment. This matrix enables the runoff generated from the basins to be applied as source terms with the river reaches.

Runoff estimates and discharge observations for the JRB were available on a daily, but not hourly, scale. Because Muskingum-Cunge (MC) routing needs to operate on smaller time steps, we quadratically interpolated daily data into hourly time steps. For training and evaluating the routing model, we collected observed discharge data for nodes intersecting United States Geological Survey (USGS) GAGES-II monitoring stations, locating a total of eight stations. Only some time periods of the most downstream station were used for training, and other stations were only used for evaluation. The observed discharge data were disaggregated using quadratic interpolation similar to LSTM-predicted runoffs. Training periods were selected based on times when the LSTM had high accuracy, and when high flashiness was observed in yearly hydrographs. Two eight-week periods, 02/01-03/29 and 11/01-12/26 fit these requirements and were used to provide training data across multiple years (2001, 2005, 2007, and 2008) for a total of eight trained models.

The hydrograph at the furthest downstream JRB gage, USGS gage 01563500 [node 4809 in our graph] on the Juniata River at Mapleton Depot, PA, was chosen as the training target. This reach has a catchment area of 5,212 km^2 contributed from the 582 reaches upstream. Seven USGS gages are located upstream of this node which enables further validation of the simulations.

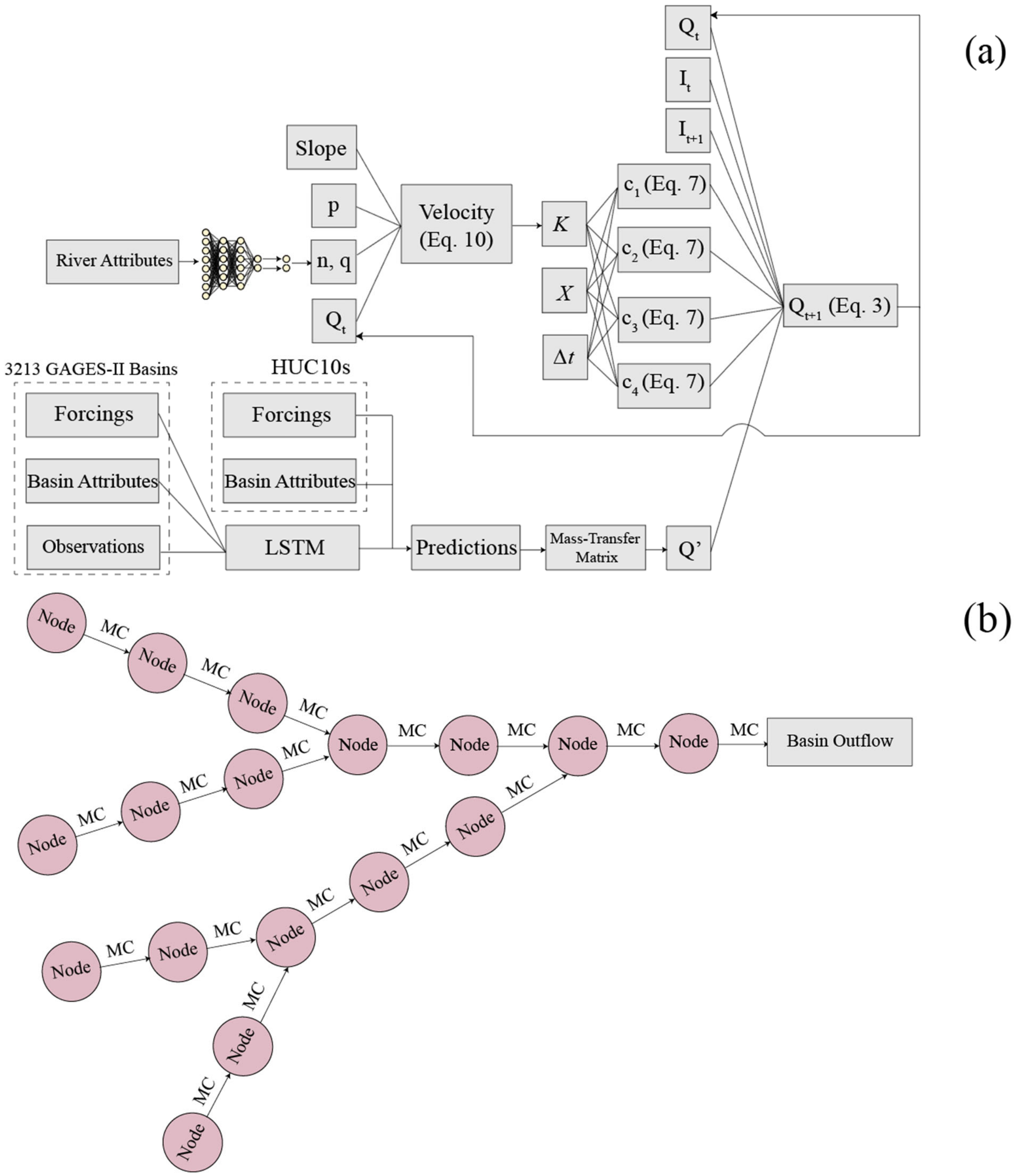

Figure 1: (a) An overview of how inputs move through our workflow to eventually be run through Muskingum-Cunge (MC). After calculating Q_{t+1}, the discharge value is then used again to predict the next node's discharge. (b) An illustration of how we traverse the graph using MC to make a discharge prediction for the final node. Our case study has 582 reaches and 544 nodes.

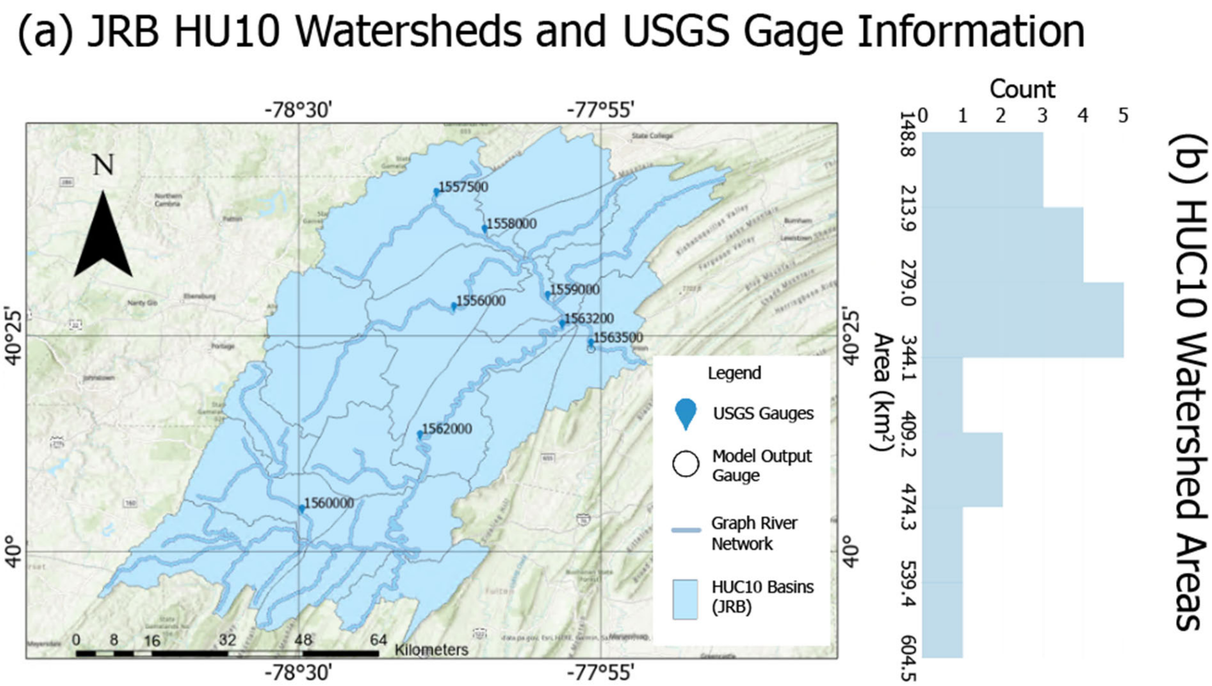

Figure 2: (a) A map of the Juniata River Basin's river network and HUC10 watersheds. Each number corresponds to a USGS gage. (b) A histogram showing the distribution of HUC10 watersheds in the JRB. The x-axis shows the distribution of the HUC10 watershed areas, and the y-axis shows the number of HUC10s that fall within the area ranges.

2.4. Differentiable Routing Model

Our parameterization scheme consists of a feed-forward multilayer perceptron (MLP) neural network with two hidden layers (altogether, three matrix multiplications) and a sigmoid activation function for the output layer. The MLP's outputs are physical parameters used in the MC river routing module. The MLP accepts an array of attributes (together abbreviated as ***A*** and each attribute was normalized based on the range of its respective values) per reach (Table A2). The network outputs Manning's roughness coefficient *n*, and channel bathymetry shape coefficient *q*:

$$n, q = NN(A) \tag{2}$$

where *n* represents a channel's resistance to flow and *q* represents the shape of the channel's cross-sectional area. Since we assumed *n* and *q* to be constant in time for this study, the MLP is invoked once at the beginning of each epoch for all reaches. The weights of the MLP were updated using backpropagation and the Adam optimizer (Kingma & Ba, 2017).

The MC routing is run once for each river reach in the network per time step:

$$Q_{t+1} = c_1 I_{t+1} + c_2 I_t + c_3 Q_t + c_4 Q' \tag{3}$$

where I represents inflow and Q represents discharge and c_1-c_4 are coefficients explained below. Thus, for *n* reaches, MC will be run *m* times in each time step, from upstream to downstream, sequentially. To enable differentiable computing, we implemented MC and Equation 3 on PyTorch, a machine learning platform. From a hydrologic perspective, this is essentially a river routing module with the ability to train the MLP described in Equation 2. From an ML perspective, it can be considered a graph neural network constrained by the topology in the river network, mass conservation, and the MC routing method as the imposed law for each edge.

The MC method calculates several coefficients using hydraulic properties K and X (Equations 4-7):

$$c_1 = \frac{\Delta t - 2KX}{2K(1-X) + \Delta t} \tag{4}$$

$$c_2 = \frac{\Delta t + 2KX}{2K(1-X) + \Delta t} \tag{5}$$

$$c_3 = \frac{2K(1-X) - \Delta t}{2K(1-X) + \Delta t} \tag{6}$$

$$c_4 = \frac{2\Delta t}{2K(1-X) + \Delta t} \tag{7}$$

We chose an hourly time step (Δt) and a weighting coefficient (X) of 0.3 with K representing travel time. To estimate K, we divided the length of the reach by its velocity (v [m /s]): $K=L/v$. Since v varies over time, it needs to be updated in each time step with connection to discharge Q, which was done with the help of a constitutive relationship to close the equations. For this, the core geometric assumption we make is that there is a power-law relationship between stream width (w [m]) and depth (d [m]):

$$w = pd^q \tag{8}$$

where p [m] and q [-] are parameters that are potentially spatially heterogeneous. p and q represent the shape of the channel's cross-sectional area. For a rectangular channel, q=0, and for a triangular channel, q=1. The cross-sectional area A [m^2] is the integral of w over d (Equation 9 & Figure 1a). The NN described earlier (Equation 2) outputs q. To simplify the task (and also because it is not sensitive based on our observations), we assumed p=21 based on some preliminary data fitting to USGS hydraulic geometries from field surveys of gages in the JRB. Note that even though we make this assumption here for model completeness, we do not posit that q is invertible from available data because it may not be that significant for the downstream discharge.

$$A = \int_0^d w \, \partial d = \int_0^d p d^q \, \partial d = \frac{p d^{q+1}}{q+1} \tag{9}$$

Reorganizing Equation 9, we have a function that estimates d from Q (Equation 10a), given the coefficients from the NN. With d, p, *and* q, we can estimate A, v, and K using Equation 10b-d, which closes the equations.

$$(a)\ d = \left[\frac{Q_t n(q+1)}{p S_0^{\frac{1}{2}}}\right]^{\frac{3}{5+3q}}; \quad (b)\ A = \frac{p d^{q+1}}{q+1}; \quad (c)\ v = \frac{Q_t}{A}; \quad (d)\ K = \frac{length}{V} \tag{10}$$

Here, S_0 represents reach slope, Q_t represents the discharge entering the reach at time t, and length is the length of the reach. represents the discharge entering the reach at time t, and length is the length of the reach.

The Q' values in the Muskingum-Cunge equation (Equation 3) were obtained from the pretrained LSTM as described above, multiplied by the mass transfer matrix. Discharge outputs from the final node of the graph network were run through a MSE function to calculate loss prior to gradient descent and backwards propagation.

Hyperparameters and training period size for our differentiable routing model were chosen through repetitive trial and error based on the training period. These trials led us to choose a hidden size of 6 for our MLP, a training size of eight weeks, and 50 and 100 epochs for synthetic and real data experiments, respectively. Since our differentiable model at t=0 assumes no inflow to the river network, relying exclusively on Q' for flow inputs, a period of 72 hours is employed to warm up the model states in the river network and the loss function is not calculated within this period.

2.5. Experiments and tests

We first ran multiple synthetic parameter recovery experiments to check if the dataset and the framework could indeed recover assumed relationships with small training periods. Our first experiment tested if we could correctly recover a single, constant set of assumed values for both *n* and *q* for the whole river network. Thus, there are only two degrees of freedom. In our second experiment, we assumed constant *n* throughout the reaches but set the trained model as *n,q = NN(A)* (Equation 2) so that the *n, q* can be different from reach to reach. In this case, ideally, the NN would learn to output a constant value regardless of what the inputs are. Our third synthetic experiments examined if we could retrieve simple assumed relationships (inverse-linear or power-law) [Equation 11-12] between *n*, *q,* and drainage area (DA), given that the MLP had far more inputs than just DA. The trained model is still Equation 2 as we assumed we did not know the functional relationship *a priori*.

$$n = 0.06 - 8e^{-6}(DA) \tag{11}$$
$$q = 2 - 0.00018(DA)$$

$$n = \frac{0.0915}{(DA)^{0.131}} \tag{12}$$
$$q = \frac{2.1}{(DA)^{0.357}}$$

After the synthetic experiments, we trained our differentiable model (still training the parameterization NN as in equation 2) against observed USGS data to infer Manning's *n* and *q* for reaches within the river network. We employed eight weeks of training periods from different years and checked whether the resulting parameters led to satisfactory routing in other years at both the training gage and untrained gages, and under what conditions. We evaluated the model using both the downstream and the inner gages. We compared the results to three benchmarks: the LSTM that modeled the whole JRB as a uniform basin, a simple summation and time shift of Q’, and fixed Manning's *n* routing for the whole JRB reaches. Lastly, we trained the differentiable routing models on several time periods in different years to determine the sensitivity to the training periods.

3. Results and Discussion

In the following, we first discuss our synthetic experiments (Section 3.1) to showcase the potential to retrieve assumed parameters from our differentiable graph neural network. Next, we confront our model with LSTM-simulated runoff as observed streamflow at the furthest downstream gage, expand the training period to other time ranges, then apply our models to different years for observation (Section 3.2). Furthermore, we discuss the stability of our trained models over several years of testing (Section 3.3). Lastly, we analyze the Manning's *n* parameters recovered for the trained models and discuss their implications (Section 3.4).

3.1 Synthetic experiments

Our first synthetic experiment (with constant parameters and the degree of freedom is only 2) showed success recovering the assumed Manning's *n* values, but not the channel geometry parameter *q* (Table 1). Recovered *n* values were within a small range of the assumed ones, with minor fluctuations, while recovered *q* values mostly stayed around the initial guesses, slightly changed after a number of iterations. This result was consistent across 10 runs, each with different "synthetic truth" values for *n* and *q*. The training led *n* to the assumed values rapidly, typically within 20 epochs (an epoch is a forward run of the model for the Juniata River Basin (JRB) and a parameter update) (Figure A1). The non-identifiability of *q* was likely because *q* has only a small influence on the storage capacity of the stream and the simulated discharge is not sensitive to *q*, making *dL/dq* (where L is the loss function) negligible. Since *p* and *q* operate on the same equation and *q* alone was already not identifiable, we deduced that *p* was also non-invertible and thus used a constant value of 21 throughout. While it is a pity that *q* and *p* cannot be estimated, the results also implied that they would not influence the routing results noticeably. Thus, in our effort below, we focused on *n.*

Table 1: Results from the constant synthetic n and q parameter recovery experiments

Run	*n*			*q*		
	Initial Guess	Synthetic Truth	Recovered	Initial Guess	Synthetic Truth	Recovered
1	0.271	0.03	0.028	2.7	2	2.327
2	0.271	0.04	0.035	2.7	2	2.37
3	0.271	0.05	0.046	2.7	2.5	2.390

4	0.271	0.06	0.059	2.7	2.5	2.456
5	0.271	0.07	0.070	2.7	3	2.480
6	0.068	0.03	0.030	0.6	1.0	0.574
7	0.068	0.04	0.042	0.6	1.0	0.592
8	0.068	0.05	0.055	0.6	1.5	0.730
9	0.068	0.06	0.067	0.6	1.5	0.777
10	0.068	0.07	0.087	0.6	2.5	0.690

Our second synthetic experiment (assumed constant *n* to be recovered by NN(A)) showed that we were able to recover the mean value, but there was some scattering for the headwater reaches (Figure 3b, 3d). There were some visible differences between the synthetic hydrographs resulting from different assumed *n* values (comparing Figures 3a and 3c). This allowed the recovered n values to mostly center around the assumed value. However, the scattering of points toward the lower-DA part of Figures 3b and 3d alluded to the fact that the downstream discharge was not a strong constraint. *n* in different ranges can fluctuate around the mean to generate overall the same pattern as a constant *n* value.

In our third synthetic experiments, which were more consistent with our expectation of *n*, the simple functions could be roughly recovered for most of the reaches, while there may be increased uncertainty for the most downstream reaches (Figure 3). There are again noticeable differences in the hydrographs (Figures 3e & 3g) from previous ones When the power-law relationship was assumed, the hydrograph matched the synthetic one almost completely (Figure 3e) and the estimated *n* outputs from MLP overlapped to a great extent with the assumed one (Figure 3f). The headwater reaches (small-DA) showed a rapid decline in *n* with respect to increasing DA. In the middle ranges of DA, the curve followed the assumed one almost exactly. Toward the higher range of DA, the recovered values are lower than the assumed relationship but the deviation is not huge because the power-law formulation becomes flat in this range. Based on the closeness of hydrographs in Figure 3g, we do not imagine further optimization can bring significant improvement to the estimations. Similar to the two-constant-parameter retrieval experiment, the *q* parameter was not recoverable and thus is not shown here.

Based on these simple experiments, it seems training on the river graphs has some promise but also some limitations. It is promising because it is likely that *n* is related to DA which we show is, to some extent, recoverable. Discharge is a widely available variable so this method can be used to estimate *n* in many regions across the world. It is simultaneously challenging because, as we have a large number of reaches contributing to one gage, it is an underdetermined system. This method was not able to fully reproduce the drastic change in the low-DA range presumably because this sharp slope was too inconsistent with the rest of the curve and NNs generally do not output extreme values. It also ran into difficulty toward the high-DA range because there were simply far fewer reaches with large DA so their roles in routing were relatively little, making the curve unconstrained in this range. This experiment informed us we should not expect values of *n*, especially toward high-DA range, to be reliable, but the overall trend may have merit, especially when we also have other constraints. These findings formed the basis for the next stage of the work where we trained *n*=NN(***A***) for real-world data. We thus expected to extract the overall patterns of *n* distribution but for the recovered *q* not to be meaningful.

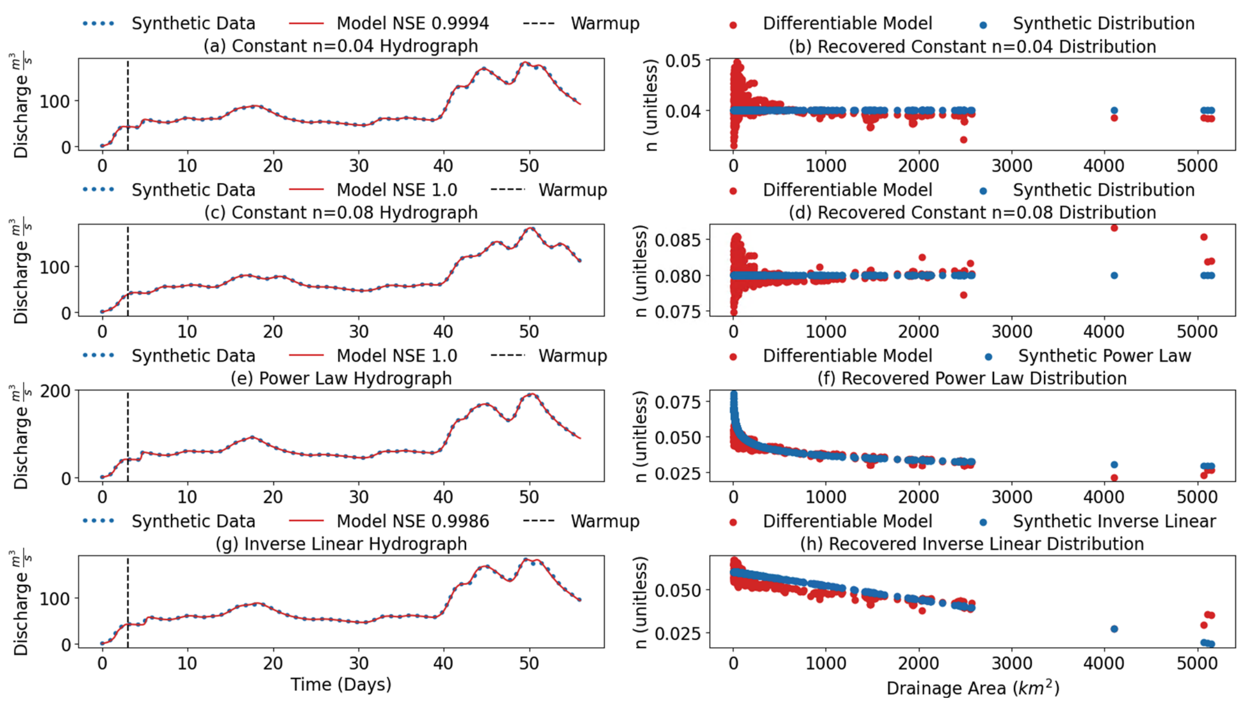

Figure 3: Synthetic discharge distribution experiments. (a, c, e, g) Synthetic and modeled discharge over time for various assumed relationships between Manning's n and drainage area. (b, d, f, h) Synthetic and modeled values of n with respect to drainage area. The NN can recover the overall pattern, but is not accurate near sharp changes or for reaches with large drainage areas. Each dot represents a 2-km river reach in the river network.

3.2. Training on eight weeks of real data

The real-world data experiment showed decent streamflow routing in the training period, showing improvements against approaches that did not employ the routing scheme despite having significant bias (Figure 4b). The hydrograph simulated by the differentiable routing model is, as expected, smoothed and delayed from the summation of runoffs during the training period. Unlike the direct summation of the runoff, which has a timing difference from the observation, the peaks of the routed hydrograph are placed almost exactly under the observed peaks, leading to a high training NSE of 0.834. We noticed a substantial bias in this training period. This is due to the mass-balance dictated by the MC formulation, which prevents the model from adding or removing mass to remove the bias. In traditional hydrologic model calibration, bias can be a significant concern as it sometimes distorts other parameters to reduce the bias. In this case, we found the model did a decent job even under bias, and rightfully focused on adjusting the timing of the flood waves. This is perhaps due to the fact that the allowable adjustments are limited to routing parameters, which blocked the model from distorting other processes.

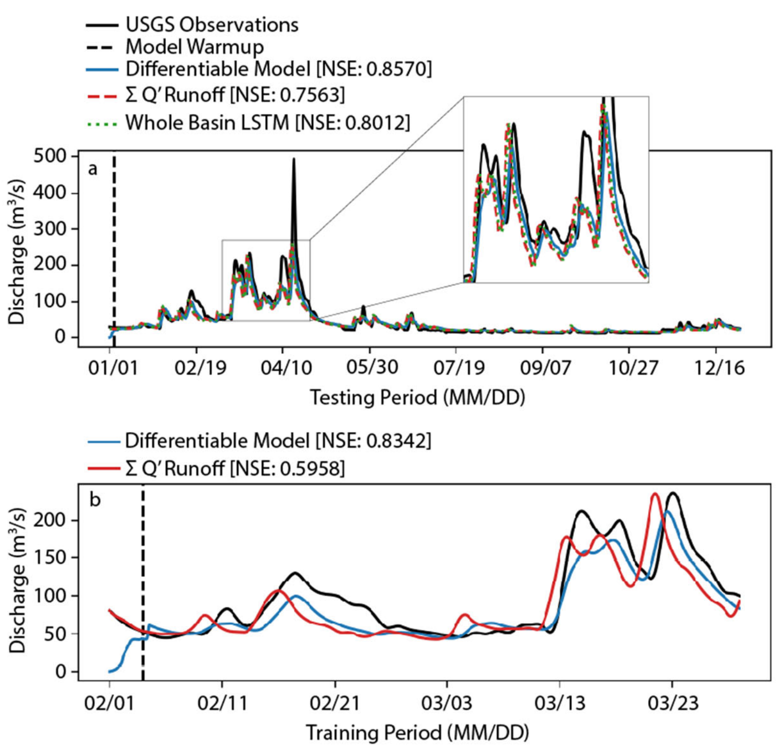

Figure 4: (a) Results from testing the trained model from Figure 4(b) over a year period (2001) compared with the summation of lateral inputs and Whole Basin LSTM benchmark (b) Results from training the differentiable model during an eight week period (2001) against USGS observations compared with the summation of lateral inputs.

The year-long test of the differentiable model yielded high metrics compared to the alternatives (Figure 4a). The differentiable model obtained a year-long NSE of 0.857, which is in line with the median NSE in the JRB. In contrast, the summation of $Q'(\tau = 9)$ and the whole-basin $LSTM\ (\tau = 0)$ were at 0.756 and 0.801, respectively. This comparison shows that if we simply added together the runoffs, the error due to timing could reduce NSE at the downstream gage by ~0.1 on a long-term basis. While the model had success especially with correctly timing the peak flows, it could not compensate for LSTM's errors, showing significant underestimation of the peak events. By design, the routing module should be detached from the errors in the runoff module.

Interestingly, without specific instructions, the scheme recovered a power-law-like relationship between Manning's *n* and drainage area, similar to the one assumed in the synthetic case (Figure 5). The *n* values were highest (near *n*=0.04) for smaller DA and declined gradually, approaching 0.015 at the lower end. The change rate of *n* as a function of DA then became more gentle as DA increased. This distribution agreed well with the general understanding that headwater streams running down ridges (this region is characterized by Ridge and Valley formations) have larger slopes, higher roughness, more vegetation, and thus higher *n*, while the high-order streams in the valley tend to have smaller slopes and smoother beds, corresponding with lower *n*. In most hydrologic handbooks (Mays, 2019), a smaller *n* is prescribed for larger rivers.

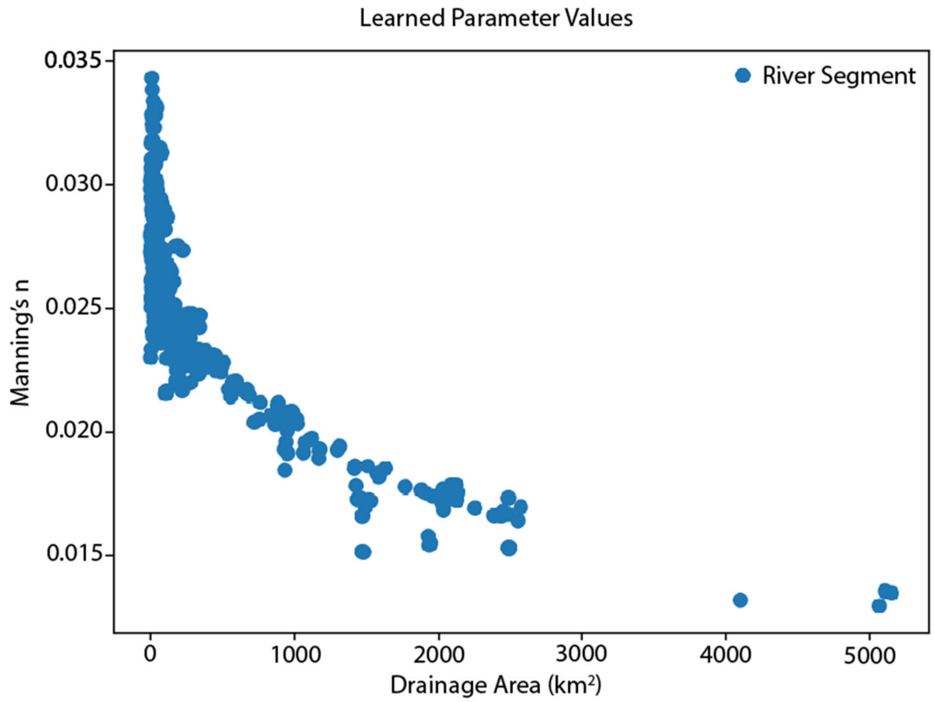

Figure 5: Learned relationship between Manning's n and drainage area for the Juniata River basin according to the trained neural network. The network was trained for the period of 2001/02/01-2001/03/29. Each dot represents a 2-km river reach.

3.3. Inner gage evaluation and effects of different training periods

Evaluating the model on the inner, untrained gages showed that the routing scheme became more competitive compared to alternatives as we looked further downstream (Table 2). For 2 of the 4 gages with larger than ~2000 km^2 of catchment area, the differentiable routing model performed noticeably better than homogeneous LSTM models for them (for the other two, they were about the same). For the three midsized subbasins (500~2000 km^2), the comparisons were mixed. For the small subbasins, and especially gage 01557500 (94.8 km^2), the uniform LSTM was noticeably better. The subbasin for 01557500 is smaller than our runoff-producing unit (HUC10s, with the smallest one ~200 km^2), suggesting predictions below this threshold can be error-prone. Our model was also consistently better than not doing routing (instead, summing and time-shifting the Q' runoff for each HUC10 produced by LSTM), or running routing with a uniform *n* of 0.02 (as would be selected for main channels from a lookup table) (Table 2), suggesting it learned useful parameterization skills.

Table 2: Internal gage NSE values for the year 2001, with the rows ranked by the size of the subbasin from small to large. The differentiable routing model was trained on the period from

2001/02/01-2001/03/29 calculating loss from the final gage but the LSTM was trained using >3000 CONUS gages. Bold font indicates the top performing model for each gage.

Edge ID	Gage Number	Basin Drainage Area (km^2)	LSTM NSE (τ = 0)	Q` Runoff NSE (τ = 9)	Differentiable routing model NSE (τ = 9)
1280	01557500	94.8	**0.8149**	0.5801	0.5849
1053	01560000	440.5	**0.7028**	0.6111	0.6627
2799	01558000	542.1	**0.8201**	0.7486	0.7758
4780	01556000	723.5	0.6624	0.6585	**0.6949**
2662	01562000	1943.5	0.7957	0.6969	**0.7997**
4801	01559000	2103.0	0.7815	0.7473	**0.8138**
2689	01563200	2482.9	0.5703	0.6556	**0.7869**
4809	01563500	5212.8	0.8024	0.7585	**0.8576**

This comparison informed us of the favorable and unfavorable ranges of applicability of our workflow. Our workflow found competitive advantages for stem rivers with catchments greater than 2,000 km^2, but may run into issues for scales smaller than the smallest runoff-producing unit (HUC10, around 200 km^2). The issues for the smallest basins may have been due to our procedure used to transfer mass between different grids (subbasin to regular grids on the river network). Smaller runoff-generating units could be used in the future to mitigate this issue. The advantages for larger basins were due to resolving both the routing process and the heterogeneity in rainfall and basin static attributes. The results imply that the advantages will increase for even larger basins, where currently LSTM does not apply, as well as basins where rainfall heterogeneity makes a big difference. The JRB is situated in the northeastern part of the CONUS; there could be many other regions where the effect of heterogeneity is more prominent. For example, past studies have always found it difficult to simulate large basins on the northern and central Great Plains (Feng et al., 2020; Martinez & Gupta, 2010), potentially due to spatially-concentrated rainfall and runoff generation (Fang & Shen, 2017). Also, in the mountainous areas of Northwest and Southeast, orographic precipitation could have significant spatial concentration. We hypothesize applying models to smaller basins and incorporating the routing scheme will allow these regions to be better modeled.

When the scheme was trained on eight-week periods from different years, it generated somewhat different but mostly functional parameterizations, unless it was trained in some unreasonable training periods where the LSTM doesn't match the observed outflows (Table 3). The maximum achievable NSEs for the years of 2001, 2005, 2007, and 2008 were 0.857, 0.87, 0.827 and 0.787, respectively. We found that if the models were trained on other periods (2001a 2001b, 2005b, 2007a), the test NSEs could still be close, and were at least not drastically worse. However, had we chosen 2007b, the results could have been worse (Figure 6a-d) Observing the characteristics of the different training periods, we see that the troublesome training periods did not contain full flood rise and recession phases (Figure 6e, 6f), and also had relatively low NSE. This could have led to ways to overfit. Hence, our experience suggests we need to pick periods that (i) contain full flood rise and recession phases; and (ii) have high NSEs for the training period.

Table 3. The NSE values correspond to testing differentiable models on different test years. Bold font indicates the highest NSE. Underlined metrics indicate poor performance.

Testing Period	Training Period							
	2001a 02/01-3/29	2001b 11/01-12/26	2005a 02/01-3/29	2005b 11/01-12/26	2007a 02/01-3/29	2007b 11/01-12/26	2008a 02/01-3/29	2008b 11/01-12/26
2001	**0.857**	0.845	0.850	0.853	0.857	<u>0.831</u>	0.782	0.856
2005	0.797	0.828	0.843	**0.870**	0.816	<u>0.713</u>	0.785	0.785
2007	0.815	0.812	0.821	**0.827**	0.819	<u>0.774</u>	0.753	0.813
2008	0.643	0.715	0.723	0.762	0.676	<u>0.534</u>	**0.787**	0.623

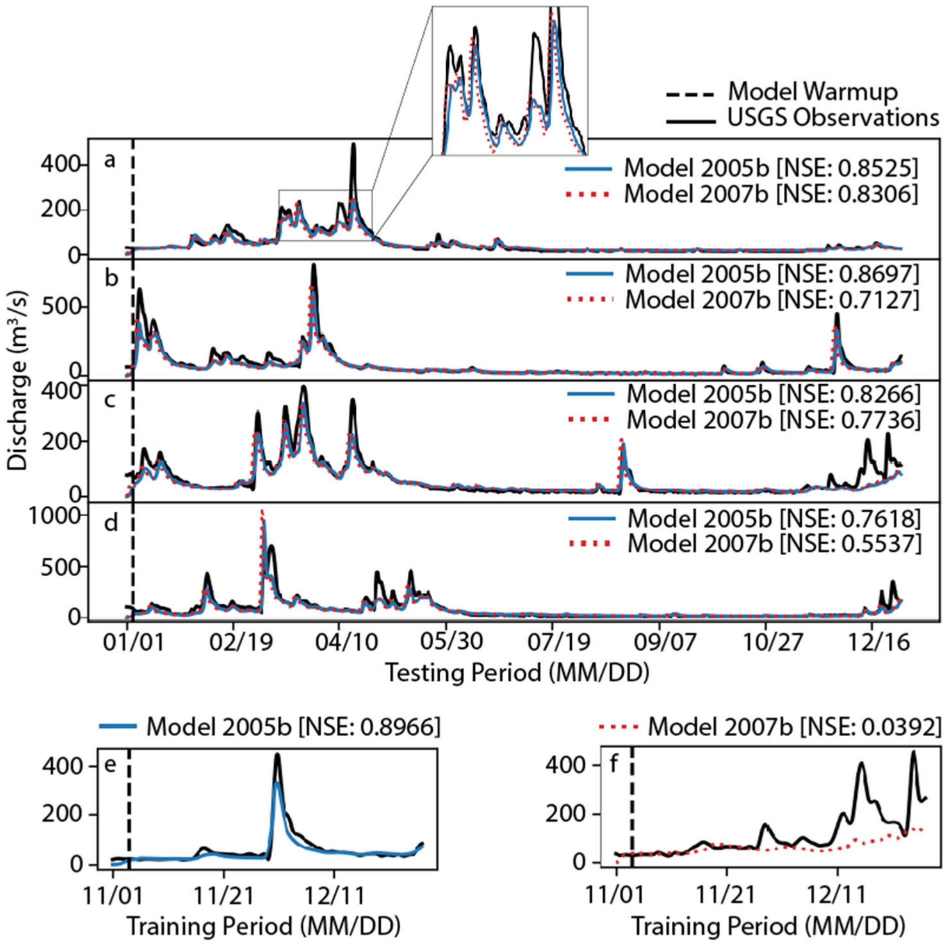

Figure 6: Training (e, f) and testing of two of the eight-week trained models during the years (a) 2001, (b) 2005, (c) 2007, and (d) 2008.

3.4. Further discussion

While the estimated *n* is functional for routing streamflow and physically-meaning, results suggest the downstream discharge only poses a moderate constraint on the *n* values, and by itself, may not be sufficient in identifying the true *n* values. Hence, we do not want to prematurely claim that the procedure retrieved highly realistic *n* parameterization in the real data case, especially considering that there are many input variables to NN covary in space and it may be difficult to disentangle their effects. Because we lacked the ground truth for *n* in the real-data case, we leave this evaluation for future effort as we compile more measurement data. Recall that we were able to retrieve the overall pattern of *n* in the synthetic experiments but there could be some parts in the parameter space with large uncertainties. This is because we have a high degree of freedom (a high-dimensional input space for the NN, influencing many reaches) constrained by only one downstream output with a relatively short training period. This training is nonetheless valuable because discharge data can be available widely. We will be able to employ it in conjunction with other constraints, e.g., scattered measurements or expert-specified relationships.

Here we employed a static parameterization scheme for *n*, but the framework would allow for a dynamic *n* to be employed (which would likely be dependent on Q). It is not clear if we must use a static parameterization, as some previous studies have found a dynamic *n* to offer better results (Ye et al., 2018). In the future, it will be interesting to see if a dynamical *n* parameterization could have a significant impact on the routing results.

Our approach, similar to a classical routing scheme, is modular --- the trained weights of the NN that generates *n* are not tied to a particular runoff model. Our work can be coupled to traditional models in multiple ways. Firstly, the trained network can be used to generate *n* for traditional models. In this way, no change is required on the part of the traditional models. Secondly, the neural network and the trained weights can be ported to other programming environments like Fortran and retraining is not necessary. This makes it possible to use the trained parameterizations as a built-in module in continental-scale models (Greuell et al., 2015; Johnson et al., 2019; Regan et al., 2018). An alternative approach is to lump both the routing and the runoff simulation into one problem and optimize them together, as done in some other studies (Jia et al., 2021). In our case, this would mean that we train both the runoff LSTM and the routing module together. In many big-data DL case studies, the lumped model could have higher performance compared to a workflow that separates the tasks into multiple minor tasks. However, in our case here, the available downstream gauge data is limited. Moreover, our approach is modular so it can be easily coupled to other runoff models, e.g., a non-differentiable, traditional model, or a differentiable one (Feng, Beck, et al., 2022; Feng, Liu, et al., 2022).

4. Conclusions

In this work, we used a combination of a pre-trained LSTM rainfall-runoff model and differentiable processed-based modeling via Muskingum-Cunge routing to create a learnable routing model (or, from the perspective of machine learning, a physics-informed graph neural network) to predict streamflow in stem rivers and learn river parameters throughout a river network. Our simple synthetic experiments succeeded in recovering the overall spatial pattern of Manning's *n* but could not recover the channel cross-sectional geometry parameter (*q*).

Furthermore, our synthetic experiments yielded good results recovering synthetic Manning's *n* and drainage area relationships, implying there is potential recoverability of some river parameters using our differential routing model.

Training the differentiable routing model on eight weeks of real-world data showed decent streamflow routing and improved upon approaches that did not use routing in their approaches. Similar results were shown when the differentiable model was tested on a full year of data. Despite the model's success, it could not compensate for errors in the LSTM causing an underestimation of significant storm events. When looking at Manning's *n* vs drainage area distribution attained by our trained model against USGS observations, we found that the *n* values agreed with the literature bounds for the area and also conforms to our knowledge of *n*. Further work can expand this analysis to other basins with different conditions (streams outside of the Ridge and Valley physiographic division) to see if the model can still identify their trends correctly. Reviewing the internal gage NSE scores over a full year of data showed a correlation between drainage area and the relative advantage of our routing scheme, highlighting the impacts of heterogeneity.

Our data suggests we need to pick periods that contain full flood rise and recession phases, and have high NSEs for the training period. We showed that systems trained on an eight-week period can be successfully applied to years outside of when they were trained and still attain high NSE scores. Our model's training size is limited to a small period of time due to memory constraints. In future work, we look to improve our graph infrastructure to allow for both cross-validation and an increased testing size.

Open Research

The LSTM code relevant to this work can be downloaded at http://doi.org/10.5281/zenodo.5015120. The differentiable routing model will be made available to reviewers upon a paper revision request, and a new Zenodo release will be published upon paper acceptance. The GAGES-II dataset can be downloaded at https://pubs.er.usgs.gov/publication/70046617. The NHDPlus data can be downloaded at https://nhdplus.com/NHDPlus/NHDPlusV2_home.php. The NLDAS forcing data can be downloaded at http://doi.org/10.5067/6J5LHHOHZHN4. Other data sources can be found in Table A1.

References

Adnan, R. M., Petroselli, A., Heddam, S., Santos, C. A. G., & Kisi, O. (2021). Comparison of different methodologies for rainfall–runoff modeling: machine learning vs conceptual approach. *Natural Hazards*, *105*(3), 2987–3011. https://doi.org/10.1007/s11069-020-04438-2

Arcement, G. J., & Schneider, V. R. (1989). *Guide for Selecting Manning's Roughness Coefficients for Natural Channels and Flood Plains* (Water-Supply Paper No. 2339). U.S. Geological Survey. Retrieved from https://pubs.usgs.gov/wsp/2339/report.pdf

Candela, A., Noto, L. V., & Aronica, G. (2005). Influence of surface roughness in hydrological response of semiarid catchments. *Journal of Hydrology*, *313*(3), 119–131. https://doi.org/10.1016/j.jhydrol.2005.01.023

Cunge, J. A. (1969). On the subject of a flood propagation computation method (Musklngum method). *Journal of Hydraulic Research*, *7*(2), 205–230. https://doi.org/10.1080/00221686909500264

Dottori, F., Szewczyk, W., Ciscar, J.-C., Zhao, F., Alfieri, L., Hirabayashi, Y., et al. (2018). Increased human and economic losses from river flooding with anthropogenic warming. *Nature Climate Change*, *8*(9), 781–786. https://doi.org/10.1038/s41558-018-0257-z

Douben, K.-J. (2006). Characteristics of river floods and flooding: a global overview, 1985–2003. *Irrigation and Drainage*, *55*(S1), S9–S21. https://doi.org/10.1002/ird.239

Duan, S., Ullrich, P., & Shu, L. (2020). Using convolutional neural networks for streamflow projection in California. *Frontiers in Water*, *2*. https://doi.org/10.3389/frwa.2020.00028

Falcone, J. A. (2011). *GAGES-II: Geospatial Attributes of Gages for Evaluating Streamflow* (Report). Reston, VA. https://doi.org/10.3133/70046617

Fan, W., Ma, Y., Li, Q., He, Y., Zhao, E., Tang, J., & Yin, D. (2019, November 22). Graph neural networks for social recommendation. arXiv. https://doi.org/10.48550/arXiv.1902.07243

Fang, K., & Shen, C. (2017). Full-flow-regime storage-streamflow correlation patterns provide insights into hydrologic functioning over the continental US. *Water Resources Research*, *53*(9), 8064–8083. https://doi.org/10.1002/2016WR020283

Fang, K., Shen, C., Kifer, D., & Yang, X. (2017). Prolongation of SMAP to spatiotemporally seamless coverage of continental U.S. using a deep learning neural network. *Geophysical Research Letters*, *44*(21), 11,030-11,039. https://doi.org/10/gcr7mq

Fang, K., Pan, M., & Shen, C. (2019). The value of SMAP for long-term soil moisture estimation with the help of deep learning. *IEEE Transactions on Geoscience and Remote Sensing*, *57*(4), 2221–2233. https://doi.org/10/gghp3v

Feng, D., Fang, K., & Shen, C. (2020). Enhancing streamflow forecast and extracting insights using long-short term memory networks with data integration at continental scales. *Water Resources Research*, *56*(9), e2019WR026793. https://doi.org/10.1029/2019WR026793

Feng, D., Beck, H., Lawson, K., & Shen, C. (2022). The suitability of differentiable, learnable hydrologic models for ungauged regions and climate change impact assessment. *Hydrology and Earth System Sciences Discussions*, 1–28. https://doi.org/10.5194/hess-2022-245

Feng, D., Liu, J., Lawson, K., & Shen, C. (2022, March 28). Differentiable, learnable, regionalized process-based models with physical outputs can approach state-of-the-art hydrologic prediction accuracy. *Water Resources Research (Accepted)*. https://doi.org/10.48550/arXiv.2203.14827

France-Presse, A. (2022, June 19). At least 59 dead and millions stranded as floods devastate India and Bangladesh. *The Guardian*. Retrieved from https://www.theguardian.com/world/2022/jun/18/at-least-18-dead-and-millions-stranded-as-floods-devastate-india-and-bangladesh

François, B., Schlef, K. E., Wi, S., & Brown, C. M. (2019). Design considerations for riverine floods in a changing climate – A review. *Journal of Hydrology*, *574*, 557–573. https://doi.org/10.1016/j.jhydrol.2019.04.068

Getirana, A. C. V., Boone, A., Yamazaki, D., Decharme, B., Papa, F., & Mognard, N. (2012). The Hydrological Modeling and Analysis Platform (HyMAP): Evaluation in the Amazon Basin. *Journal of Hydrometeorology*, *13*(6), 1641–1665. https://doi.org/10/f4jbcx

Ghanem, A., Steffler, P., Hicks, F., & Katopodis, C. (1996). Two-dimensional hydraulic simulation of physical habitat conditions in flowing streams. *Regulated Rivers: Research & Management*, *12*(2–3), 185–200. https://doi.org/10.1002/(SICI)1099-1646(199603)12:2/3<185::AID-RRR389>3.0.CO;2-4

Greuell, W., Andersson, J. C. M., Donnelly, C., Feyen, L., Gerten, D., Ludwig, F., et al. (2015). Evaluation of five hydrological models across Europe and their suitability for making projections under climate change. *Hydrology and Earth System Sciences Discussions*, *12*(10), 10289–10330. https://doi.org/10.5194/hessd-12-10289-2015

He, M., Wu, S., Huang, B., Kang, C., & Gui, F. (2022). Prediction of Total Nitrogen and Phosphorus in Surface Water by Deep Learning Methods Based on Multi-Scale Feature Extraction. *Water*, *14*(10), 1643. https://doi.org/10.3390/w14101643

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

HorizonSystems. (2016). *NHDPlus version 2*. Retrieved from http://www.horizon-systems.com/nhdplus/NHDplusV2_home.php

Hrnjica, B., Mehr, A. D., Jakupović, E., Crnkić, A., & Hasanagić, R. (2021). Application of Deep Learning Neural Networks for Nitrate Prediction in the Klokot River, Bosnia and Herzegovina. In *2021 7th International Conference on Control, Instrumentation and Automation (ICCIA)* (pp. 1–6). https://doi.org/10.1109/ICCIA52082.2021.9403565

IPCC. (2012). *Managing the Risks of Extreme Events and Disasters to Advance Climate Change Adaptation* (p. 582). Retrieved from https://www.ipcc.ch/report/managing-the-risks-of-extreme-events-and-disasters-to-advance-climate-change-adaptation/

Ji, X., Lesack, L., Melack, J. M., Wang, S., Riley, W. J., & Shen, C. (2019). Seasonal and inter-annual patterns and controls of hydrological fluxes in an Amazon floodplain lake with a surface-subsurface processes model. *Water Resources Research*, *55*(4), 3056–3075. https://doi.org/10/gghp4s

Jia, X., Zwart, J., Sadler, J., Appling, A., Oliver, S., Markstrom, S., et al. (2021). Physics-Guided Recurrent Graph Model for Predicting Flow and Temperature in River Networks. In *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)* (pp. 612–620). Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9781611976700.69

Johnson, J. M., Munasinghe, D., Eyelade, D., & Cohen, S. (2019). An integrated evaluation of the National Water Model (NWM)–Height Above Nearest Drainage (HAND) flood mapping methodology. *Natural Hazards and Earth System Sciences*, *19*(11), 2405–2420. https://doi.org/10.5194/nhess-19-2405-2019

Kalyanapu, A. J., Burian, S. J., & McPherson, T. N. (2009). Effect of land use-based surface roughness on hydrologic model output. *Journal of Spatial Hydrology*, *9*(2), 51–71. Retrieved from https://scholarsarchive.byu.edu/josh/vol9/iss2/2

Khorashadi Zadeh, F., Nossent, J., Sarrazin, F., Pianosi, F., van Griensven, A., Wagener, T., & Bauwens, W. (2017). Comparison of variance-based and moment-independent global sensitivity analysis approaches by application to the SWAT model. *Environmental Modelling & Software*, *91*, 210–222. https://doi.org/10.1016/j.envsoft.2017.02.001

Kingma, D. P., & Ba, J. (2017, January 29). Adam: A Method for Stochastic Optimization. arXiv. https://doi.org/10.48550/arXiv.1412.6980

Koks, E. E., & Thissen, M. (2016). A Multiregional Impact Assessment Model for disaster analysis. *Economic Systems Research*, *28*(4), 429–449. https://doi.org/10.1080/09535314.2016.1232701

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

Leclerc, M., Boudreault, A., Bechara, T. A., & Corfa, G. (1995). Two-dimensional hydrodynamic modeling: a neglected tool in the instream flow incremental methodology. *Transactions of the American Fisheries Society*, *124*(5), 645–662. https://doi.org/10.1577/1548-8659(1995)124<0645:TDHMAN>2.3.CO;2

Li, H., Wigmosta, M. S., Wu, H., Huang, M., Ke, Y., Coleman, A. M., & Leung, L. R. (2013). A physically based runoff routing model for land surface and earth system models. *Journal of Hydrometeorology*, *14*(3), 808–828. https://doi.org/10/ggj7ph

Li, H.-Y., Tan, Z., Ma, H., Zhu, Z., Abeshu, G. W., Zhu, S., et al. (2022). A new large-scale suspended sediment model and its application over the United States. *Hydrology and Earth System Sciences*, *26*(3), 665–688. https://doi.org/10.5194/hess-26-665-2022

Lin, G.-Y., Chen, H.-W., Chen, B.-J., & Yang, Y.-C. (2022). Characterization of temporal PM2.5, nitrate, and sulfate using deep learning techniques. *Atmospheric Pollution Research*, *13*(1), 101260. https://doi.org/10.1016/j.apr.2021.101260

Liu, J., Rahmani, F., Lawson, K., & Shen, C. (2022). A multiscale deep learning model for soil moisture integrating satellite and in situ data. *Geophysical Research Letters*, *49*(7), e2021GL096847. https://doi.org/10.1029/2021GL096847

Liu, L., Ao, T., Zhou, L., Takeuchi, K., Gusyev, M., Zhang, X., et al. (2022). Comprehensive evaluation of parameter importance and optimization based on the integrated sensitivity analysis system: A case study of the BTOP model in the upper Min River Basin, China. *Journal of Hydrology*, *610*, 127819. https://doi.org/10.1016/j.jhydrol.2022.127819

Martinez, G. F., & Gupta, H. V. (2010). Toward improved identification of hydrological models: A diagnostic evaluation of the "abcd" monthly water balance model for the conterminous United States. *Water Resources Research*, *46*(8). https://doi.org/10.1029/2009WR008294

Mays, L. W. (2010). *Water Resources Engineering* (2nd editio). Tempe, AZ: Wiley.

Mays, L. W. (2019). *Water Resources Engineering* (3nd editio). Tempe, AZ: Wiley. Retrieved from https://www.wiley.com/en-us/Water+Resources+Engineering%2C+3rd+Edition-p-9781119493167

Meyal, A. Y., Versteeg, R., Alper, E., Johnson, D., Rodzianko, A., Franklin, M., & Wainwright, H. (2020). Automated cloud based long short-term memory neural network based SWE prediction. *Frontiers in Water*, *2*. https://doi.org/10.3389/frwa.2020.574917

Mizukami, N., Clark, M. P., Sampson, K., Nijssen, B., Mao, Y., McMillan, H., et al. (2016). mizuRoute version 1: A river network routing tool for a continental domain water resources applications. *Geoscientific Model Development*, *9*(6), 2223–2238. https://doi.org/10.5194/gmd-9-2223-2016

Moore, R. B., & Dewald, T. G. (2016). The Road to NHDPlus — Advancements in Digital Stream Networks and Associated Catchments. *JAWRA Journal of the American Water Resources Association*, *52*(4), 890–900. https://doi.org/10.1111/1752-1688.12389

O, S., & Orth, R. (2021). Global soil moisture data derived through machine learning trained with in-situ measurements. *Scientific Data*, *8*(1), 170. https://doi.org/10.1038/s41597-021-00964-1

Ouyang, W., Lawson, K., Feng, D., Ye, L., Zhang, C., & Shen, C. (2021). Continental-scale streamflow modeling of basins with reservoirs: Towards a coherent deep-learning-based strategy. *Journal of Hydrology*, *599*, 126455. https://doi.org/10.1016/j.jhydrol.2021.126455

Papaioannou, G., Papadaki, C., & Dimitriou, E. (2020). Sensitivity of habitat hydraulic model outputs to DTM and computational mesh resolution. *Ecohydrology*, *13*(2), e2182. https://doi.org/10.1002/eco.2182

Prein, A. F., Rasmussen, R. M., Ikeda, K., Liu, C., Clark, M. P., & Holland, G. J. (2017). The future intensification of hourly precipitation extremes. *Nature Climate Change*, *7*(1), 48–52. https://doi.org/10.1038/nclimate3168

Rahmani, F., Shen, C., Oliver, S., Lawson, K., & Appling, A. (2021). Deep learning approaches for improving prediction of daily stream temperature in data-scarce, unmonitored, and dammed basins. *Hydrological Processes*, *35*(11), e14400. https://doi.org/10.1002/hyp.14400

Rahmani, F., Lawson, K., Ouyang, W., Appling, A., Oliver, S., & Shen, C. (2021). Exploring the exceptional performance of a deep learning stream temperature model and the value of streamflow data. *Environmental Research Letters*. https://doi.org/10.1088/1748-9326/abd501

Regan, R. S., Markstrom, S. L., Hay, L. E., Viger, R. J., Norton, P. A., Driscoll, J. M., & LaFontaine, J. H. (2018). *Description of the National Hydrologic Model for use with the Precipitation-Runoff Modeling System (PRMS)* (No. 6-B9). *Techniques and Methods*. U.S. Geological Survey. https://doi.org/10.3133/tm6B9

Rice, D. (2019, May 28). Mississippi River flood is longest-lasting in over 90 years, since "Great Flood" of 1927. *USA Today*. Retrieved from https://www.usatoday.com/story/news/nation/2019/05/28/mississippi-river-flooding-longest-lasting-since-great-flood-1927/1261049001/

Shen, C., & Lawson, K. (2021). Applications of Deep Learning in Hydrology. In *Deep Learning for the Earth Sciences* (pp. 283–297). John Wiley & Sons, Ltd. https://doi.org/10.1002/9781119646181.ch19

Shen, C., & Phanikumar, M. S. (2010). A process-based, distributed hydrologic model based on a large-scale method for surface–subsurface coupling. *Advances in Water Resources*, *33*(12), 1524–1541. https://doi.org/10/c4r8k5

Shen, C., Niu, J., & Phanikumar, M. S. (2013). Evaluating controls on coupled hydrologic and vegetation dynamics in a humid continental climate watershed using a subsurface - land surface processes model. *Water Resources Research*, *49*(5), 2552–2572. https://doi.org/10/f5gcrx

Shen, C., Niu, J., & Fang, K. (2014). Quantifying the effects of data integration algorithms on the outcomes of a subsurface–land surface processes model. *Environmental Modelling & Software*, *59*, 146–161. https://doi.org/10/ggj7mp

Shen, C., Riley, W. J., Smithgall, K. M., Melack, J. M., & Fang, K. (2016). The fan of influence of streams and channel feedbacks to simulated land surface water and carbon dynamics. *Water Resources Research*, *52*(2), 880–902. https://doi.org/10/f8gppj

Shen, C., Chen, X., & Laloy, E. (2021). Editorial: Broadening the use of machine learning in hydrology. *Frontiers in Water*, *3*. https://doi.org/10.3389/frwa.2021.681023

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

Sun, A. Y., Jiang, P., Yang, Z.-L., Xie, Y., & Chen, X. (2022). A graph neural network approach to basin-scale river network learning: The role of physics-based connectivity and data fusion. *Hydrology and Earth System Sciences Discussions*. https://doi.org/10.5194/hess-2022-111

Tsai, W.-P., Feng, D., Pan, M., Beck, H., Lawson, K., Yang, Y., et al. (2021). From calibration to parameter learning: Harnessing the scaling effects of big data in geoscientific modeling. *Nature Communications*, *12*(1), 5988. https://doi.org/10.1038/s41467-021-26107-z

USGS ScienceBase-Catalog. (2022). National Elevation Dataset (NED). Retrieved September 13, 2022, from https://www.sciencebase.gov/catalog/item/4fcf8fd4e4b0c7fe80e81504

Winsemius, H. C., Aerts, J. C. J. H., van Beek, L. P. H., Bierkens, M. F. P., Bouwman, A., Jongman, B., et al. (2016). Global drivers of future river flood risk. *Nature Climate Change*, *6*(4), 381–385. https://doi.org/10.1038/nclimate2893

Wunsch, A., Liesch, T., & Broda, S. (2022). Deep learning shows declining groundwater levels in Germany until 2100 due to climate change. *Nature Communications*, *13*(1), 1221. https://doi.org/10.1038/s41467-022-28770-2

Xia, Y., Mitchell, K., Ek, M., Sheffield, J., Cosgrove, B., Wood, E., et al. (2009). NLDAS Primary Forcing Data L4 Hourly 0.125 x 0.125 degree V002 (NLDAS_FORA0125_H) [Data set]. Goddard Earth Sciences Data and Information Services Center (GES DISC). https://doi.org/10.5067/6J5LHHOHZHN4

Xia, Y., Mitchell, K., Ek, M., Sheffield, J., Cosgrove, B., Wood, E., et al. (2012). Continental-scale water and energy flux analysis and validation for the North American Land Data Assimilation System project phase 2 (NLDAS-2): 1. Intercomparison and application of model products. *Journal of Geophysical Research: Atmospheres*, *117*(D3). https://doi.org/10.1029/2011JD016048

Xiang, Z., Yan, J., & Demir, I. (2020). A rainfall-runoff model with LSTM-based sequence-to-sequence learning. *Water Resources Research*, *56*(1), e2019WR025326. https://doi.org/10.1029/2019WR025326

Ye, A., Zhou, Z., You, J., Ma, F., & Duan, Q. (2018). Dynamic Manning's roughness coefficients for hydrological modelling in basins. *Hydrology Research*, *49*(5), 1379–1395. https://doi.org/10.2166/nh.2018.175

Zhi, W., Feng, D., Tsai, W.-P., Sterle, G., Harpold, A., Shen, C., & Li, L. (2021). From hydrometeorology to river water quality: Can a deep learning model predict dissolved oxygen at the continental scale? *Environmental Science & Technology*, *55*(4), 2357–2368. https://doi.org/10.1021/acs.est.0c06783

Zhu, F., Li, X., Qin, J., Yang, K., Cuo, L., Tang, W., & Shen, C. (2021). Integration of multisource data to estimate downward longwave radiation based on deep neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 1–15. https://doi.org/10.1109/TGRS.2021.3094321

Appendix

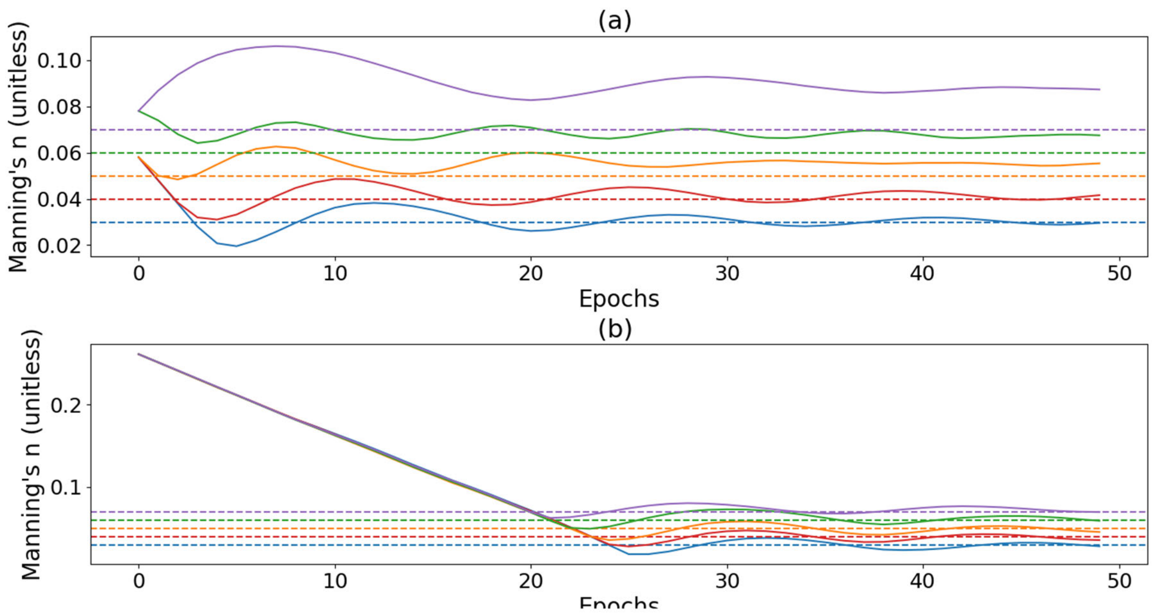

Figure A1: The synthetic parameter recovery of Manning's n after each epoch run with each colored line representing a different recovered value. (a) The initial value of n is set to 0.068 (b) the initial value of n is set to 0.271

Table A1: The attributes and forcings used to predict streamflow in the LSTM

Attribute	**Unit**	**Dataset**
Mean Elevation	m	SRTMGL1
Mean Slope	unitless	SRTMGL1
Basin Area	km^2	SRTMGL1
Dominant Land Cover	Class	MODIS
Dominant Land Cover Fraction	Percent	MODIS
Forest Fraction	Percent	MODIS
Root Depth (50)	m	MODIS
Soil Depth	m	MODIS
Ksat (0-5)	$\log_{10}$(cm/hr)	POLARIS
Ksat (5-15)	$\log_{10}$(cm/hr)	POLARIS

Theta s (0-5)	m^3/m^3	POLARIS
Theta s (5-15)	m^3/m^3	POLARIS
Theta r (5-15)	m^3/m^3	POLARIS
Ksat average (0-15)	$\log_{10}$(cm/hr)	POLARIS
Ksat e (0-5)	cm/hr	POLARIS
Ksat e (5-15)	cm/hr	POLARIS
Ksat average e (0-15)	cm/hr	POLARIS
Theta average s (0-15)	$e^{m3/m3}$	POLARIS
Theta average r (0-15)	$e^{m3/m3}$	POLARIS
Porosity	Percent	GLHYMPS
Permeability Permafrost	m^2	GLHYMPS
Permeability Permafrost (Raw)	m^2	GLHYMPS
Major Number of Dams	Unitless	GAGES-II
General Purpose of Dam	Unitless	National Inventory of Dams (NID)
Max of Normal Storage	Acre-ft	National Inventory of Dams (NID)
Standard Deviation of Normal Storage	Unitless	National Inventory of Dams (NID)
Number of dams within river (2009)	Unitless	GAGES-II
Normal Storage (2009)	Acre-ft	National Inventory of Dams (NID)
Precipitation hourly total	kg/m^2	NLDAS2
Surface downward longwave radiation	W/m^2	NLDAS2
Surface downward shortwave radiation	W/m^2	NLDAS2
Pressure	Pa	NLDAS2

Air Temperature	K	NLDAS2

SRTMGL1: https://doi.org/10.5067/MEaSUREs/SRTM/SRTMGL1.003
MODIS: https://modis.gsfc.nasa.gov/data/dataprod/mod12.php
POLARIS: https://doi.org/10.1029/2018WR022797
GLHYMPS: https://doi.org/10.5683/SP2/DLGXYO
NID: https://nid.usace.army.mil/
NLDAS2: https://ldas.gsfc.nasa.gov/nldas/v2/forcing

Table A2: The attributes used by the MLP to predict *n* and *q*

Attribute	**Unit**
Reach Width	m
Average-Reach Elevation	m
Slope	m/m
Reach Area	km^2
Total Drainage Area	km^2
Area Per Reach Length	km^2/km
Sinuosity	m/m
Bank Elevation	m