Shape Derivative-Informed Reduced Basis Neural Operator

with applications to shape optimization under uncertainty

Xindi Gong*, Dingcheng Luo◇, Thomas O'Leary-Roseberry†, Ruanui Nicholson‡, Omar Ghattas*

(*):The University of Texas at Austin, (◇): Queensland University of Technology, (†): The Ohio State University (‡): The University of Auckland

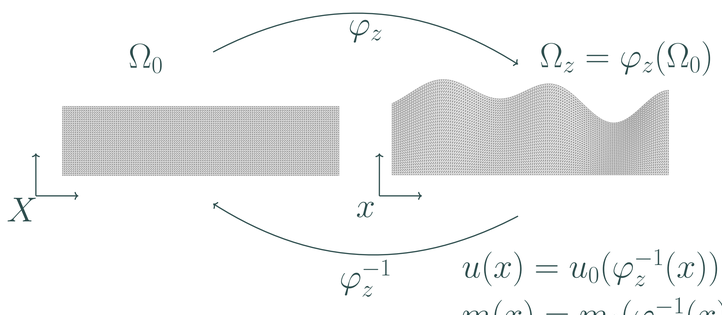
OPTIMUS

ODEN INSTITUTE FOR COMPUTATIONAL ENGINEERING & SCIENCES

Key idea of Shape-DINOs

- Provide formulations on reference domain via Piola transformation φ_z, a diffeomorphism, to avoid mesh movement.
- Design derivative-informed neural operators for shape parametric PDE problems, which provide accurate Jacobian training in reduced basis architectures.
- Use Shape-DINOs for shape optimization under uncertainty (OUU) problems.

We formulate parametric PDE problems in the reference domain Ω_0 via Piola transformation:



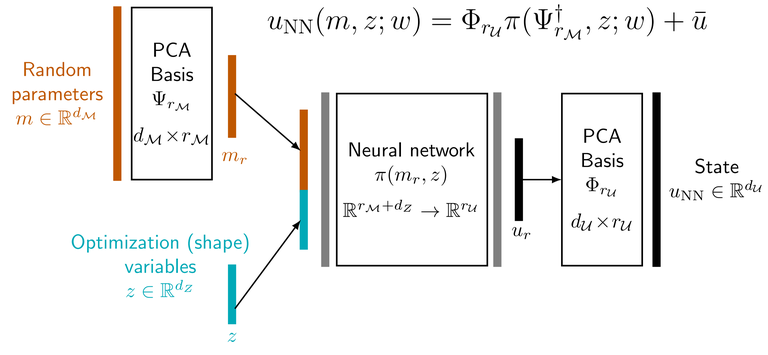
$$u(x) = u_0(\varphi_z^{-1}(x))$$
$$m(x) = m_0(\varphi_z^{-1}(x))$$

$$R_0(u_0, m_0, z) \quad \Longleftrightarrow \quad R(u, m, z)$$

With risk measure ρ, PDE-constrained OUU problem:

$$\min_{z \in Z_{\text{ad}}} J(z) \coloneqq \rho(Q)(z) + \alpha P(z) \qquad (1)$$
$$\text{s.t.} \quad R_0(u_0, m_0, z) = 0$$

Shape-DINO Construction

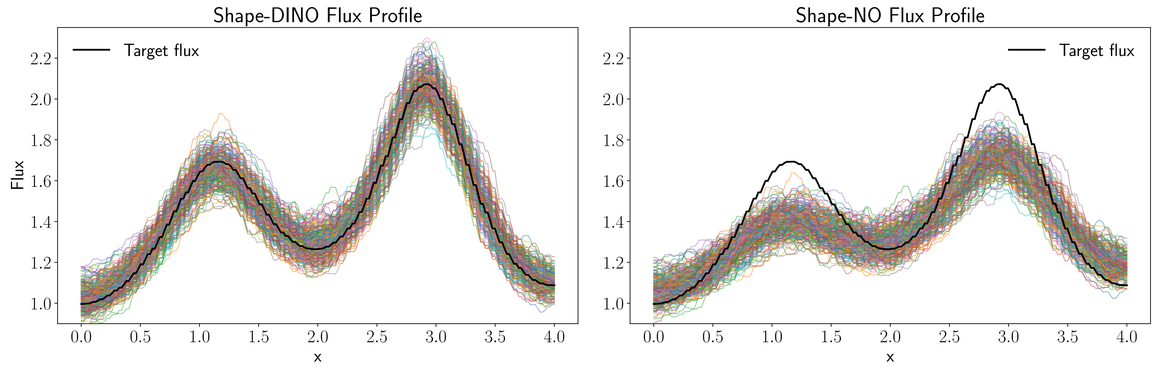


$$u_{\text{NN}}(m, z; w) = \Phi_{r_\mathcal{U}} \pi(\Psi^\dagger_{r_\mathcal{M}}, z; w) + \bar{u}$$

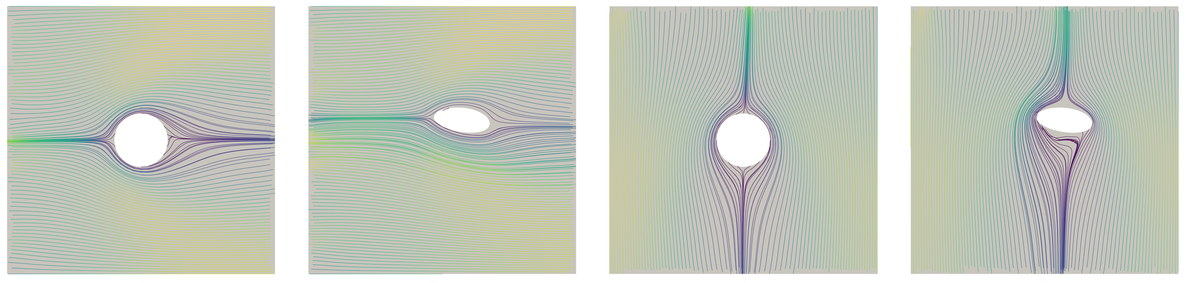
$$\min_w \mathbb{E}_{\nu_m \otimes \nu_z} \left[\|u_{\text{NN}}(m,z) - u(m,z)\|^2_\mathcal{U} + \|\partial_m u_{\text{NN}}(m,z) - \partial_m u(m,z)\|^2_{\text{HS}(\mathcal{M},\mathcal{U})} + \|\partial_z u_{\text{NN}}(m,z) - \partial_z u(m,z)\|^2_{\text{HS}(\mathcal{Z},\mathcal{U})} \right]$$

Poisson Problem with Varying Top Boundary

The shape of the top boundary is optimized to match target flux across the bottom boundary.

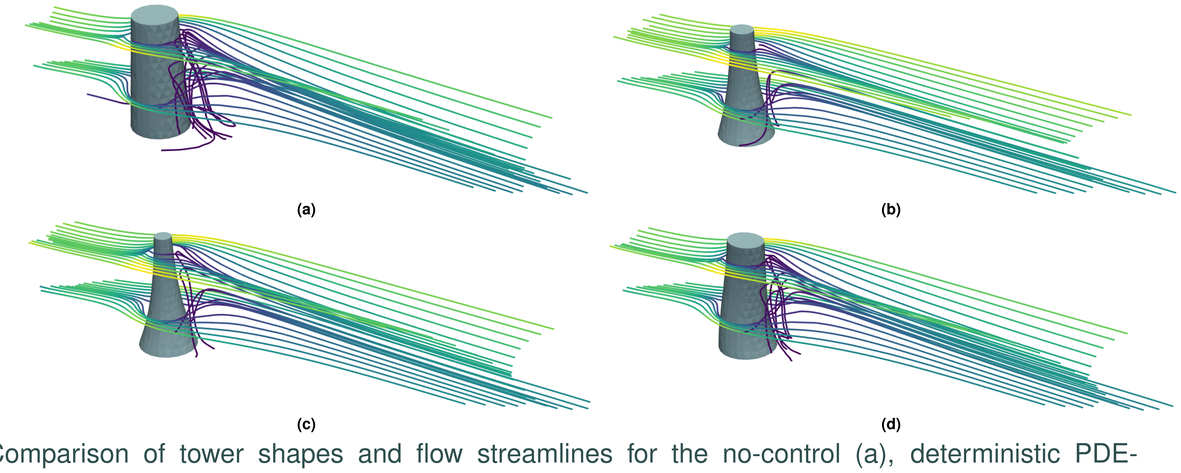


2D Pellet Shape Optimization with Mixed Inflow Conditions



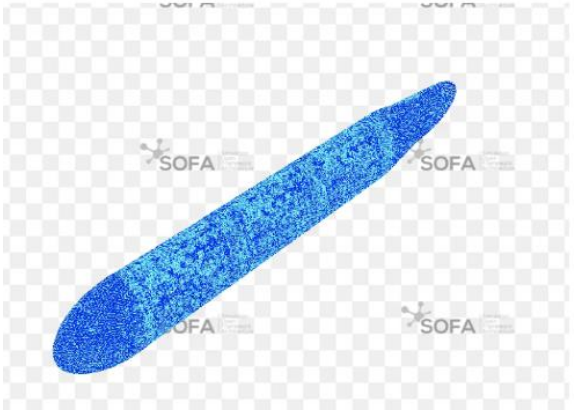
Comparison of flow fields in initial shape and optimal pellet shape generated by Shape-DINO.

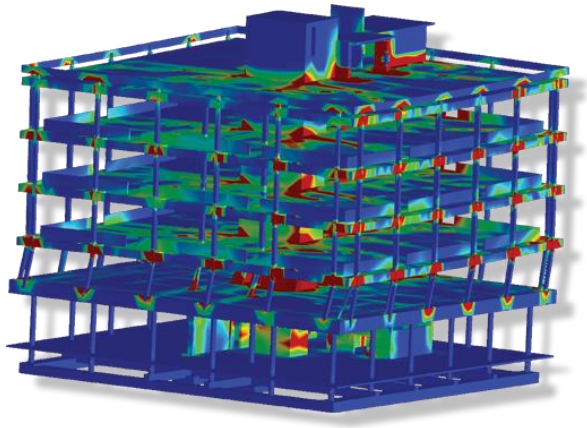
3D Tower Shape Optimization with Navier-Stokes Equations



Comparison of tower shapes and flow streamlines for the no-control (a), deterministic PDE-optimized (b), Shape-DINO-optimized (c), and Shape-NO-optimized (d) designs.

Nonlinear mechanical models in structural and mechanical engineering often possess an underlying Lagrangian structure

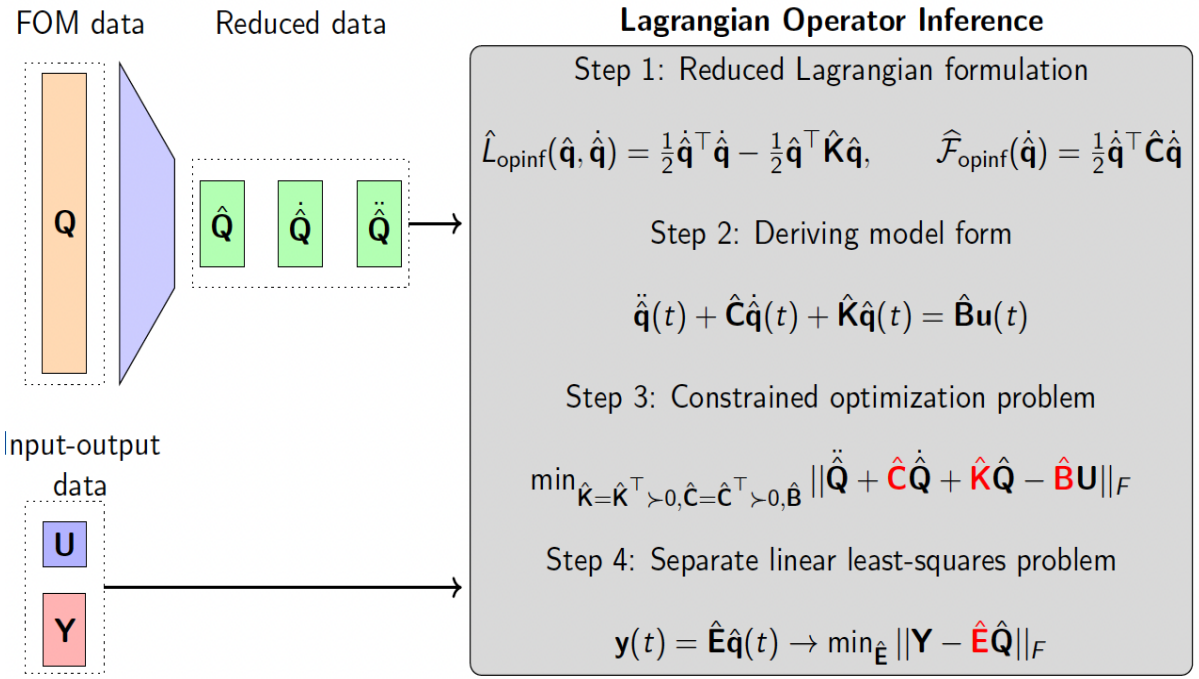


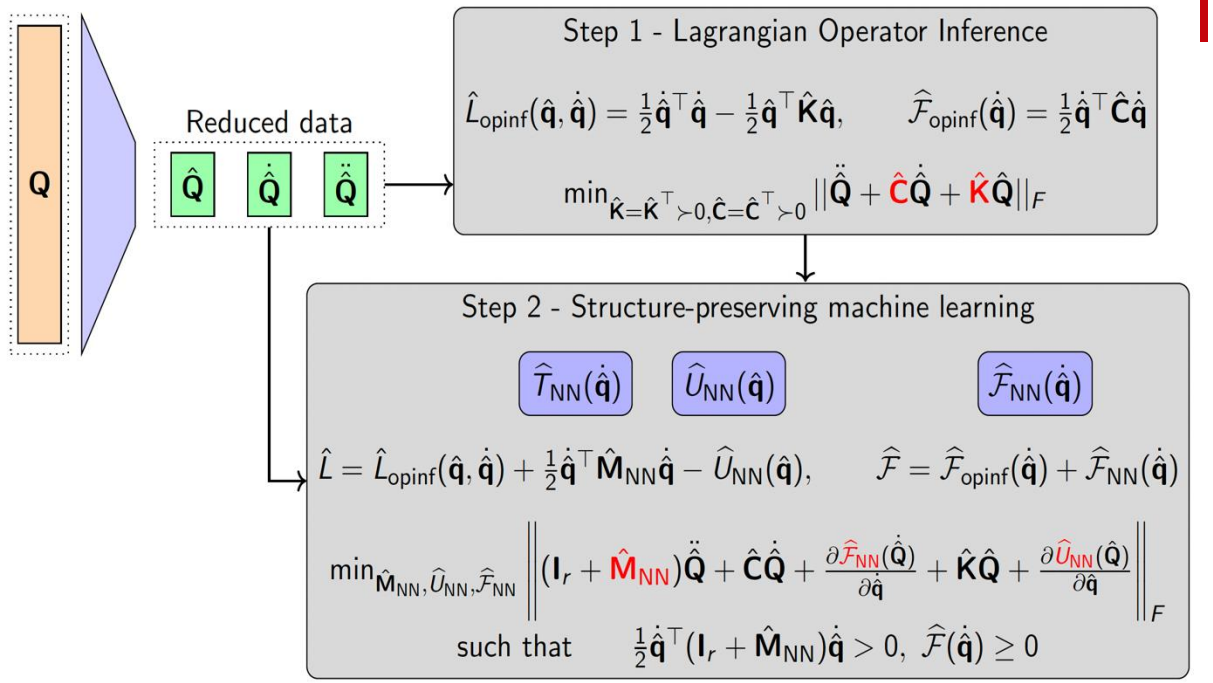


$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \frac{\partial \mathcal{F}_{\text{nl}}(\dot{\mathbf{q}})}{\partial \dot{\mathbf{q}}} + \mathbf{K}\mathbf{q} + \frac{\partial U_{\text{nl}}(\mathbf{q})}{\partial \mathbf{q}} = \mathbf{B}\mathbf{u}(t)$$

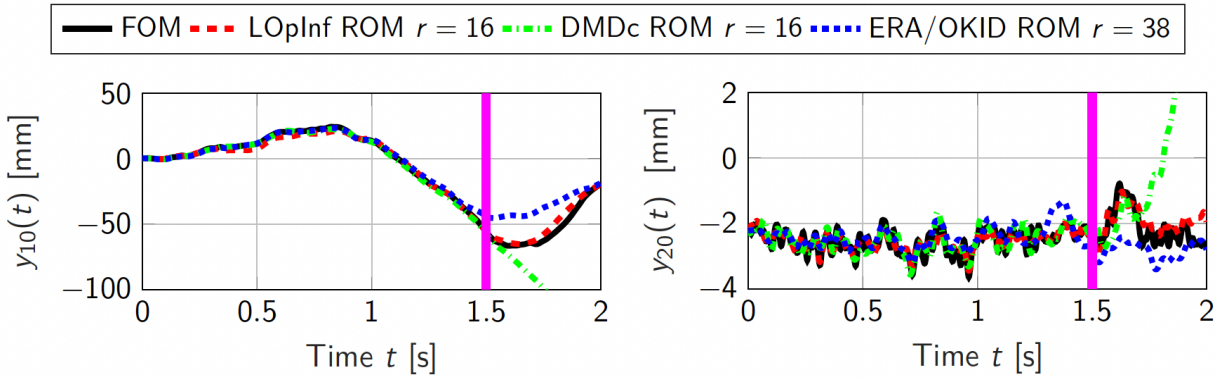
However, standard data-driven ROMs of high-dimensional mechanical models are not guaranteed to be Lagrangian

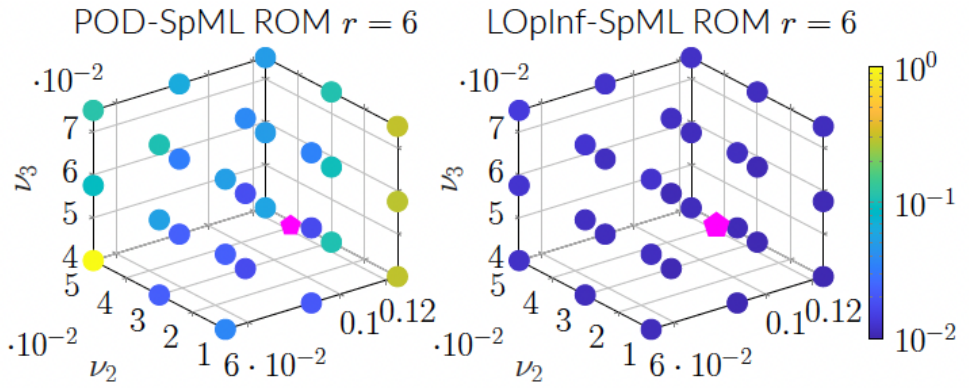
Structure-preserving Operator Inference for learning Lagrangian ROMs



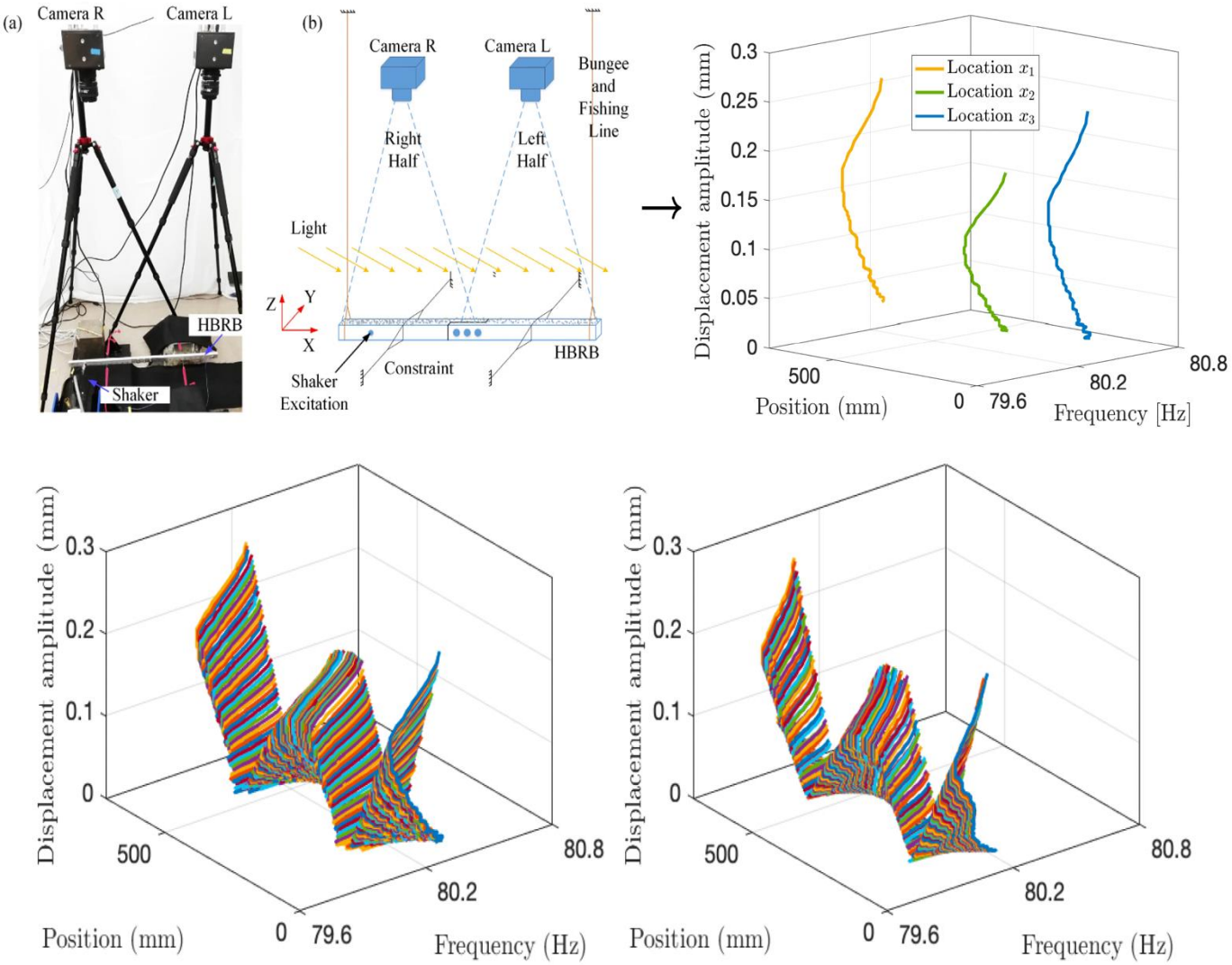


Learned ROMs provide accurate predictions both outside the training time interval, as well as for unseen initial conditions





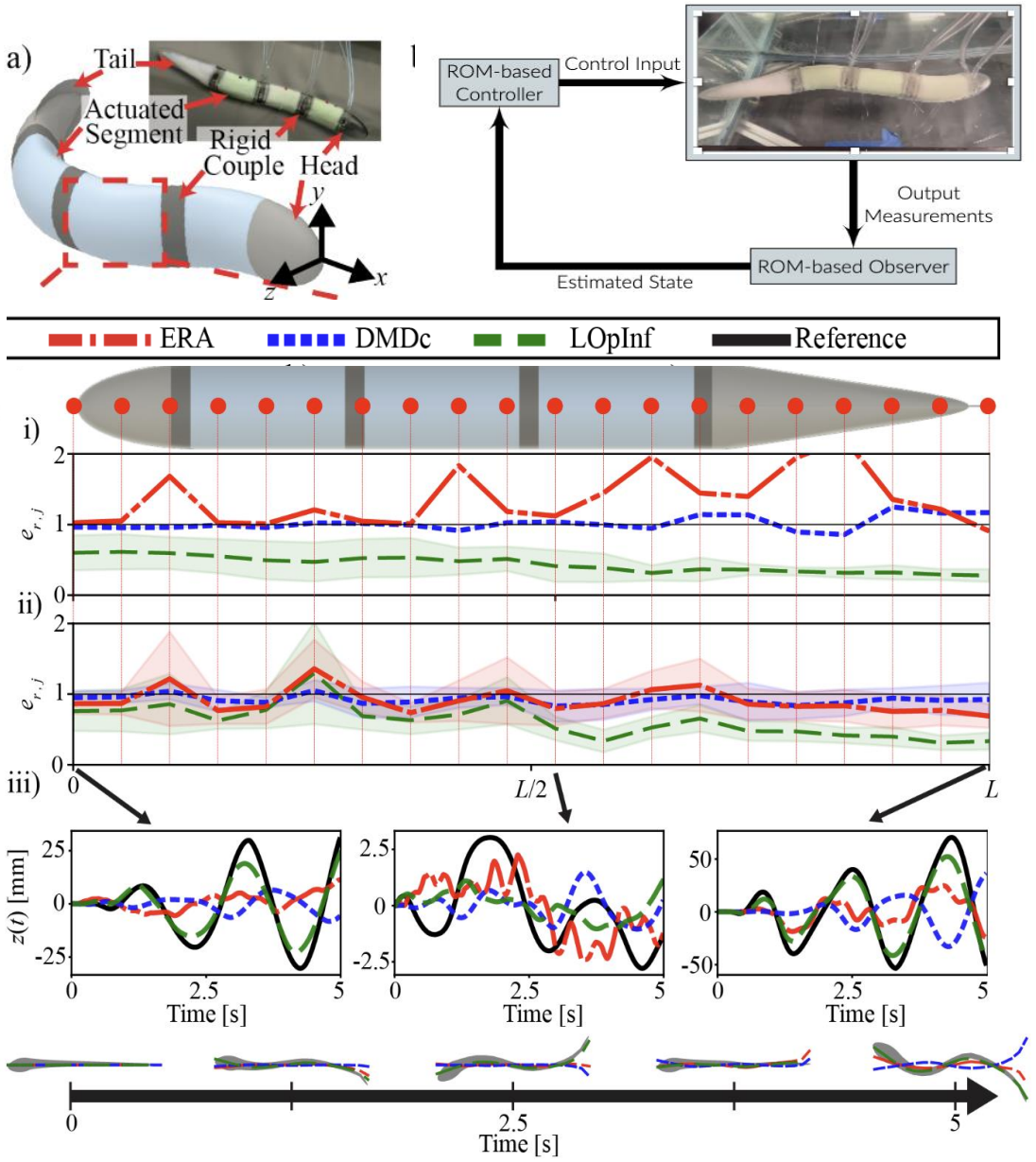
Learning nonlinear Lagrangian ROMs from experimental data for backbone curve prediction



(a) Experimental data

(b) LOpInf-SpML ROM $r = 3$

Dynamic shape control of soft robots via linear Lagrangian ROMs



Goal: Model Reduction

Full Order Model (FOM)

- We have a massive, complex model.
- Simulating it is **extremely slow** and computationally expensive.

Reduced Order Model (ROM)

- **Goal:** Create a *small*, fast model that behaves identically.

Motivation: Why do we need ROMs?

- **Digital Twins:** We need models that run in **real-time**.
- **Optimization:** We need to run thousands of simulations quickly.
- **Control Design:** We need simple models to design controllers.

Black-Box Challenge

This is easy if you have the system matrices (E, A, B, C). What if the system is a physical experiment where you can only get **simulation data**?

Intrusive to Data-Driven Methods

Classic Balanced Truncation (BT)
[Moore '81]

- Computes Gramian matrices (P, $E^\mathsf{T}QE$) that describe the system's dynamics.

QuadBT [Gosea, Gugercin, Beattie '22]

- **Data-driven.** It implicitly *approximates* the Gramians using only samples of the system's response.

Problem

It's **intrusive**. It requires explicit access to the full system matrices (E, A, B, C).

New Problem

QuadBT must first construct a **massive** data matrix $\mathbb{L}$ from all the samples.

Computational Bottleneck

Computing the SVD (Singular Value Decomposition) of this massive, dense matrix $\mathbb{L}$ is the primary **memory and processing bottleneck**.

Our Solution: Tangential QuadBT

Main Idea: Tangential QuadBT

- We **avoid building** the giant block matrix $\mathbb{L}$ entirely.

How It Works

1. Apply tangential directions to the data samples *first*.
2. Construct a **single, compressed matrix** $\widehat{\mathbb{L}}$.
3. Compute the SVD of $\widehat{\mathbb{L}}$

Result

We achieve the accuracy of QuadBT at a fraction of the computation time and memory.

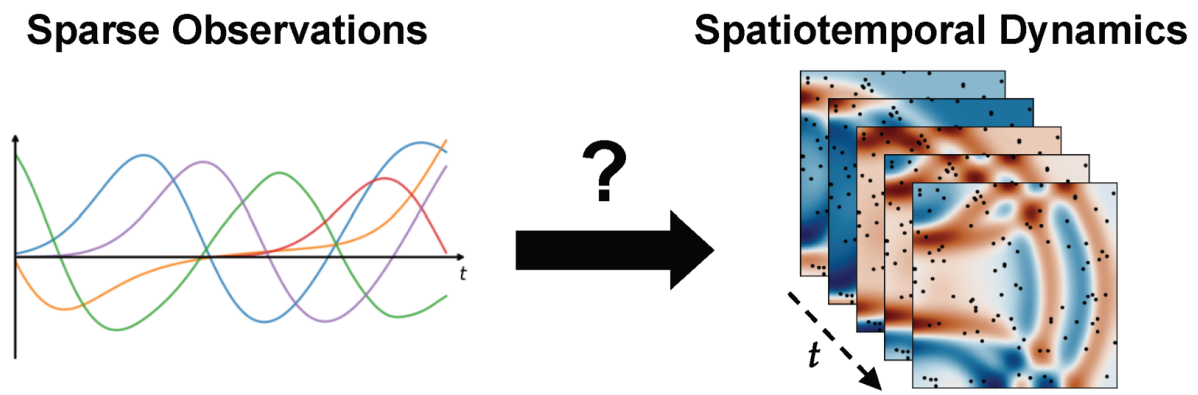


Figure 1. Conceptual illustration of the challenge: Decoding Sparse Observations to Spatiotemporal Dynamics.

ST-MLRN (SpatioTemporal Modulated Low-Rank Networks) combines a recurrent network for dynamics with a novel Implicit Neural Representation (INR) decoder for full-state reconstruction.

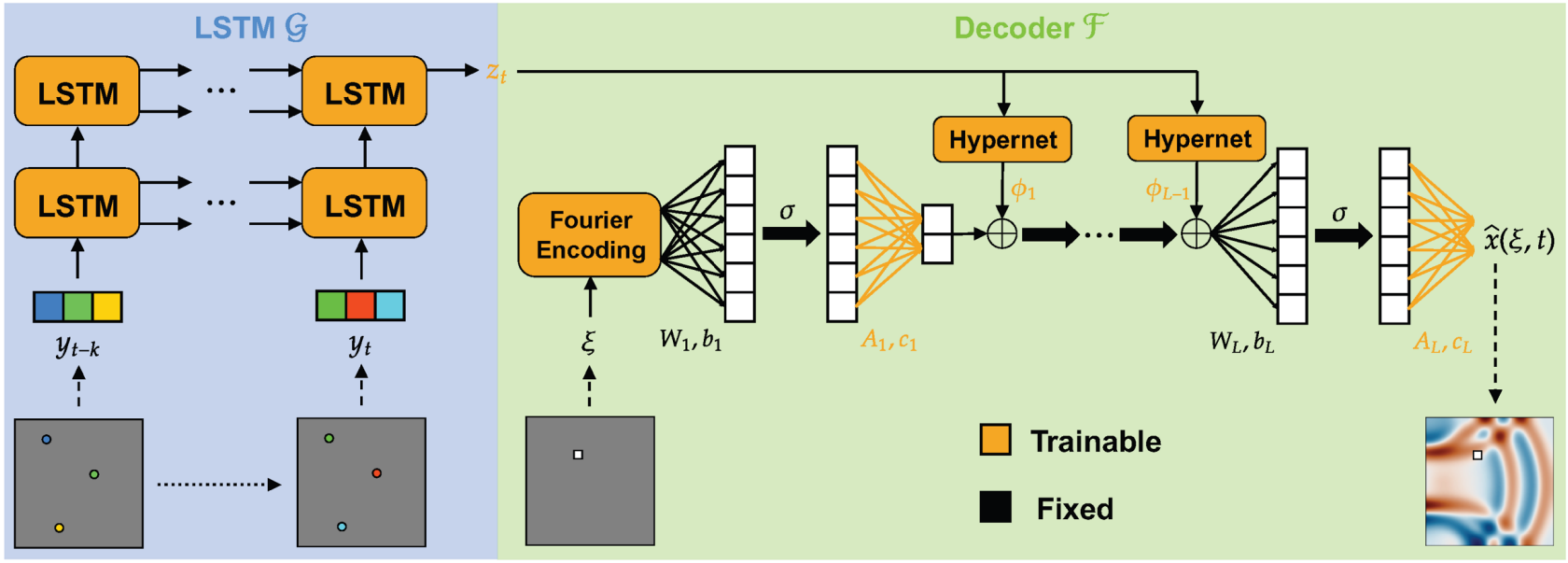


Figure 2. Summary diagram of ST-MLRN.

Georgia Tech

Table 1. Comparative performance of ST-MLRN and baseline models. ST-MLRN achieves the lowest prediction errors (highlighted in bold) across four datasets with a comparable number of parameters.

Model	dataset →	*KS*	*FlowAO*	*SWE (128×128)*	*Seismic*
SHRED	#parameters	425K	32.5M	20.1M	3.09M
	Error ϵ	6.20e-2	4.13e-2	1.75e-1	1.67e-1
SHRED-ROM	#parameters	393K	395K	917K	1.93M
	Error ϵ	6.28e-2	6.53e-2	5.13e-1	4.77e-1
ModSIREN	#parameters	448K	121K	397K	1.49M
	Error ϵ	6.34e-2	3.12e-2	4.62e-2	1.12e-1
ST-MLRN	#parameters	348K/484K	103K/149K	337K/407K	1.39M/1.80M
	Error ϵ	**3.09e-2**	**2.97e-2**	**2.78e-2**	**8.41e-2**

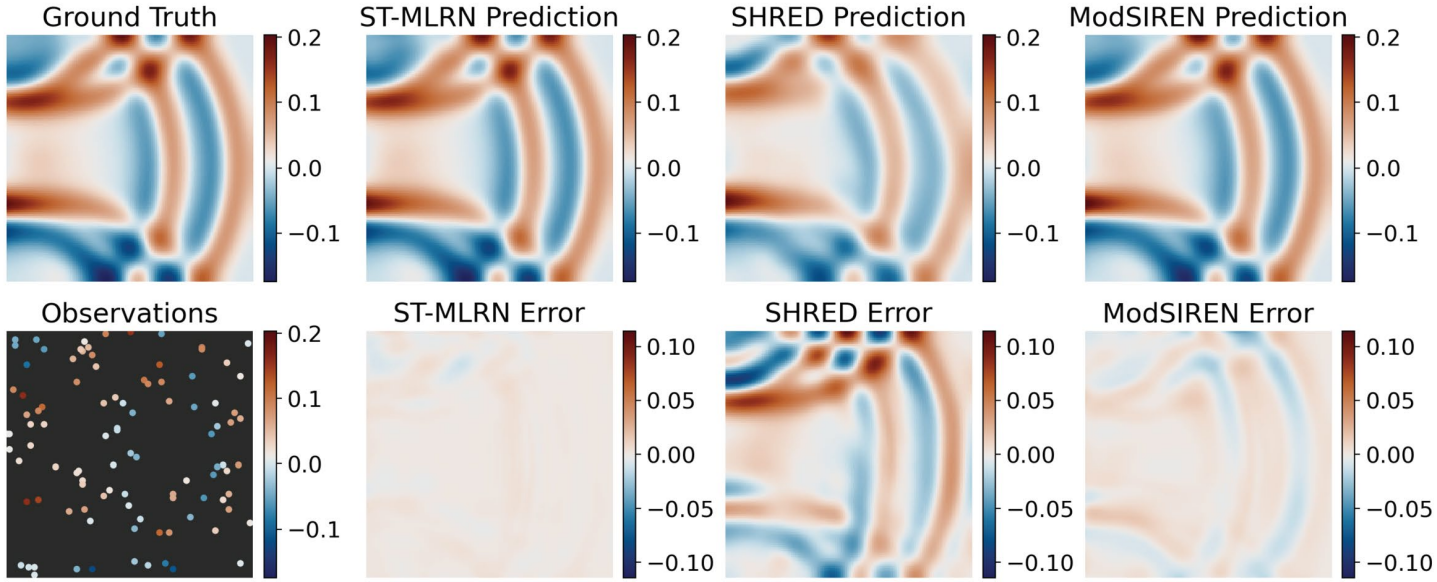


Figure 3. ST-MLRN achieves significantly lower error in reconstructing the surface elevation (η) for the Shallow Water Equations (SWE, 128×128) at the 200th time step. The visualization highlights how the ST-MLRN Error panel demonstrates superior fidelity compared to both SHRED and ModSIREN given the same observations as model input.

Georgia Tech

Table 2. ST-MLRN prediction errors on Standard and Super-Resolution tasks for the SWE case. The results confirm resolution-invariance under a constant training budget (standard scenarios: $64, 128, 256$ resolutions). Errors are also reported for super-resolution scenarios (e.g., $128 \rightarrow 256$), showing minimal accuracy loss when testing on higher resolutions than those used for training.

Variable	Standard			Super-Resolution	
	64	128	256	$64 \rightarrow 128$	$128 \rightarrow 256$
u	2.77e-2	2.90e-2	3.91e-2	9.69e-2	5.21e-2
v	2.68e-2	2.38e-2	3.55e-2	9.89e-2	4.88e-2
η	2.90e-2	2.78e-2	4.28e-2	8.14e-2	4.60e-2

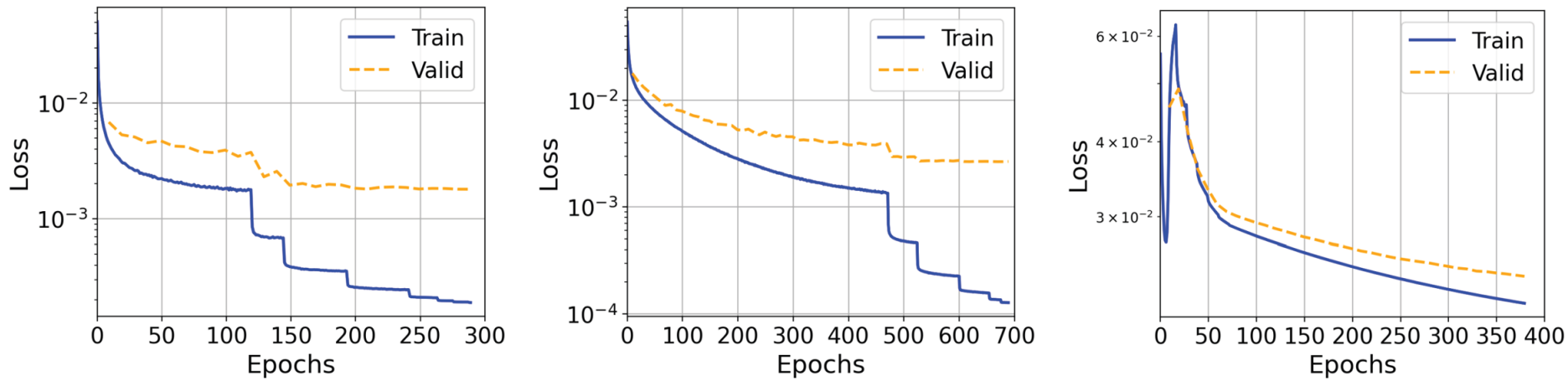


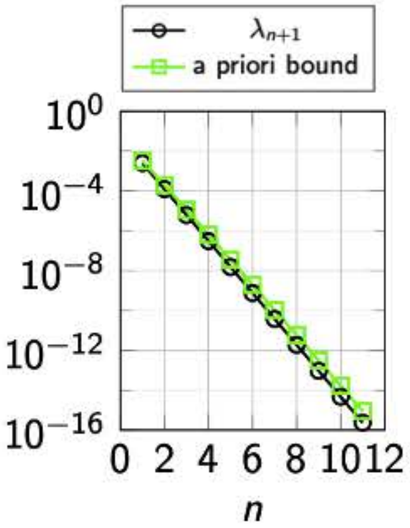
Figure 5. Comparison of training dynamics in the KS case. ST-MLRN enables stable convergence in fewer epochs (left, lr=2e-3) compared to ModSIREN, which requires hyperparameter tuning and suffers instability with larger learning rates (middle and right, lr=1e-4, 2e-4), using the same optimizer and scheduler.

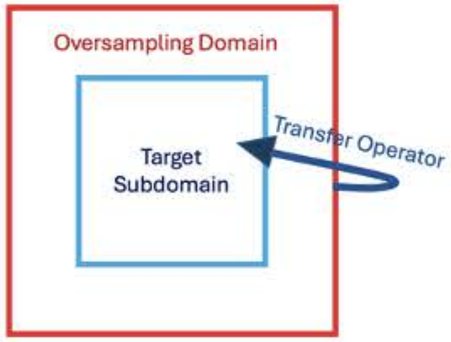
Georgia Tech

Probabilistic Error Analysis of a Randomized Proper Orthogonal Decomposition

- Kathrin Smetana[1], Tommaso Taddei[2], Marissa Whitby[1] and Zhiyu Yin[1]
- [1] Stevens Institute of Technology [2] Sapienza University of Rome

M = 80 samples





Main goals: To approximate the range of a bounded **nonlinear** parameter to solution map using POD and derive error bounds that do not rely on the dimension of the ambient space.

Current Result: We have a nonasymptotic probabilistic error bound of the covariance estimator used in POD/PCA for bounded random variables.

Current Direction: We are currently using arguments from [Reiß, Wahl '20] to derive a priori error bounds for POD/PCA for bounded random variables.

Motivation: For a restrictive subclass of sub-Gaussian random variables, which excludes some bounded random variables, [Reiß, Wahl '20] shows for a covariance operator with exponentially decaying eigenvalues that the number of samples needed to produce (quasi-)optimal results scales linearly in the dimension of the reduced space.

Applications:

- Construction of local multiscale ansatz spaces [Smetana, Taddei '23]
- PCA of the gradient of the log-likelihood function for Bayesian Inverse Problems [Zahm, Cui, Law, Spantini, Marzouk '22]

Supported by NSF Award # 2145364