

Property-preserving operator learning for incompressible flows

Ramansh Sharma[1]

[1] Kahlert School of Computing, University of Utah
[2] Department of Computing and Mathematical Sciences, California Institute of Technology







Matthew Lowery[1] Houman Owhadi[2] Varun Shankar[1]

2026 SIAM Conference on Mathematics of Data Science (MDS26)

November 16, 2026

Motivation: incompressible flow

- Design and prediction require many flow solves.
- High fidelity simulations are expensive.
- A fast surrogate should still respect incompressibility.

Incompressible Navier-Stokes equations

$$
\begin{aligned}
\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} &= \nu\nabla^2\mathbf{u} - \frac{1}{\rho}\nabla p + \frac{1}{\rho}\mathbf{f}, && \text{in } \Omega\times(0,T],\\
\nabla\cdot\mathbf{u} &= 0, && \text{in } \Omega\times(0,T],\\
\mathbf{u}(\mathbf{x},0) &= \mathbf{u}_0(\mathbf{x}), && \text{in } \Omega,\\
\mathcal{B}\mathbf{u} &= \mathbf{g}, && \text{on } \partial\Omega\times(0,T].
\end{aligned}
$$



NASA Ames Research Center

THE UNIVERSITY of UTAH

Surrogate modeling with operator learning



$$\mathcal{G} : a \longmapsto v$$

- Operator surrogates learn maps between function spaces.

Li et al., "Fourier neural operator for parametric partial differential equations" (2020).
Lu et al., "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators" (2021).

Surrogate modeling with operator learning



$$\mathcal{G} : a \longmapsto v$$

- Operator surrogates learn maps between function spaces.

Example operator: initial velocity a(x) = v(x, 0) ⟶ final velocity v(y, T).

Li et al., "Fourier neural operator for parametric partial differential equations" (2020).
Lu et al., "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators" (2021).

Surrogate modeling with operator learning



$$\widetilde{\mathcal{G}} = \chi \circ \widetilde{f} \circ \varphi$$

χ : velocity reconstruction $\quad$ $\widetilde{f}$: coefficient prediction $\quad$ φ : input sampling

- Operator surrogates learn maps between function spaces.
- We use kernel methods for the input-to-output map.
- The output representation will encode desirable physical properties.

Batlle, Darcy, Hosseini & Owhadi, "Kernel methods are competitive for operator learning" (2024).

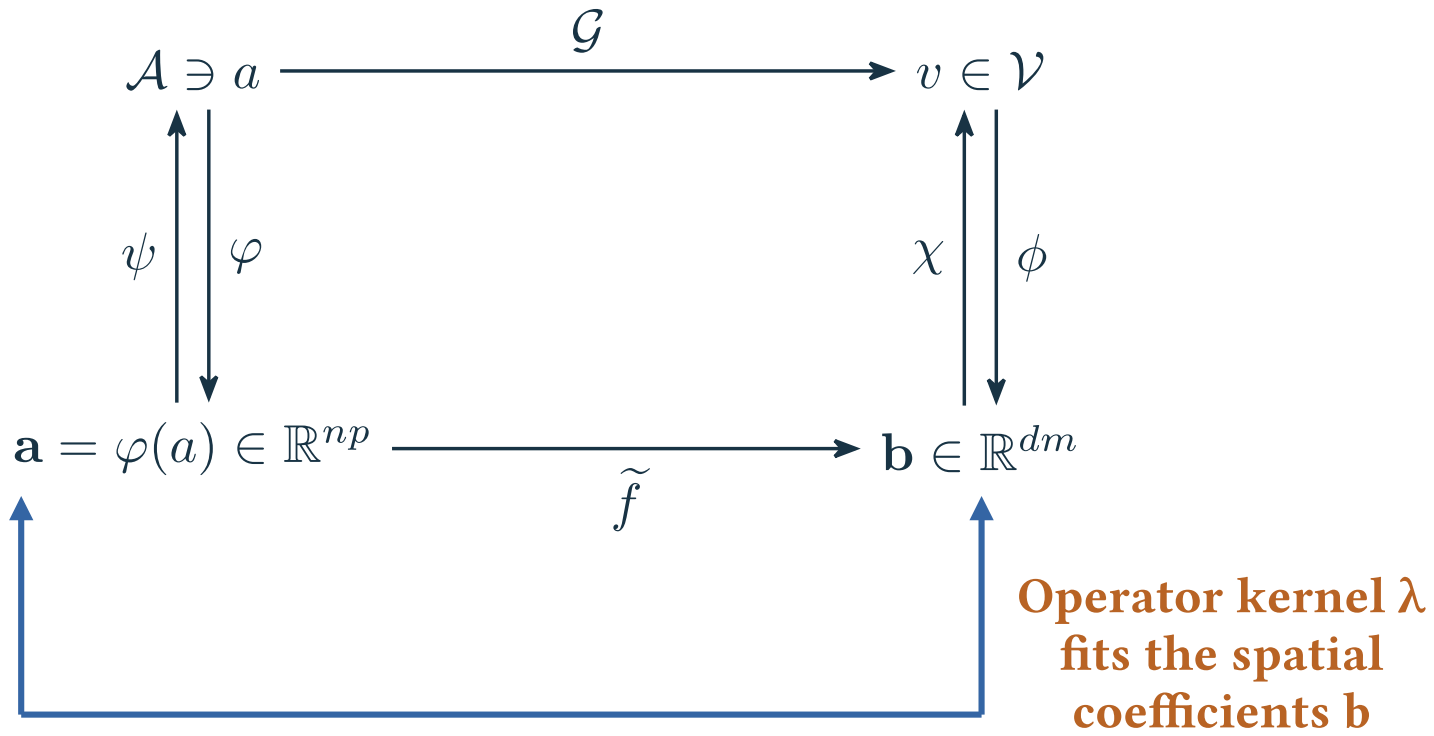
A property-preserving kernel basis

$$\widetilde{\mathbf{v}}(y) = \sum_{j=1}^{m} \Phi(y, y_j)\mathbf{b}_j$$

- Each basis column is divergence free, so any construction preserves incompressibility *analytically*.
- Periodic kernels enforce periodicity when the problem requires it.
- Multiscale kernels can encode prescribed turbulence-informed power laws.

THE UNIVERSITY of UTAH

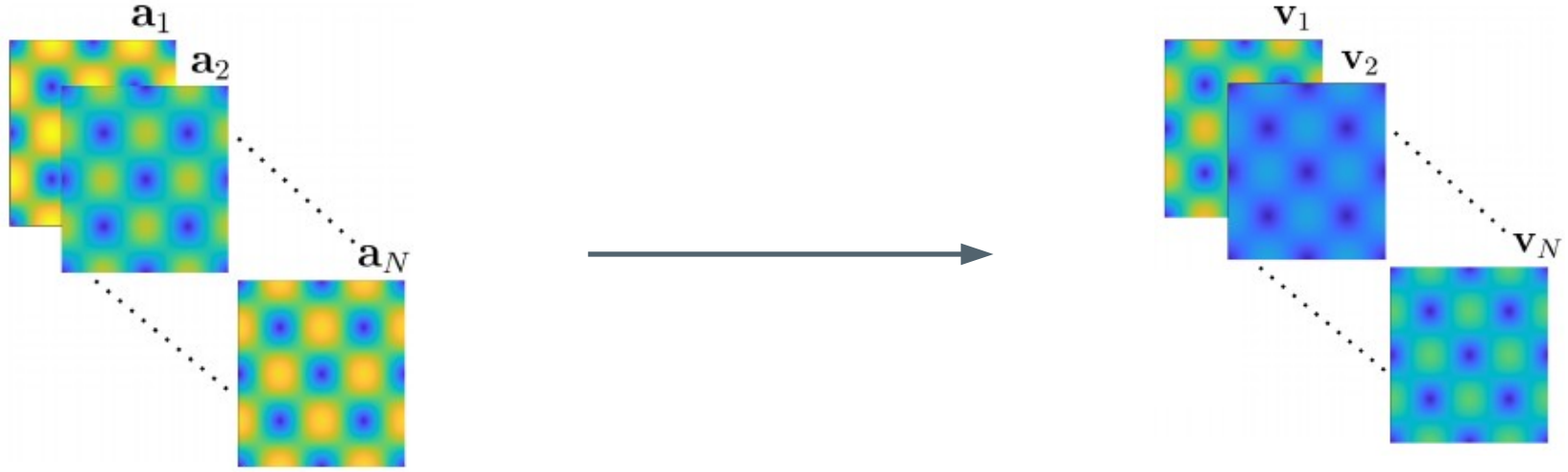
Fit the coefficients



- Fit one input-to-coefficient map for all output spatial coefficients.
- Reconstruct at any query point using the property-preserving kernel basis.

Property-preserving kernel method
Training and prediction

Training pairs → Spatial coefficients → Operator regression → Unseen input function



- Start with N pairs of input and output functions.

Property-preserving kernel method
Training and prediction

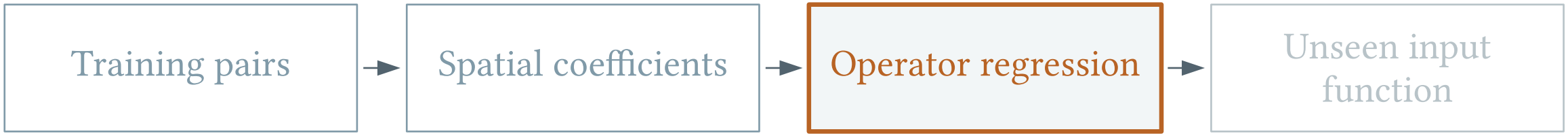


$$\mathbf{\Phi} \mathbf{b}_i = \mathbf{v}_i \quad (i = 1, \ldots, N)$$

encodes incompressibility, periodicity,
and turbulence-informed power laws

- Solve the spatial interpolation problem for each training velocity field.

Property-preserving kernel method
Training and prediction



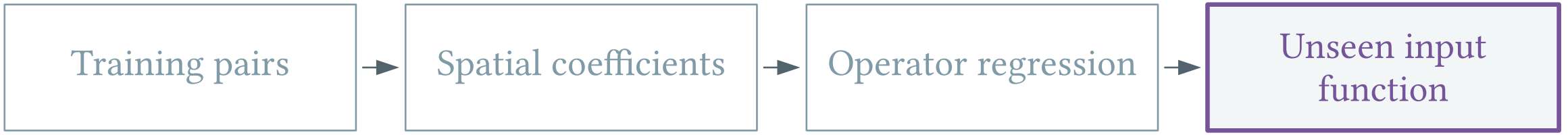
$$K_{ij} = \lambda(\mathbf{a}_i, \mathbf{a}_j)$$

Operator kernel evaluated on the training inputs

$$(K + \theta I)C = B, \qquad B = [\mathbf{b}_1, \ldots, \mathbf{b}_N]^\mathsf{T}$$

- Fit all spatial coefficients with the operator kernel.

Property-preserving kernel method
Training and prediction



$$\mathbf{b}^\star = \sum_{i=1}^{N} \lambda(\mathbf{a}^\star, \mathbf{a}_i)\, \mathbf{c}_i, \qquad \widetilde{\mathbf{v}}^\star(y) = \Phi(y, Y)\mathbf{b}^\star$$

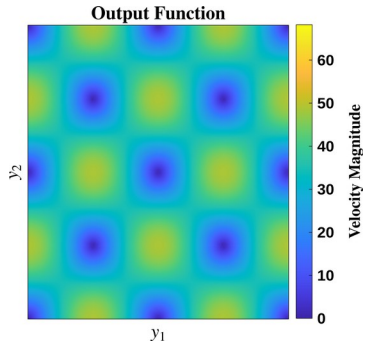
Unseen input function

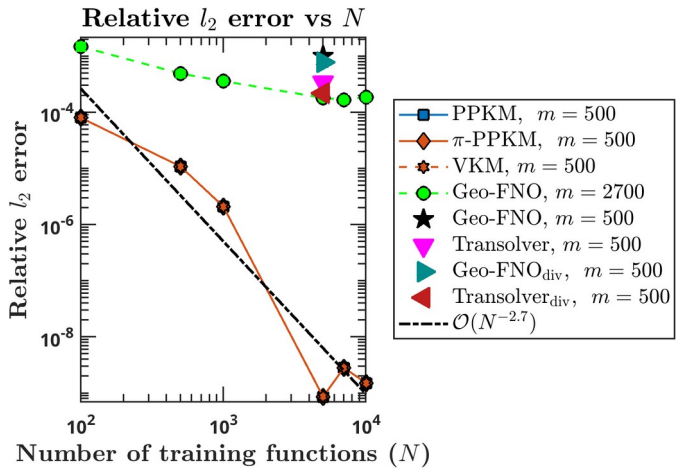
- Evaluate the operator kernel at the unseen input function.
- Predict the spatial coefficients, then evaluate the property-preserving interpolant.

THE UNIVERSITY of UTAH

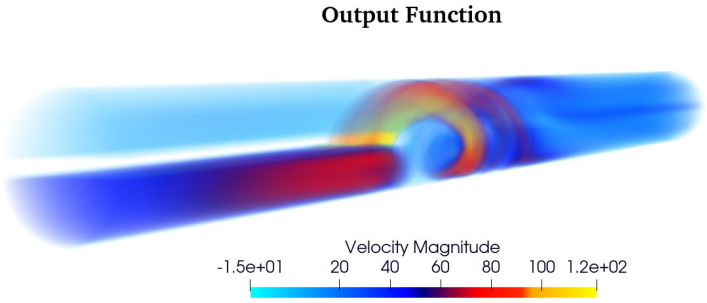
Property-preserving kernel method: accuracy against baseline methods

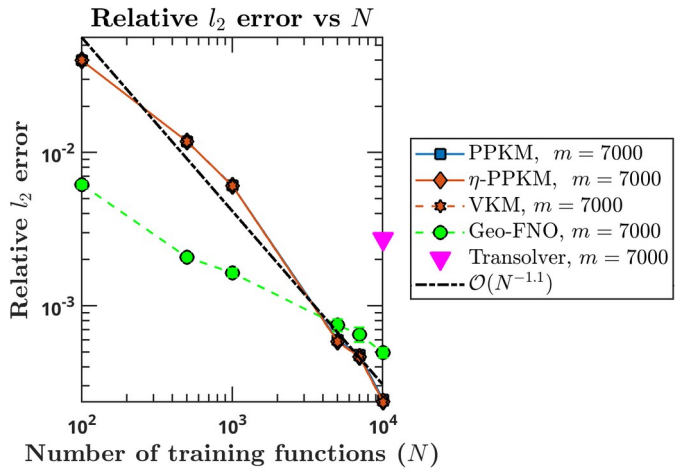
Taylor–Green (spacetime)



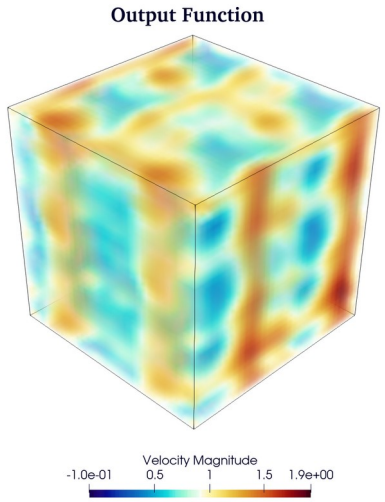


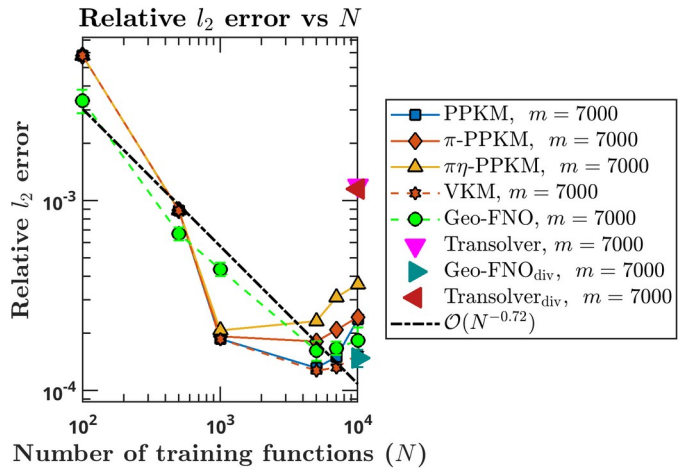
3D species transport





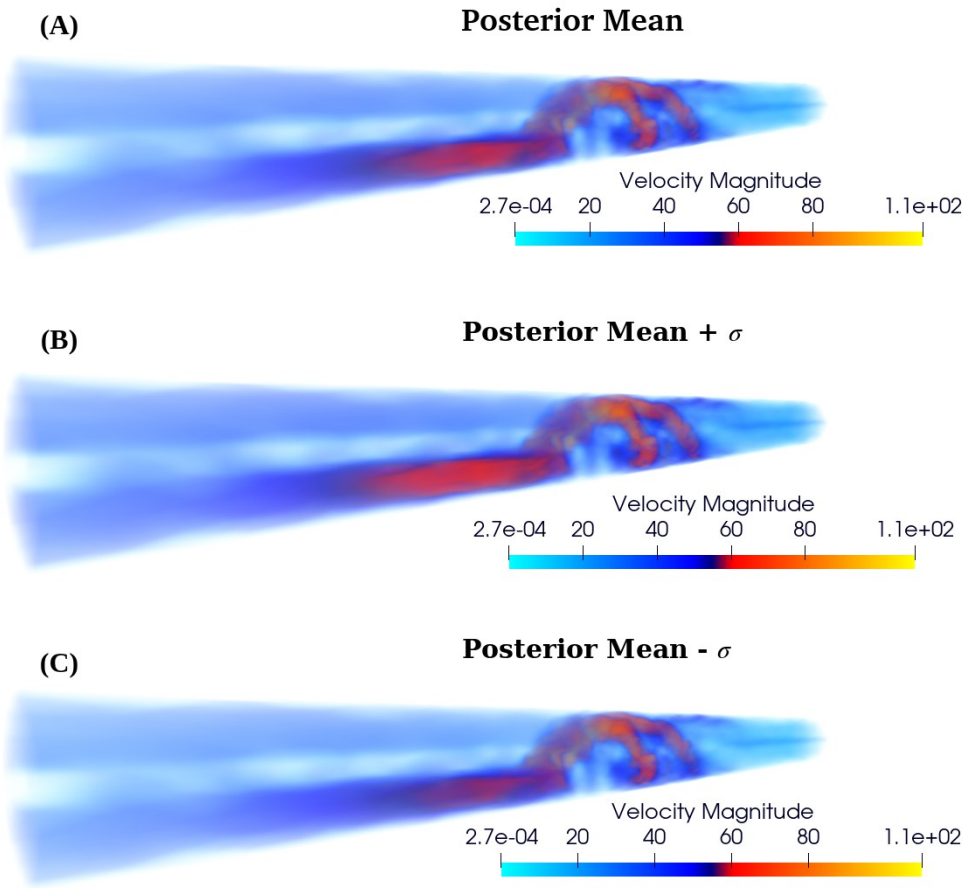
3D forced homogeneous isotropic turbulence





THE UNIVERSITY of UTAH

Property-preserving uncertainty quantification



Posterior mean velocity and corresponding velocity fields within one std. of the mean

$$\overline{\mathbf{b}} = \mathbf{\Lambda}^{\star}(\mathbf{\Lambda} + \theta I)^{-1}\mathbf{b}_{\text{train}}$$

$$\Sigma^{\star} = \mathbf{\Lambda}^{\star\star} - \mathbf{\Lambda}^{\star}(\mathbf{\Lambda} + \theta I)^{-1}(\mathbf{\Lambda}^{\star})^{\mathsf{T}}$$

$\overline{\mathbf{b}}$: posterior mean of the coefficients

$\mathbf{\Lambda}$: prior covariance on training inputs

$\Sigma^{\star}$: posterior covariance of the coefficients

- A Gaussian Process is naturally induced by the operator kernel.

THE UNIVERSITY of UTAH

Scalability challenges

N training functions

Operator matrix: $N \times N$

Storage: $O(N^2)$
Factorization: $O(N^3)$

m spatial points

Spatial matrix: $dm \times dm$

Storage: $O((dm)^2)$
Factorization: $O((dm)^3)$

- Streaming formation helps assembly, it does not eliminate dense factorization.

A solution on both sides

Operator side

- Introduce sparsity in the operator kernel.

$N \times N$ dense matrix

↓

sparse interactions

Spatial side

- Use a localized property-preserving basis.

one global spatial solve

↓

fixed size local systems

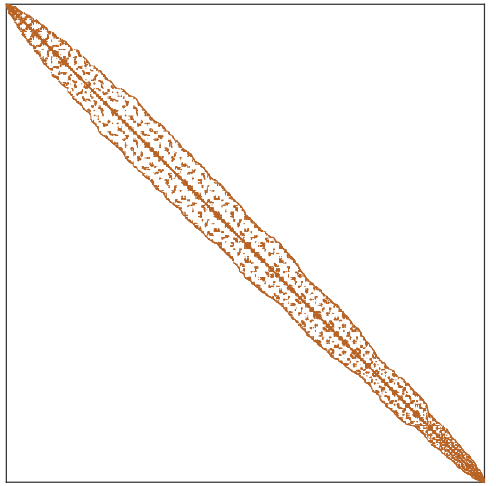
Sparse operator kernel

$$\lambda(\mathbf{a}, \mathbf{a}') = \eta(\mathbf{a})\eta(\mathbf{a}') \; w\left(\frac{\|D_{\|}(z - z')\|_2}{\varrho}\right) \; M_{5/2}\left(\frac{\alpha_{\perp}\|\zeta - \zeta'\|_2}{\delta}\right)$$

Gate **C⁴ Wendland** **Matérn 5/2**

z: up to 3 principal coordinates, ζ: the residual outside that subspace.

$$\widetilde{f}(\mathbf{a}) = \sum_i \lambda(\mathbf{a}, \mathbf{a}_i)\mathbf{c}_i + \underbrace{\boldsymbol{\beta}^{\mathsf{T}} p(\mathbf{a})}_{\text{affine polynomial terms}}$$



- Affine augmentation reproduces constant and linear trends in the principal coordinates.

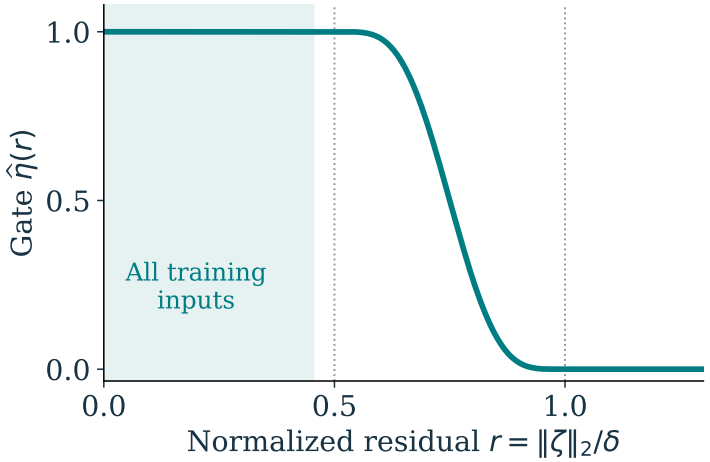
THE UNIVERSITY of UTAH

Operator kernel: the residual gate

$$\lambda(\mathbf{a}, \mathbf{a}') = \eta(\mathbf{a})\eta(\mathbf{a}') \; w\left(\frac{\|D_{\|}(z - z')\|_2}{\varrho}\right) \; M_{5/2}\left(\frac{\alpha_{\perp}\|\zeta - \zeta'\|_2}{\delta}\right)$$

Gate | C^4 Wendland | Matérn 5/2

$$r = \frac{\|\zeta(\mathbf{a})\|_2}{\delta}, \qquad \widehat{\eta}(r) = \begin{cases} 1, & r \le \frac{1}{2}, \\ H(2(1-r)), & \frac{1}{2} < r < 1, \\ 0, & r \ge 1. \end{cases}$$



- Large residuals turn off the kernel contribution.
- Outside the gate's support, prediction falls back to the affine model.

THE UNIVERSITY of UTAH

Operator kernel: compact support

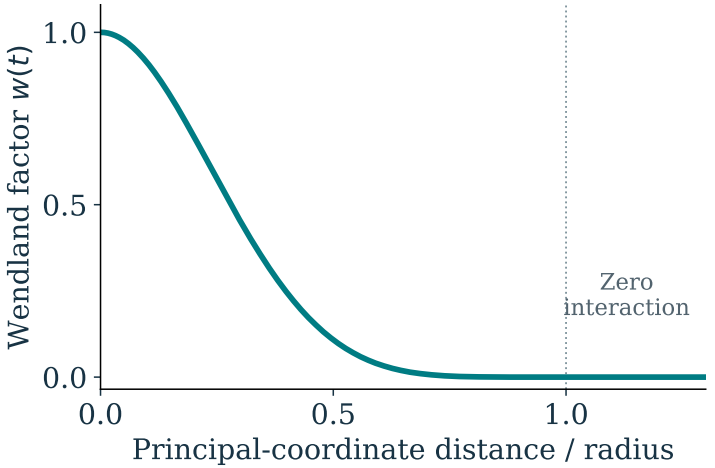
$$\lambda(\mathbf{a}, \mathbf{a}') = \eta(\mathbf{a})\eta(\mathbf{a}') \; w\left(\frac{\|D_{\|}(z - z')\|_2}{\varrho}\right) \; M_{5/2}\left(\frac{\alpha_{\perp}\|\zeta - \zeta'\|_2}{\delta}\right)$$

Gate | **C⁴ Wendland** | Matérn 5/2

$$z(\mathbf{a}) = V_z^{\mathsf{T}}\bar{\mathbf{a}}, \qquad t = \frac{\|D_{\|}(z - z')\|_2}{\varrho}$$

$$w(t) = (1 - t)_+^6\left(1 + 6t + \frac{35}{3}t^2\right)$$

- Uses at most three principal coordinates.
- Introduces sparsity in the final kernel.



THE UNIVERSITY of UTAH

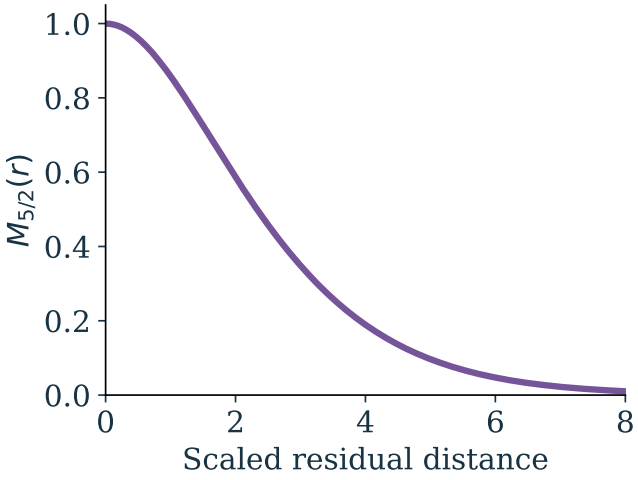
Operator kernel: residual similarity

$$\lambda(\mathbf{a}, \mathbf{a}') = \eta(\mathbf{a})\eta(\mathbf{a}') \; w\left(\frac{\|D_{\|}(z - z')\|_2}{\varrho}\right) \; M_{5/2}\left(\frac{\alpha_{\perp}\|\zeta - \zeta'\|_2}{\delta}\right)$$

Gate | C^4 Wendland | **Matérn 5/2**

$$\zeta(\mathbf{a}) = (I - V_z V_z^{\mathsf{T}})\bar{\mathbf{a}}$$

$$M_{5/2}(r) = \left(1 + r + \frac{r^2}{3}\right) e^{-r}$$



- Accounts for input differences outside the leading principal subspace.
- Positive definite in any residual dimension.

THE UNIVERSITY of UTAH

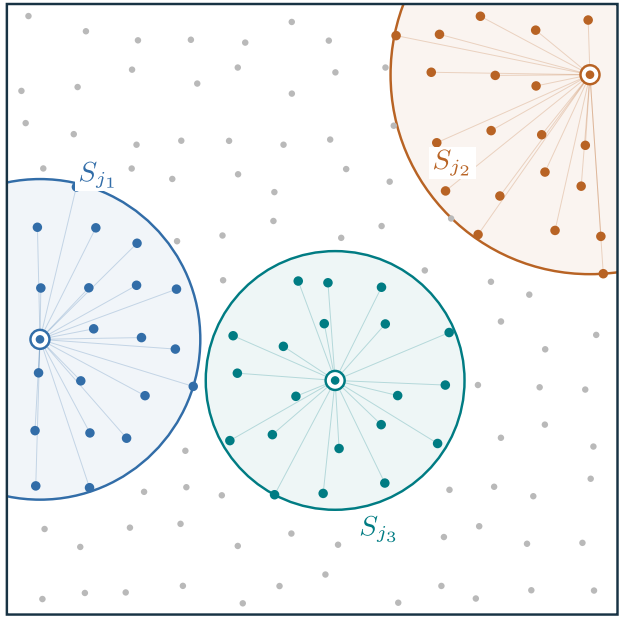
Localized property-preserving kernels

Each output node defines a stencil: a small set of nearby nodes used for local reconstruction.

$$\chi_i^j(y) = \sum_{l=1}^{n_s} \Phi_{\mathrm{DF}}(y, \tilde{y}_{j,l}) \mathbf{b}_{i,\Phi}^{j,l} + \mathbf{P}_j(y) \mathbf{b}_{i,P}^j$$

i: training field j: stencil l: node within the stencil

- Divergence-free kernels + divergence-free polynomials (every column of $\mathbf{P}_j(y)$ is divergence free).
- Periodic stencils and multiscale Matérn extensions.



THE UNIVERSITY of UTAH

Localized property-preserving kernels

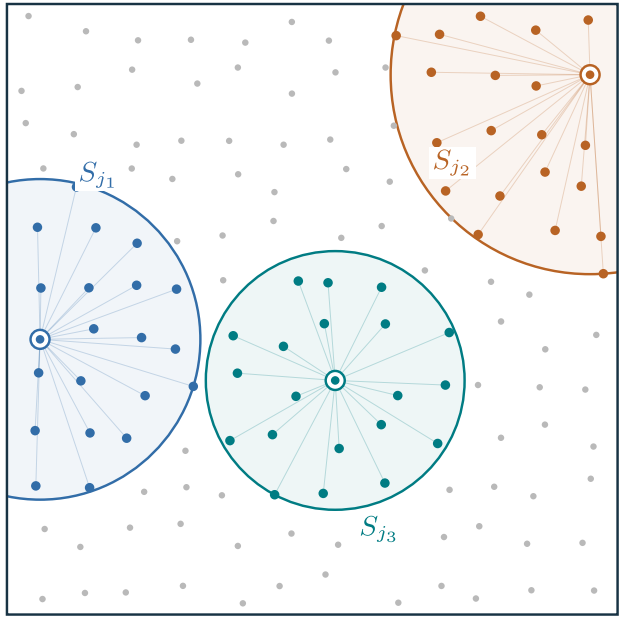
Each output node defines a stencil: a small set of nearby nodes used for local reconstruction.

$$\chi_i^j(y) = \sum_{l=1}^{n_s} \Phi_{\mathrm{DF}}(y, \tilde{y}_{j,l}) \mathbf{b}_{i,\Phi}^{j,l} + \mathbf{P}_j(y) \mathbf{b}_{i,P}^{j}$$

i: training field j: stencil l: node within the stencil

- Divergence-free kernels + divergence-free polynomials (every column of $\mathbf{P}_j(y)$ is divergence free).

- Periodic stencils and multiscale Matérn extensions.



THE UNIVERSITY of UTAH

Divergence-free polynomials from potentials

Polynomial reproduction provides a target approximation order for smooth fields.

$$\xi = \frac{y - \bar{y}_j}{\rho_j}, \qquad J_\alpha(\xi) = \prod_{k=1}^{d} P_{\alpha_k}(\xi_k), \qquad |\alpha| \le \varpi + 1$$

J_α uses normalized Legendre polynomials.

$$\text{2D:} \quad g_\alpha(y) = \nabla_\xi^\perp J_\alpha(\xi)$$

$$\text{3D:} \quad g_{\alpha,k}(y) = \nabla_\xi \times \left(J_\alpha(\xi)\mathbf{e}_k\right), \quad k = 1, 2, 3$$

- Candidates span divergence-free vector polynomials through degree ϖ.
- Some candidates vanish or repeat.

QR selects and normalizes the polynomial basis

Use column-pivoted QR on the sampled fields to remove redundancy and normalize the basis.

$$\underbrace{G_{\text{selected}}}_{\text{sampled candidate fields}} = \underbrace{Q_{\text{kept}}}_{\text{orthonormal columns}} \underbrace{R_{\text{kept}}}_{\text{mixing and scaling}}$$

Apply the same change of basis to the selected polynomial functions.

$$\mathbf{P}_j(y) := G_{\text{selected}}(y) R_{\text{kept}}^{-1}$$

At the stencil nodes, the new basis evaluates to the orthonormal columns of Q_{kept}.

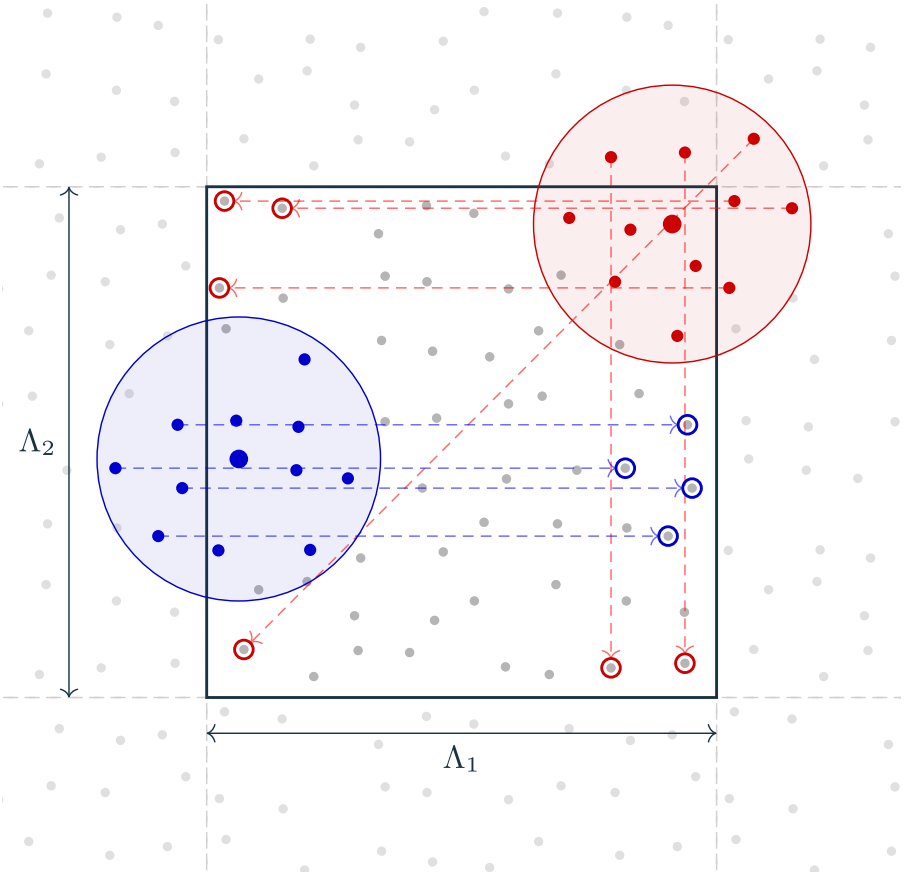
- Divergence-free basis fields with orthonormal stencil samples.
- Use this basis in each stencil's interpolation and polynomial moment equations.

Periodic boundaries through ghost points

- Wrap each displacement to its nearest periodic image.
- Select the nearest n_s representatives, one per original node.
- Each ghost point carries its original node's velocity.

$$\delta_l = y_l - y_j\,,$$

$$\delta_{l,k} \leftarrow \delta_{l,k} - \Lambda_k \,\mathrm{round}\left(\frac{\delta_{l,k}}{\Lambda_k}\right)$$



Multiscale Matérn kernels for turbulent outputs

- Rescaling a polyharmonic spline (PHS) changes amplitude, not length scale.
- For turbulence, replace the PHS by five Matérn scales and keep the polynomials.

$$\chi_i^j(y) = \sum_{l=1}^{n_s} \Phi_{\mathrm{PHS}}^{\mathrm{DF}}(y, \tilde{y}_{j,l})\, \mathbf{b}_{i,\Phi}^{j,l} + \mathbf{P}_j(y)\mathbf{b}_{i,P}^j$$

Divergence-free PHS

Divergence-free polynomials

Multiscale Matérn kernels for turbulent outputs

- Rescaling a polyharmonic spline (PHS) changes amplitude, not length scale.
- For turbulence, replace the PHS by five Matérn scales and keep the polynomials.

$$\chi_i^j(y) = \sum_{l=1}^{n_s} \Phi_j^{\mathrm{M}}(y, \tilde{y}_{j,l})\, \mathbf{b}_{i,\Phi}^{j,l} + \mathbf{P}_j(y)\mathbf{b}_{i,P}^j$$

Divergence-free multiscale Matérn

Divergence-free polynomials

$$\Phi_j^{\mathrm{M}}(x,y) = \frac{1}{c_j} \sum_{\kappa=1}^{5} w_{j,\kappa}\, \Phi_{\mathrm{M,DF}}(x, y; \sigma_{j,\kappa})$$

$\Phi_{\mathrm{M,DF}}(x, y; \sigma)$: divergence-free Matérn $5/2$ kernel at scale σ

$$\sigma_{j,\kappa} = \frac{16\rho_j}{2^\kappa}, \qquad w_{j,\kappa} = 2^{-8\kappa/3}, \qquad \kappa = 1, \ldots, 5$$

ρ_j : stencil radius. Choose the scalar c_j so that $\Phi_j^{\mathrm{M}}(y,y) = I$.

THE UNIVERSITY of UTAH

Lagrange evaluation - motivation

Each training velocity field gives a row of spatial coefficients.

$$K_{\text{train}} \underbrace{C}_{\text{fitted operator coefficients}} = \underbrace{B_{\text{train}}}_{\text{training spatial coefficients}}$$

- The same input kernel matrix is used for every output column.
- Can we predict them all with one set of weights on the training examples?

Direct vs Lagrange: change the evaluation order

b_{train} contains N training coefficients. b_{pred} contains predictions at new inputs.

$$K_{\text{train}} = K^{\theta}_{UU} = K + \theta I, \qquad K_{\text{test}} = K_{U_* U}$$

Direct evaluation

$$\mathbf{b}_{\text{pred}} = K_{\text{test}} \underbrace{\left(K^{-1}_{\text{train}} \mathbf{b}_{\text{train}}\right)}_{\mathbf{c}\text{: fit at training}}$$

Lagrange evaluation

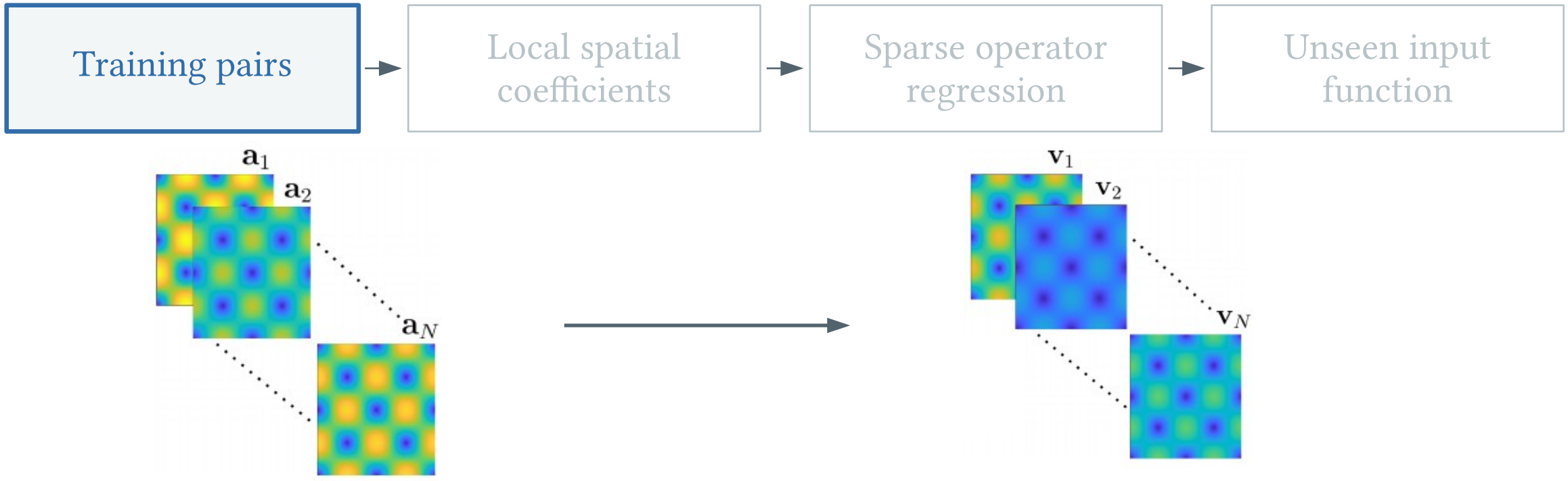
$$\mathbf{b}_{\text{pred}} = \underbrace{\left(K_{\text{test}} K^{-1}_{\text{train}}\right)}_{L\text{: weights for new inputs}} \mathbf{b}_{\text{train}}$$

Solve for L. Each row gives weights for one new input.

$$K^{\theta}_{UU} L^{\mathsf{T}} = K^{\mathsf{T}}_{U_* U}, \qquad L_{r,:} = \ell(\mathbf{a}^{\star}_r)^{\mathsf{T}}$$

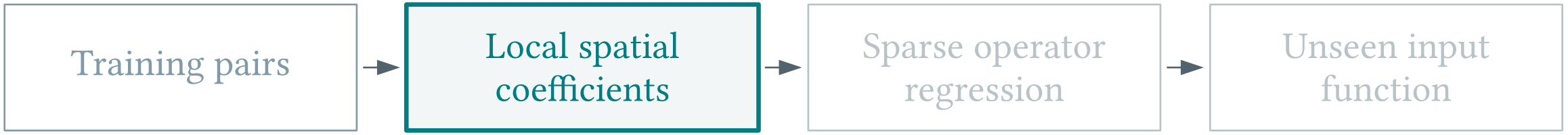
THE UNIVERSITY of UTAH

Sparse and local property-preserving kernel method
Training and prediction



- Start with the same pairs of input and output functions.

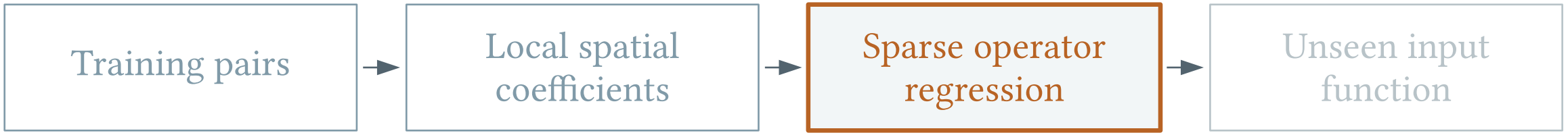
Sparse and local property-preserving kernel method
Training and prediction



$$\begin{bmatrix} \mathbf{\Phi}_j & \mathbf{P}_j \\ \mathbf{P}_j^\intercal & 0 \end{bmatrix} \begin{bmatrix} \mathbf{b}_{i,\Phi}^j \\ \mathbf{b}_{i,P}^j \end{bmatrix} = \begin{bmatrix} \mathbf{v}_i|_{S_j} \\ 0 \end{bmatrix}$$

- Solve a small spatial system on each stencil for each training field.
- Local divergence-free kernels and polynomials encode desirable properties.

Sparse and local property-preserving kernel method
Training and prediction

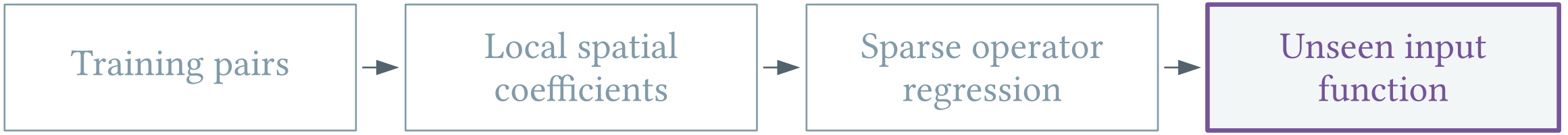


$$\begin{bmatrix} K + \theta I & P \\ P^{\mathsf{T}} & 0 \end{bmatrix} \begin{bmatrix} \ell^\star \\ \mu^\star \end{bmatrix} = \begin{bmatrix} \boldsymbol{\lambda}^\star \\ p(\mathbf{a}^\star) \end{bmatrix}, \qquad \lambda_i^\star = \lambda(\mathbf{a}^\star, \mathbf{a}_i)$$

Factor sparse K + θI once.
For a⋆, solve for the shared weights ℓ⋆.

- The sparse operator kernel using Lagrange weights replaces the dense input Gramian.

Sparse and local property-preserving kernel method
Training and prediction



$$\mathbf{b}_*^j = B_j^\mathsf{T} \ell^\star$$

B_j stores both kernel and polynomial coefficients.

$$\widetilde{\mathbf{v}}^\star(y) = \begin{bmatrix} \mathbf{\Phi}_j(y) & \mathbf{P}_j(y) \end{bmatrix} \begin{bmatrix} \mathbf{b}_{*,\Phi}^j \\ \mathbf{b}_{*,P}^j \end{bmatrix}$$

- Reconstruct the velocity from the predicted kernel and polynomial coefficients.
- Evaluate the local basis at each spatial query point.

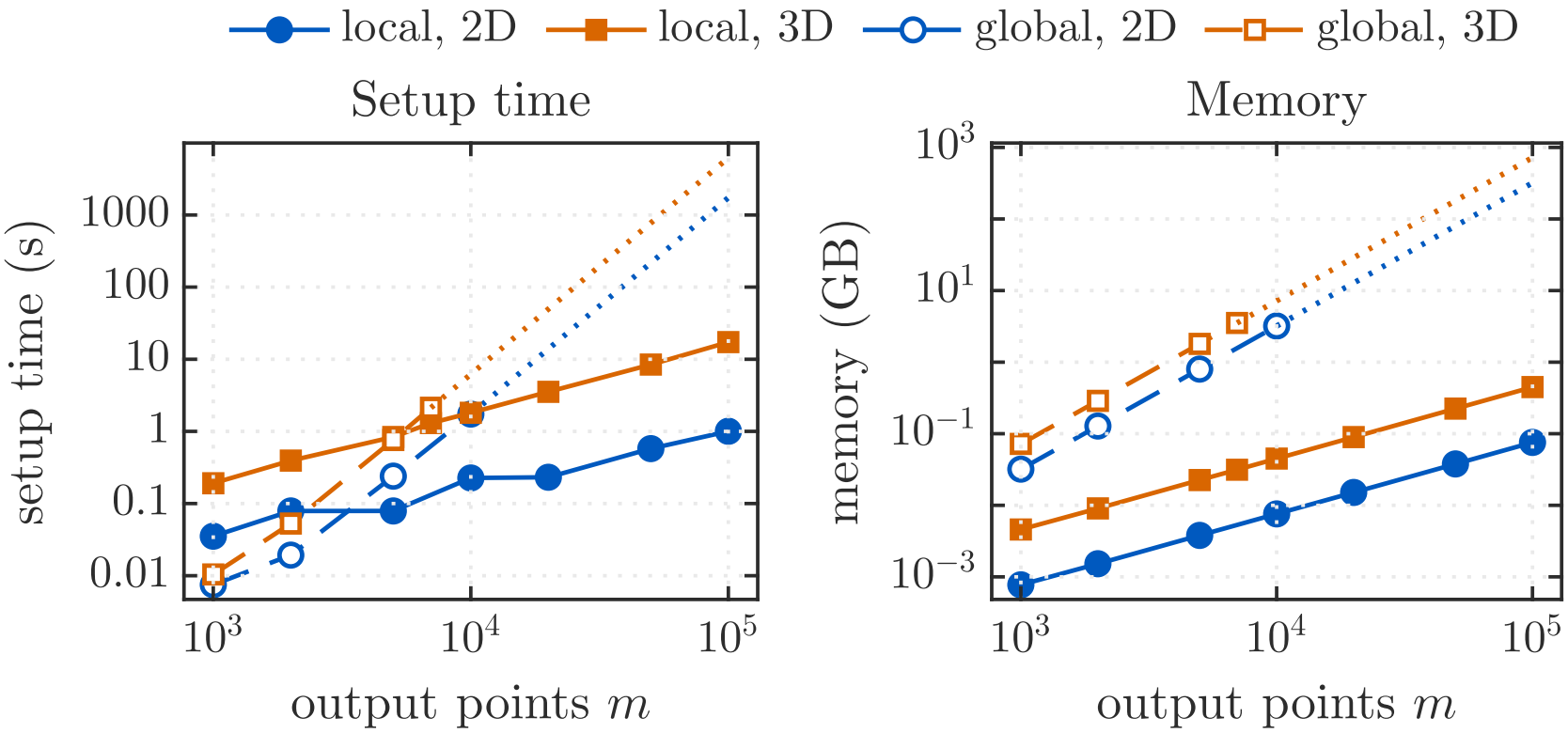
Accuracy against the property-preserving kernel method and baseline methods

Data set	N	Sparse-Local-PPKM	PPKM	VKM	Geo-FNO	Transolver
Forced isotropic turbulence	10000	$\mathbf{7.1 \times 10^{-5}}$	3.6×10^{-4}	1.6×10^{-4}	1.5×10^{-4}	1.2×10^{-3}
Turbulent species transport	10000	$\mathbf{3.7 \times 10^{-5}}$	2.4×10^{-4}	2.4×10^{-4}	5.0×10^{-4}	2.7×10^{-3}
Merging vortices	500	$\mathbf{1.9 \times 10^{-6}}$	1.1×10^{-4}	1.1×10^{-4}	1.1×10^{-3}	3.4×10^{-3}
Lid-driven cavity	10000	$\mathbf{4.7 \times 10^{-7}}$	1.4×10^{-6}	1.5×10^{-6}	1.5×10^{-4}	2.1×10^{-4}
Cylinder flow (no shedding)	100	$\mathbf{2.8 \times 10^{-5}}$	4.0×10^{-5}	5.3×10^{-5}	1.6×10^{-4}	5.7×10^{-3}
Cylinder flow (vortex shedding)	10000	$\mathbf{3.0 \times 10^{-7}}$	3.1×10^{-7}	2.3×10^{-6}	3.7×10^{-5}	1.5×10^{-4}
Backward-facing step	500	$\mathbf{9.5 \times 10^{-8}}$	1.5×10^{-6}	2.9×10^{-6}	9.3×10^{-5}	7.2×10^{-4}
Buoyancy-driven cavity	10000	$\mathbf{4.5 \times 10^{-6}}$	6.3×10^{-6}	6.1×10^{-6}	1.4×10^{-4}	1.9×10^{-4}
Taylor–Green vortices (space)	5000	$\mathbf{6.0 \times 10^{-11}}$	8.8×10^{-10}	1.7×10^{-9}	4.2×10^{-4}	6.1×10^{-5}
Taylor–Green vortices (spacetime)	5000	$\mathbf{5.4 \times 10^{-11}}$	8.7×10^{-10}	8.7×10^{-10}	7.8×10^{-4}	2.2×10^{-4}

Selected operator matrices are about 10% dense on most problems.
Forced homogeneous isotropic turbulence is the exception (~90% dense).

THE UNIVERSITY of UTAH

Faster and more scalable reconstruction



- Fixed size local systems reduce output setup time and memory.
- Dotted global curves are extrapolations beyond GPU capacity.

THE UNIVERSITY of UTAH

Lagrange evaluation accuracy advantage

	Global output reconstruction			Local output reconstruction	
Data set	Models	Median	Max	Models	Max $\lvert\text{ratio}-1\rvert$
Forced isotropic turbulence	28	1.00	1.2	28	5.9×10^{-11}
Turbulent species transport	28	1.00	7.1	28	8.6×10^{-7}
Merging vortices	28	48.96	868.1	28	4.2×10^{-8}
Lid-driven cavity	28	5.07	34.5	28	1.4×10^{-7}
Cylinder flow (no shedding)	28	1.05	4.8	28	7.0×10^{-8}
Cylinder flow (vortex shedding)	28	1.01	2.0	28	2.4×10^{-8}
Backward-facing step	28	16.51	215.3	28	2.0×10^{-8}
Buoyancy-driven cavity	28	1.47	14.8	28	1.9×10^{-8}
Taylor–Green vortices (space)	28	1.05	6.1	28	1.7×10^{-6}
Taylor–Green vortices (spacetime)	28	1.00	1.0	28	2.4×10^{-6}

Direct test error / Lagrange test error; same unaugmented models.
Values >1 favor Lagrange; local accuracy is essentially unchanged.

THE UNIVERSITY of UTAH

Lagrange evaluation accuracy advantage

	Global output reconstruction			Local output reconstruction	
Data set	Models	Median	Max	Models	Max $\lvert\text{ratio}-1\rvert$
Forced isotropic turbulence	28	1.00	1.2	28	5.9×10^{-11}
Turbulent species transport	28	1.00	7.1	28	8.6×10^{-7}
Merging vortices	28	48.96	868.1	28	4.2×10^{-8}
Lid-driven cavity	28	5.07	34.5	28	1.4×10^{-7}
Cylinder flow (no shedding)	28	1.05	4.8	28	7.0×10^{-8}
Cylinder flow (vortex shedding)	28	1.01	2.0	28	2.4×10^{-8}
Backward-facing step	28	16.51	215.3	28	2.0×10^{-8}
Buoyancy-driven cavity	28	1.47	14.8	28	1.9×10^{-8}
Taylor–Green vortices (space)	28	1.05	6.1	28	1.7×10^{-6}
Taylor–Green vortices (spacetime)	28	1.00	1.0	28	2.4×10^{-6}

Direct test error / Lagrange test error; same unaugmented models.
Values >1 favor Lagrange; local accuracy is essentially unchanged.

THE UNIVERSITY of UTAH

Conclusions

- We propose a property preserving kernel method for learning incompressible flows.

- Across classical and challenging benchmarks, our errors are up to 2x to 10^6x lower than the baseline kernel and neural operators.

- Local reconstruction reduces setup cost and memory. Lagrange evaluation makes the operator kernel solve more stable.

- These results support the method's extensions to larger flow applications while retaining physical constraints in the reconstruction *analytically*.

Thank you

Fluids You Can Trust: Property-Preserving Operator Learning for Incompressible Flows

https://arxiv.org/abs/2602.15472

Local and Sparse Kernel Operator Learning for Incompressible Flows: An Operator Lagrange Formulation

https://arxiv.org/abs/2610.29009

Funding support

National Science Foundation
DMS 2505986 · Cooperative Agreement 2421782

Simons Foundation
MPS-AI-00010515
NSF–Simons AI Institute for Cosmic Origins (CosmicAI)

Air Force Office of Scientific Research
LRIR FA9550-25-1-0042
MURI FOA-AFRL-AFOSR-2023-0004

Department of Energy · DE-SC0023163

DoD Vannevar Bush Faculty Fellowship
ONR N00014-18-1-2363

Computing support

NCSA Delta through ACCESS
OAC 2005572 · Allocation CIS260011
ACCESS NSF grants 2138259, 2138286, 2138307, 2137603, 2138296

THE UNIVERSITY of UTAH