Region Interaction Graph Neural Operator

A Graph-based framework for robust and accurate operator learning for PDEs on arbitrary domains

Sepehr Mousavi, *Shizheng Wen*, *Levi Lingsch*,
Maximilian Herde, *Bogdan Raonić*, *Siddhartha Mishra*

Computational Mechanics Group, ETH Zurich, Switzerland
Seminar for Applied Mathematics, ETH Zurich, Switzerland
ETH AI Center, Switzerland

ETH zürich

NEURAL INFORMATION PROCESSING SYSTEMS

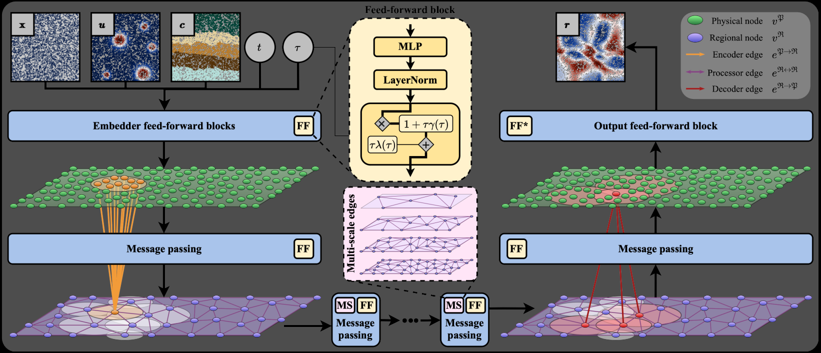
Problem: A system of m time-dependant PDEs in $\Omega_t = (0, T)$ and $\Omega_x \subset \mathbb{R}^d$:

$$\begin{aligned} \partial_t u &= \mathcal{F}(c, t, u, \nabla_x u, \nabla_x^2 u, \dots), && \forall (t, x) \in \Omega_t \times \Omega_x, \\ \mathcal{B}u(t, x) &= 0, && \forall (t, x) \in \Omega_t \times \partial\Omega_x, \\ u(0, x) &= a(x), && \forall x \in \Omega_x, \end{aligned}$$

Assumption: a solution operator $\mathcal{G}^\dagger : \mathcal{X} \times \mathcal{Q} \times \Omega_t \times \mathbb{R}^+ \to \mathcal{X}$ exists that maps the solution at any time $t \in \Omega_t$ to the solution at a later time $t + \tau \in \Omega_t$:

$$\mathcal{G}^\dagger(u^t, c^t, t, \tau) = u^{t+\tau}$$

Objective: Learn a parameterized operator $\mathcal{G}_\theta$ which approximates $\mathcal{G}^\dagger$ for $\tau < \tau_{\max}$



Space-continuous operator:

1. (possibly) independent input and output coordinates
2. invariant to the training discretization
3. (approximately) invariant to the training resolution
   with the help of *edge masking*

Random edge masking

How?

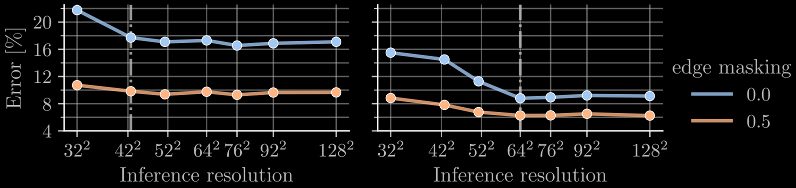
- A percentage of edges are randomly selected and *disabled*
- Different edges in the encoder, the decoder, and every processor step
- The masked edges change in every training iteration
- All edges are stochastically used during training

Why?

- Forces the model to discover weak correlations
- Training efficiency – fewer edges
- Ensemble predictions at inference

RIGNO is approximately resolution invariant.

possible to be trained with low resolutions



Edge masking

- helps in zero-shot sub-resolution inference;
- makes the training roughly 2x faster; and
- helps the optimization process, leading to improved accuracies.

RIGNO accurately learns the solution operators of PDEs on arbitrary domains.

Benchmarks on datasets with unstructured space discretizations.
Lowest (blue) and second lowest (orange) errors are highlighted.

Dataset	Median relative L^1 error [%]				
	RIGNO-18	RIGNO-12	GeoFNO	FNO DSE	GINO
Heat-L-Sines	**0.04**	**0.05**	0.15	0.53	0.19
Wave-C-Sines	**5.35**	6.25	13.1	**5.52**	5.82
NS-Gauss	**2.29**	**3.80**	41.1	38.4	13.1
NS-PwC	**1.58**	**2.03**	26.0	56.7	5.85
NS-SL	**1.28**	**1.91**	24.3	29.6	4.48
NS-SVS	**0.56**	**0.73**	9.75	26.0	1.19
CE-Gauss	**6.90**	**7.44**	42.1	30.8	25.1
CE-RP	**3.98**	**4.92**	18.4	27.7	12.3
ACE	**0.01**	**0.01**	1.09	1.29	3.33
Wave-Layer	**6.77**	**9.01**	11.1	28.3	19.2
AF	**1.00**	**1.09**	4.48	1.99	2.00
Elasticity	**4.31**	4.63	5.53	4.81	**4.38**

RIGNO's performance is also comparable with SOTA neural operators.

Benchmarks on datasets with uniform-grid space discretizations.
Lowest (blue) and second lowest (orange) errors are highlighted.

Dataset	Median relative L^1 error [%]				
	RIGNO-18	RIGNO-12	CNO	scOT	FNO
NS-Gauss	2.74	3.78	10.9	2.92	14.2
NS-PwC	1.12	1.82	5.03	7.11	11.2
NS-SL	1.13	1.82	2.12	2.49	2.08
NS-SVS	0.56	0.75	0.70	0.99	6.21
CE-Gauss	5.47	7.56	22.0	9.44	28.7
CE-RP	3.49	4.43	18.4	9.74	31.2
ACE	0.01	0.01	0.28	0.21	0.60
Wave-Layer	6.75	8.97	8.28	13.4	28.0
Poisson-Gauss	1.80	2.44	2.04	0.68	11.5

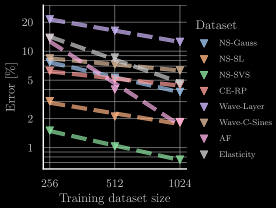
RIGNO's accuracy scales with dataset size.

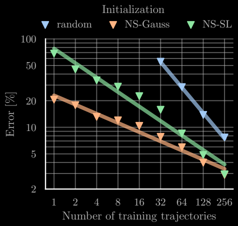
RIGNO's accuracy scales with model size.

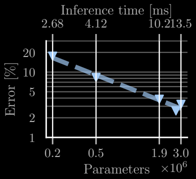
RIGNO can benefit hugely from pre-training.

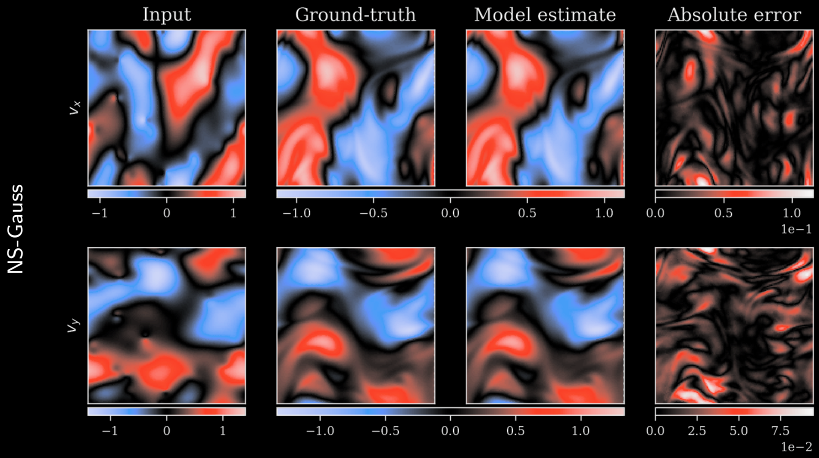
RIGNO is robust to noisy inputs.

RIGNO can provide indicators of model uncertainties.

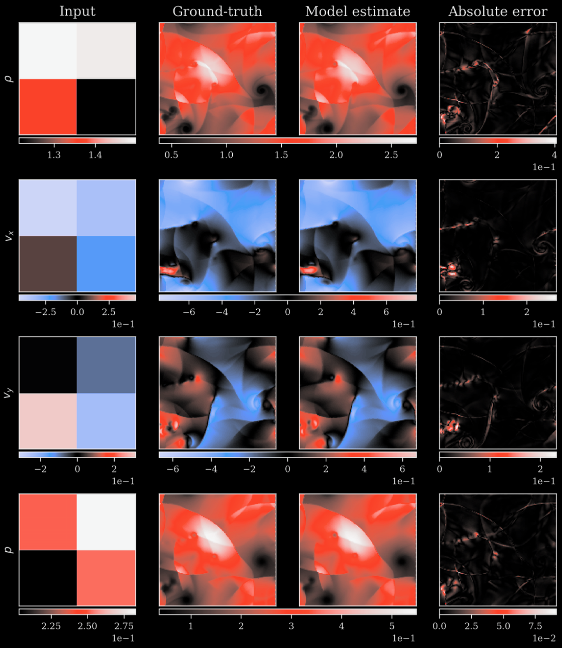







Input
Ground-truth
Model estimate
Absolute error
NS-Gauss
v_x
v_y
−1 0 1
−1.0 −0.5 0.0 0.5 1.0
0.0 0.5 1.0
1e−1
0.0 2.5 5.0 7.5
1e−2

CE-RP

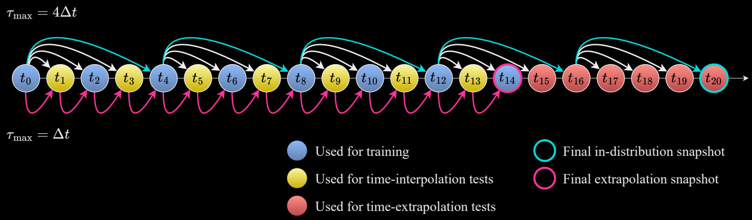

Input
Ground-truth
Model estimate
Absolute error
ρ
1.3
1.4
0.5
1.0
1.5
2.0
2.5
0
2
4
1e−1
v_x
−2.5
0.0
2.5
1e−1
−6
−4
−2
0
2
4
6
1e−1
0
1
2
1e−1
v_y
−2
0
2
1e−1
−6
−4
−2
0
2
4
6
1e−1
0
1
2
1e−1
p
2.25
2.50
2.75
1e−1
1
2
3
4
5
1e−1
0.0
2.5
5.0
7.5
1e−2

Recall that

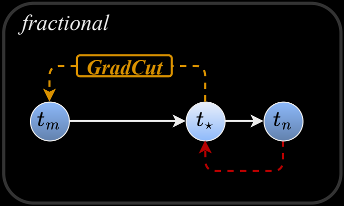
$$u^{t+\tau} \simeq \mathcal{G}_\theta(u^t, c^t, t, \tau)$$

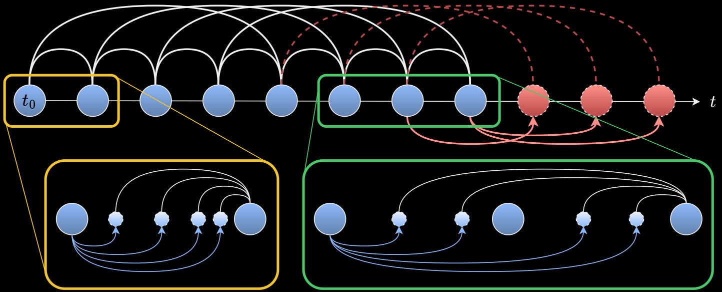
Autoregressive inference:

$$u^0 \xrightarrow[\mathcal{G}_\theta]{} u^{\tau_1} \xrightarrow[\mathcal{G}_\theta]{} u^{\tau_1+\tau_2} \xrightarrow[\mathcal{G}_\theta]{} \cdots \xrightarrow[\mathcal{G}_\theta]{} u^{\sum_{i=1}^{n-1}\tau_i} \xrightarrow[\mathcal{G}_\theta]{} u^t$$



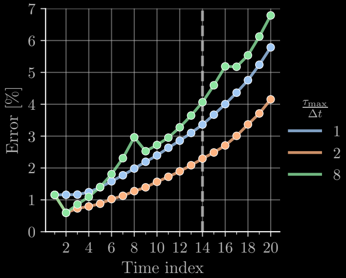
Variable	Dataset distribution	Interpolation	Extrapolation
t:	$\{t_0, t_2, t_4, \ldots, t_{12}\}$	$\{t_1, t_3, \ldots\}$	$\{t_{15}, t_{16}, \ldots\}$
u:	$\{u^{t_0}, u^{t_2}, \ldots, u^{t_{12}}\}$	$\{u^{t_1}, u^{t_3}, \ldots\}$	$\{u^{t_{15}}, u^{t_{16}}, \ldots\}$
τ:	$\{2\Delta t, 4\Delta t, \ldots, 14\Delta t\}$	$\{3\Delta t, 5\Delta t, \ldots\}$	$\{\Delta t, 15\Delta t, 16\Delta t, \ldots\}$
(output) r:	$\{u^{t_2}, u^{t_4}, \ldots, u^{t_{14}}\}$	$\{u^{t_3}, u^{t_5}, \ldots\}$	$\{u^{t_1}, u^{t_{15}}, u^{t_{16}}, \ldots\}$

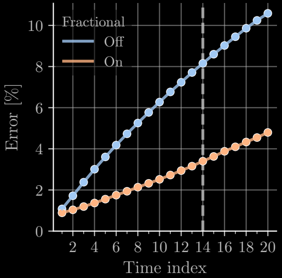

fractional
GradCut
t_m
$t_\star$
t_n


t_0
t

Fractional pairing allows for small time differences (τ)

not restricted to RIGNO





- Near-perfect interpolation in both τ and t
- Reasonable extrapolation in t after t_{14}
- Reasonable extrapolation in τ — time-continuous operator
- Rate of error accumulation with $\tau_{max} = \Delta t$ comparable to $\tau_{max} = 2\Delta t$
  First snapshot (u^{t_1}) is still out-of-distribution

Read the full article on

Check the codes on