Shifting Time: Time-series Forecasting with Khatri-Rao Neural Operators

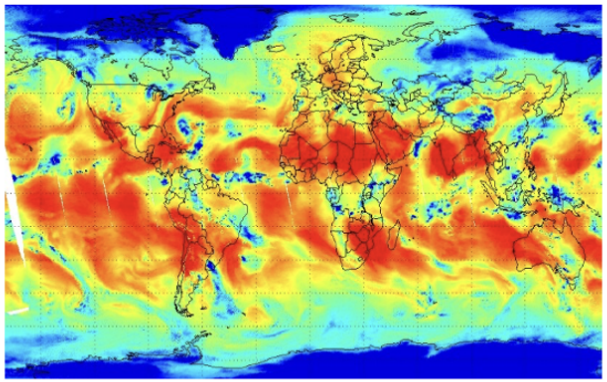
Srinath Dama, Kevin Course, Prasanth B. Nair

Institute for Aerospace Studies
University of Toronto

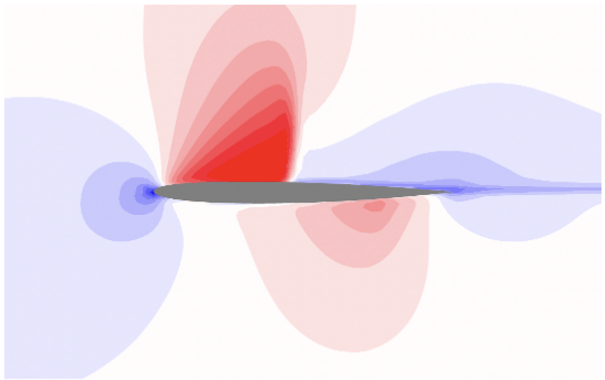
June 26, 2025

Time-series forecasting

Finance



Weather



Flow around an airfoil

- Autoregressive models
  - Examples: ARIMA, GRU, N-BEATS, etc.
  - Require regularly sampled time-series data
  - Cannot handle irregularly sampled data
- Continuous-time models
  - Examples: Neural ODEs, Neural SDEs, etc.
  - Can learn from irregularly sampled time-series data
  - Difficult to scale and train for large temporal or spatio-temporal datasets

Time-shift operator

$$\mathcal{A}_{t_p}^{t,t_f} : L^2(\underbrace{[t_p, t]}_{\text{past}}; \mathbb{R}^n) \longrightarrow L^2(\underbrace{(t, t_f]}_{\text{future}}; \mathbb{R}^n)$$

- Existence of $\mathcal{A}_{t_p}^{t,t_f}$ follows from Picard-Lindelöf theorem

Properties:

- Semigroup property: $\mathcal{A}_{t_p}^{t_2,t_f} = \mathcal{A}_{t_1}^{t_2,t_f} \circ \mathcal{A}_{t_p}^{t_1,t_2}$, where $t_p < t_1 < t_2 < t_f$,
- Continuity property: $\exists C > 0$ such that

  $||A_{t_p}^{t,t_f} z_1 - A_{t_p}^{t,t_f} z_2||_{L^2((t,t_f];\mathbb{R}^n)} \leq C||z_1 - z_2||_{L^2([t_p,t];\mathbb{R}^n)}, \ \forall z_1, z_2 \in L^2([t_p, t]; \mathbb{R}^n)$

Advantages:

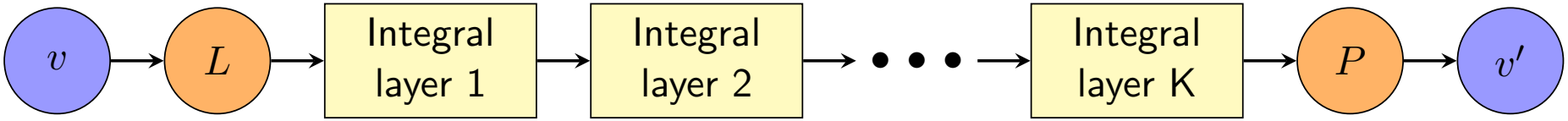
- Learn directly from irregularly sampled observations
- Forecasting at super-resolution in both space and time

Spatio-temporal time-shift operator

$$\mathcal{A}_{t_p}^{t,t_f} : \mathcal{U}(\Omega \times \underbrace{[t_p, t]}_{\text{past}}; \mathbb{R}^n) \longrightarrow \mathcal{U}(\Omega \times \underbrace{(t, t_f]}_{\text{future}}; \mathbb{R}^n)$$

We parametrize the time-shift operator $\mathcal{A}_{t_p}^{t,t_f}$ using Khatri-Rao neural operator

Khatri-Rao neural operators (KRNOs)



- An integral transform layer maps the input spatio-temporal vector field $v_\ell : \Omega \times [0,\tau] \to \mathbb{R}^p$ to $v_{\ell+1} : \Omega \times [0,\tau] \to \mathbb{R}^q$, is given by

$$v_{\ell+1}(t,x) = \mathcal{K}(v_\ell)(t,x) = \int_\Omega \int_0^\tau \overbrace{\kappa(\{t,x\},\{t',x'\})}^{\text{non-stationary kernel}} v_\ell(t',x')dt'dx' + W v_\ell(t,x) + b,$$

- Evaluating the integral transform layer scales as $\mathcal{O}(N^2)$ which is prohibitively expensive

Khatri-Rao product structure

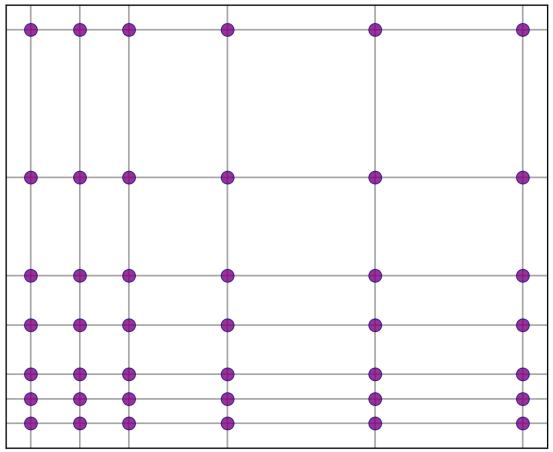


Figure: Rectilinear grid.

Theorem

If the quadrature nodes lie on a product grid, $X = \bar{t} \times x^{(1)} \times \ldots x^{(d-1)}$*, where* $\bar{t} \in \mathbb{R}^n$ *and* $x^{(i)} \in \mathbb{R}^n$*,* $\kappa(X, X)$ *inherits the Khatri-Rao product structure,*

$$\kappa(X, X) = \kappa^{(1)}(\bar{t}, \bar{t}) * \left(\underset{i=2}{\overset{d}{*}} \kappa^{(i)}(x^{(i-1)}, x^{(i-1)}) \right),$$

where $\kappa^{(i)}(\cdot, \cdot) \in \mathbb{R}^{qn \times pn}$ *is a block-partitioned matrix where block* jk *is the* jk^{th} *output from the component kernel* $\kappa^{(i)}$ *evaluated on the outer product of the quadrature nodes along the* i^{th} *dimension.*

Computational advantages of KRNOs

- Computational complexity reduces from $\mathcal{O}(N^2)$ to $\mathcal{O}(N^{1+1/d})$ with $\mathcal{O}(N^{2/d} + N)$ memory
- Exact evaluation of integral transforms without approximating the kernel function

Method	Time	Non-stationary	Exact kernel
GNO	$\mathcal{O}(NN')$	✓	✗
MGNO	$\mathcal{O}(N)$	✓	✗
FNO	$\mathcal{O}(N \log N)$	✗	✓
KRNO	$\mathcal{O}(N^{1+1/d})$	✓	✓

Numerical Studies

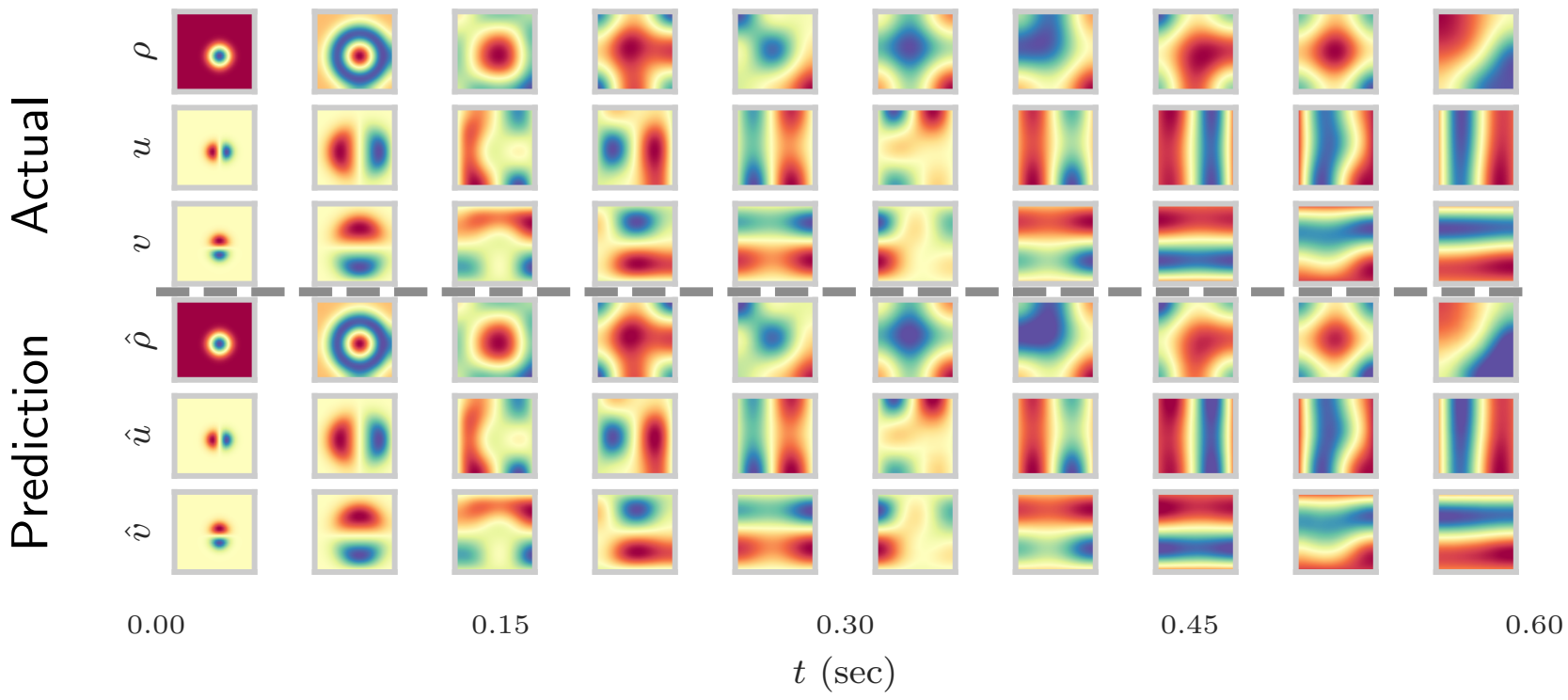
- Time-series datasets:
  - 5 irregularly sampled: MuJoCo, MIMIC, USHCN, Human Activity, etc
  - 16 regularly sampled: Darts, M4, Crypto, etc.
- Spatio-temporal datasets:
  - Shallow Water problem and Climate modeling problem
- In total, we considered 39 test cases or error metrics
- KRNOs achieves state-of-the-art or highly competitive performance on several benchmarks

MuJoCo Physics simulation dataset

Table: Forecasting performance on MuJoCo dataset with different levels of missing observations

Methods	Test MSE			
	Regular	30% dropped	50% dropped	70% dropped
GRU-Δt	0.223 ± 0.020	0.198 ± 0.036	0.193 ± 0.015	0.196 ± 0.028
ODE-RNN	0.328 ± 0.225	0.274 ± 0.213	0.237 ± 0.110	0.267 ± 0.217
Latent-ODE	0.029 ± 0.011	0.056 ± 0.001	0.055 ± 0.004	0.058 ± 0.003
ACE-NODE	0.039 ± 0.003	0.053 ± 0.007	0.053 ± 0.005	0.052 ± 0.006
NCDE	0.028 ± 0.002	0.027 ± 0.000	0.027 ± 0.001	0.026 ± 0.001
LEAP	0.022 ± 0.002	0.022 ± 0.001	0.022 ± 0.002	0.022 ± 0.001
Neural SDE	0.028 ± 0.004	0.029 ± 0.001	0.029 ± 0.001	0.027 ± 0.000
Neural LSDE	$\underline{0.013 \pm 0.000}$	0.014 ± 0.001	0.014 ± 0.000	$\underline{0.013 \pm 0.001}$
KRNO	$\mathbf{0.007 \pm 0.002}$	$\mathbf{0.008 \pm 0.002}$	$\mathbf{0.0114 \pm 0.004}$	$\mathbf{0.0115 \pm 0.002}$

Shallow water problem



Shallow water problem

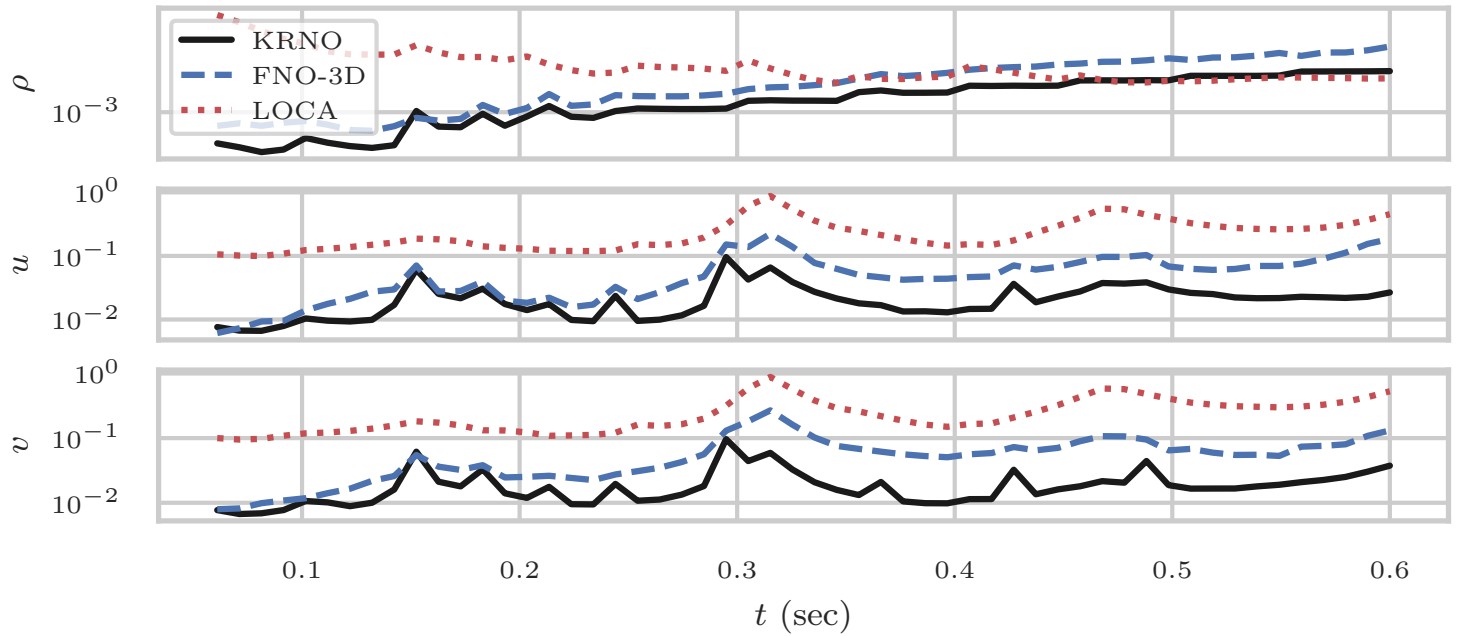


Table: Comparison of KRNO with FNO-3D and LOCA

Method	#Parameters	L^2 relative error		
		ρ	u	v
FNO-3D	$2,462,895$	0.00211	0.02606	0.02637
LOCA	$94,477,220$	0.00314	0.15221	0.14999
KRNO	$\mathbf{146,159}$	$\mathbf{0.00145}$	$\mathbf{0.01497}$	$\mathbf{0.01459}$

Shifting Time: Time-series Forecasting with Khatri-Rao Neural Operators

Srinath Dama, Kevin Course, Prasanth B. Nair

Institute for Aerospace Studies
University of Toronto

June 26, 2025