DiffRoute: Global Climatic Scale River Routing in less than a minute

Tristan Hascoet, Victor Pellet, Takemasa Miyoshi

RIKEN RCCS, RIKEN iThems, X-LMD

Overview

Contributions

We show that LTI RRMs are algebraically equivalent to DL 1D Convolution layers.

Leveraging this equivalence, we propose DiffRoute, an LTI RMM implementation with following features:

- **Generality**: The same implementation generalizes all LTI schemes (Muskingum, Linear wave, etc.). Any LTI scheme can be integrated with minimum code.
- **Differentiability**: Integration to Automatic Differentiation frameworks allow for efficient computations of gradients, enabling joint learning of different hydrological model components
- **Speed & Scalability**: Leveraging GPU parallel processing power, we achieve global climatic scale in 20s on a single GPU (85 years, 6M reaches). This was allowed by a combination of (1) Block-Sparse Computations (2) Fourier Analysis and (3) Tree Partitioning techniques

Limitations and Future Work

DiffRoute's expressivity is currently limited by two main assumptions:

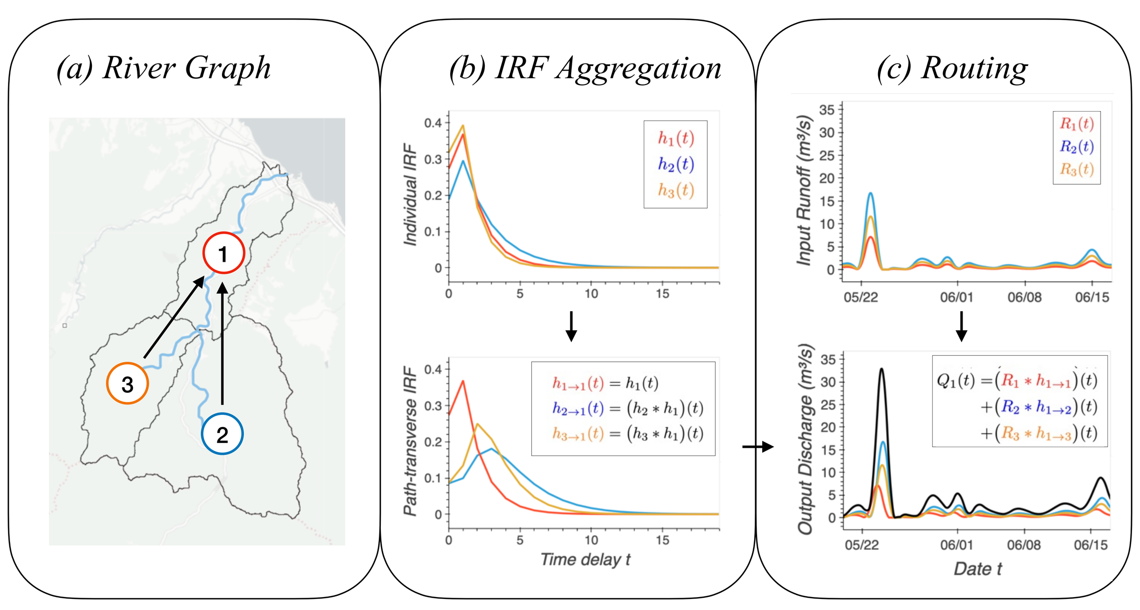
Linear Time Invariance: Computational optimizations leverage the LTI property of the system. This prevents us from expressing non-linear phenomena like dynamic flow velocity, backwater effects, flood-plain interactions, etc.

Tree-shaped River Networks: The current implementation does not allow to represent river network branching out downstream. Extension from tree structure to Directed Acyclic Graphs is possible.

In addition, while we have demonstrated end-to-end learning on several problem settings, the question of **identifiability** of hydrological parameters remains open, especially at global scale given the very sparse and noisy nature of global observations

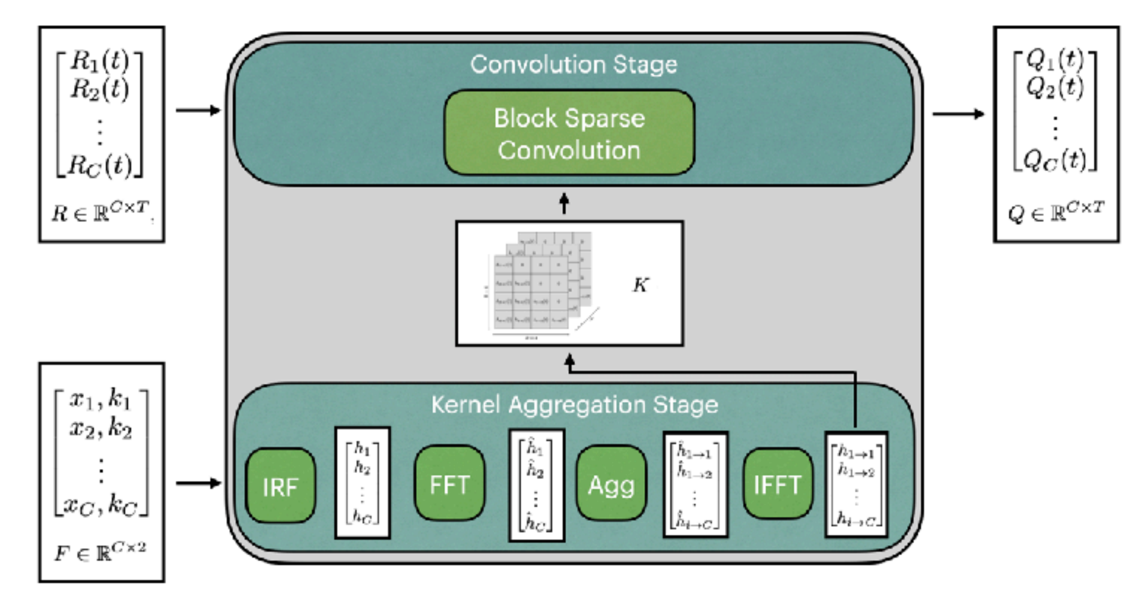
Methods

LTI RRM = Convolution Layers



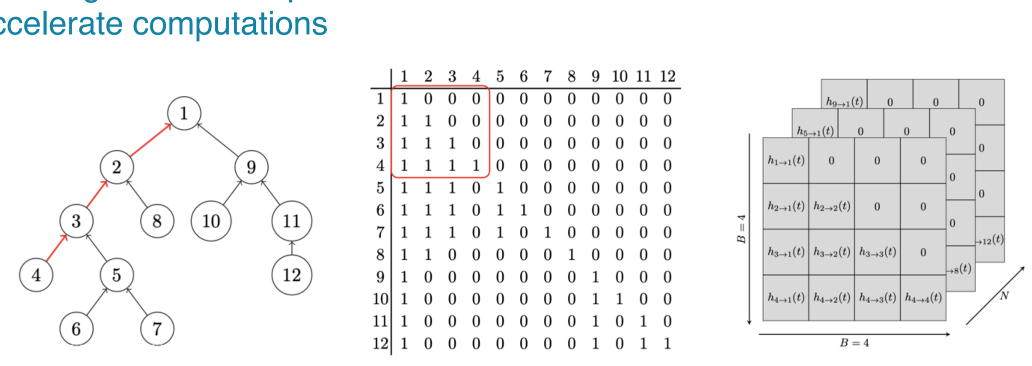
Model	Parameter F	IRF Formula h
Pure Lag	Δt: lag (time steps)	$h(t) = \max\{1 - \lvert t - \Delta t\rvert, 0\}$
Linear Reservoir	τ: residence time dt: model time step	$h(t) = x(1-x)^t$, with $x = \frac{1}{1+\frac{\tau}{dt}}$
Nash Cascade	τ: residence time n: number of reservoirs dt: model time step	$h(t) = \binom{t+n-1}{t} x^n (1-x)^t$ with $x = \frac{1}{1+\frac{\tau}{dt}}$.
Muskingum	x: weighting coefficient k: storage time constant dt: model time step	$h(t) = \begin{cases} C_0, & t=0, \\ (C_1 + C_2 C_0) C_2^{t-1}, & t \geq 1, \end{cases}$ with $C_0 = \frac{\frac{dt}{2} - kx}{D_e}$, $C_1 = \frac{\frac{dt}{2} + kx}{D_e}$, $C_2 = \frac{k(1-x) - \frac{dt}{2}}{D_e}$, and $D_e = k(1-x) + \frac{dt}{2}$.
Linear Diffusive Wave (Hayami)	L: channel length D: hydraulic diffusivity c: wave celerity	$h(t) = \frac{L}{2\sqrt{4\pi D t^3}} \exp\left(-\frac{(L-ct)^2}{4Dt}\right)$

Two-stages computation architecture



Leveraging Block Sparsity

We leverage the block-sparse structure of tree's transitive closure matrix to accelerate computations

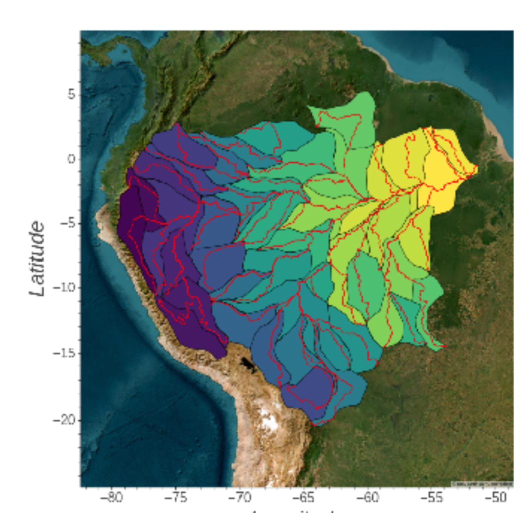


(a) Graph structure (b) Reachability matrix (c) Block-sparse kernel

Clustering

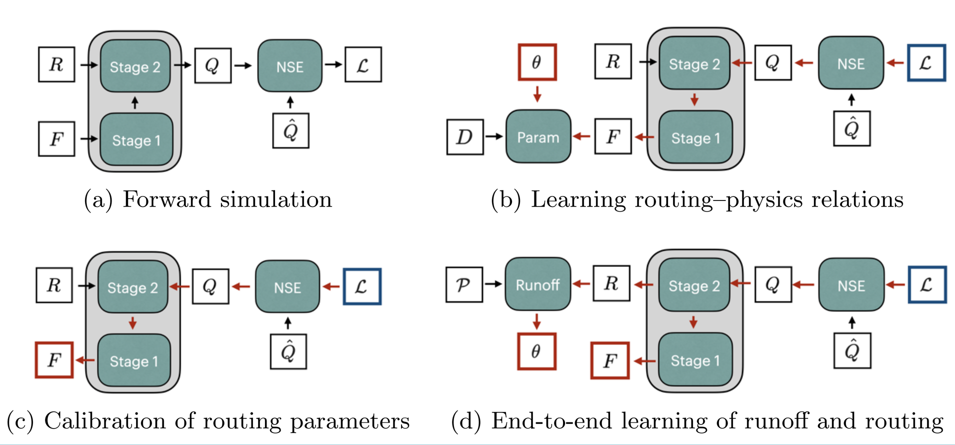
Computations become intractable in deep river systems.

We segment such deep river systems into clusters of manageable size



Use Cases

Automatic Differentiation allows for a variety of learning use-case



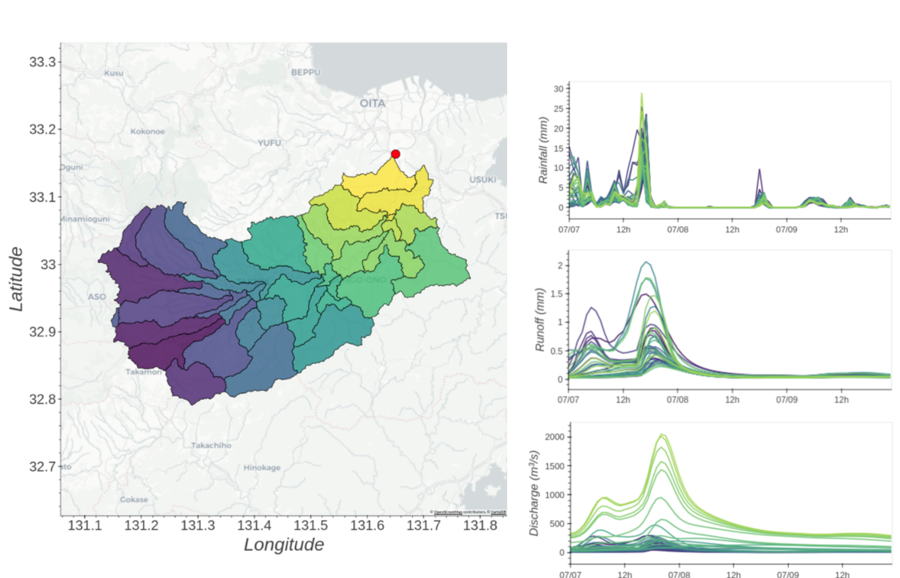
(a) Forward simulation (b) Learning routing-physics relations

(c) Calibration of routing parameters (d) End-to-end learning of runoff and routing

Results

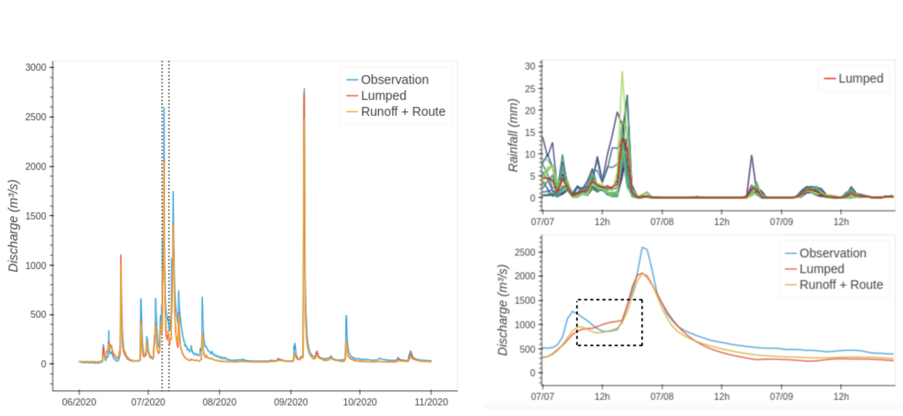
Basin Scale

Use-case D: End-to-end learning (runoff generation LSTM + DIffroute) from a single downstream gauge supervision



(a) Study basin and river network (37 catchments) (b) Model variables per catchment

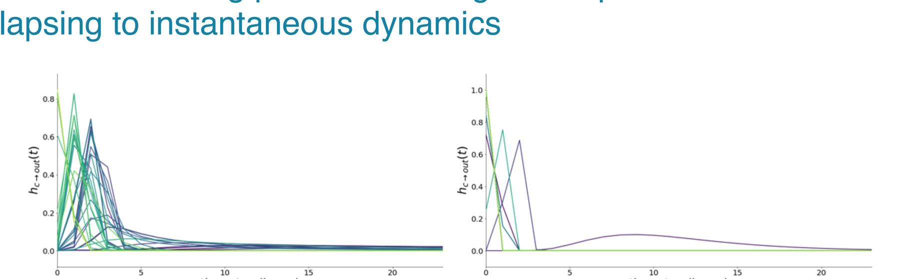
End-to-end learned dynamics outperform lumped LSTM baseline



Routing Scheme	Lumped	End-to-end	Two-steps
Pure Lag		0.905 ± 0.005	0.888 ± 0.004
Linear Reservoir		0.921 ± 0.003	0.893 ± 0.004
Muskingum	0.896 ± 0.002	0.910 ± 0.005	0.893 ± 0.004
Nash Cascade		0.919 ± 0.008	0.892 ± 0.004
Linear Diffusive Wave (Hayami)		0.923 ± 0.003	0.874 ± 0.005

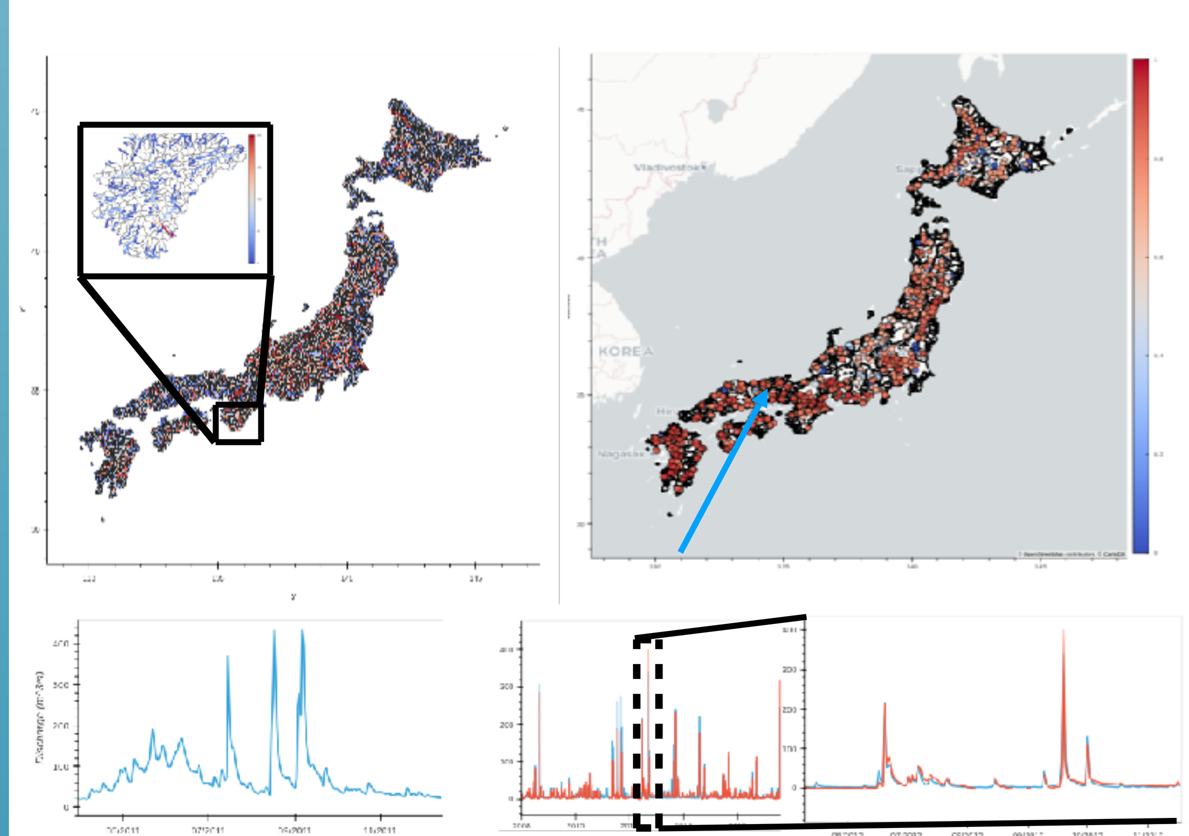
NSE Results for different approaches

End-to-end learning prevents routing model parameter estimation from collapsing to instantaneous dynamics



National Scale

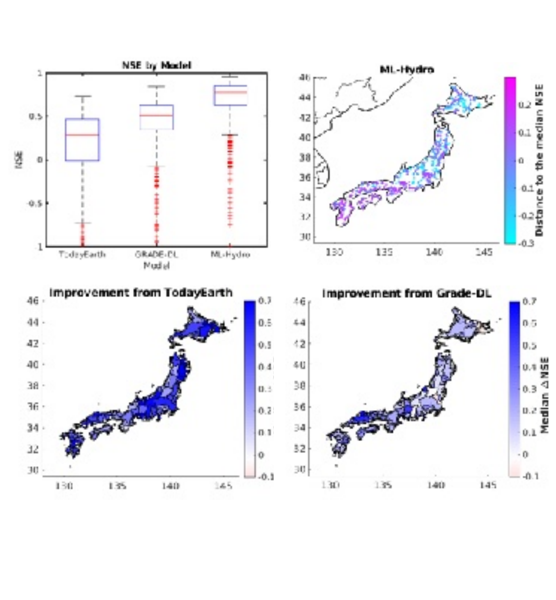
Use-case D: Applying the end-to-end learning methodology at national scale allows us to produce accurate and physically consistent spatially distributed river discharge predictions



We train and evaluate our system (Runoff Generation LSTM + DiffRoute) on 500 in-situ river discharge measurements across Japan. Generalization in space is investigated.

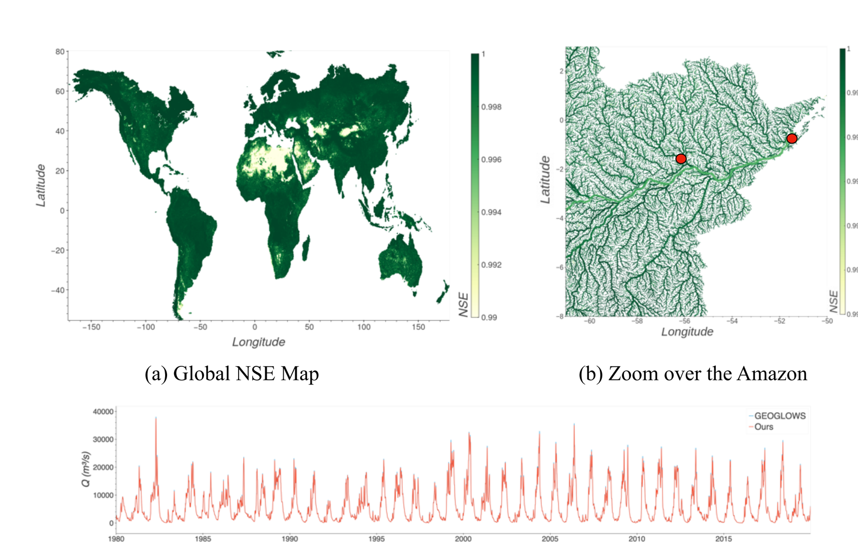
We compare our model results to **TodayEarth** and **GradesDL** river discharge outputs on our proposed dataset.

Models	NSE	KGE'				nRMSD
		α	γ	β	kge'	
TodayEarth	0.29	0.71	1.58	0.62	0.21	1.35
Grade-DL	0.51	0.78	0.81	0.91	0.57	1.07
MLHydro	**0.78**	**0.91**	**0.87**	**1.05**	**0.69**	**0.72**



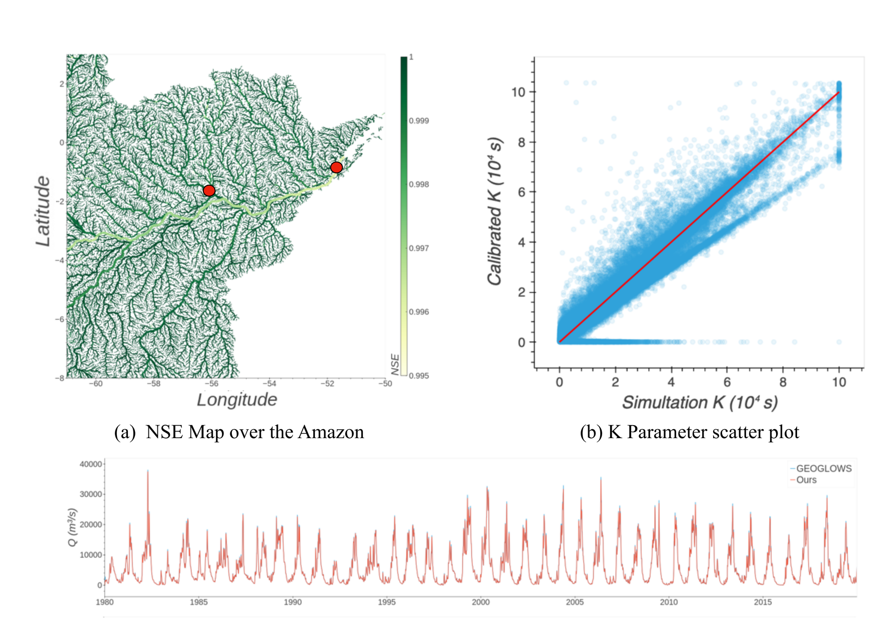
Global Scale

Use-case A: We demonstrate the scalability of DiffRoute by reproducing the GEOGlows V2 simulation, routing 85 years of ERA5 input runoff through 6M reaches in 20s on a single GPU chip.The output matches the original simulation with NSE of .9996.



(a) Global NSE Map (b) Zoom over the Amazon

Use-case B: AD allows for efficient automated calibration by gradient descent. We demonstrate efficient calibration over the Amazon basin.



(a) NSE Map over the Amazon (b) K Parameter scatter plot

Use-case C: Routing model parameters can also be inferred from physical properties of the river channels

We train a parameterized MLP ***f*** to regress Muskingum routing parameter ***k*** from channel distance ***D*** and upstream drainage area ***U***