



Decision-Ready Surrogates with Derivative-Informed Training

Thomas O'Leary-Roseberry + many collaborators

IMSI Digital Twins Workshop

The Ohio State University
Department of Mathematics

Main collaborators featured in this talk

UT CSEM Alumni



Omar Ghattas
Professor, UT



Lianghao Cao
Postdoc, Caltech



Josh Chen
Research Scientist, CSU



Dingcheng Luo
Junior faculty, QUT

MIT Collaborators



Youssef Marzouk
Professor, MIT



Michael Brennan
Solea Energy

Current UT PhD Students



Xindi Gong
CSEM PhD student



Joseph Kirchhoff
ME PhD student



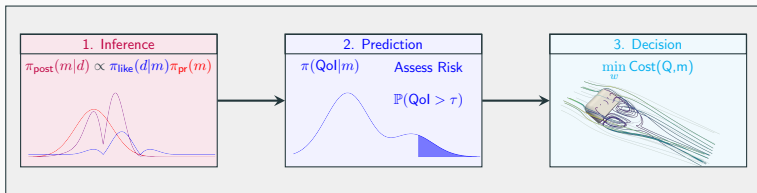
Graham Pash
CSEM PhD student



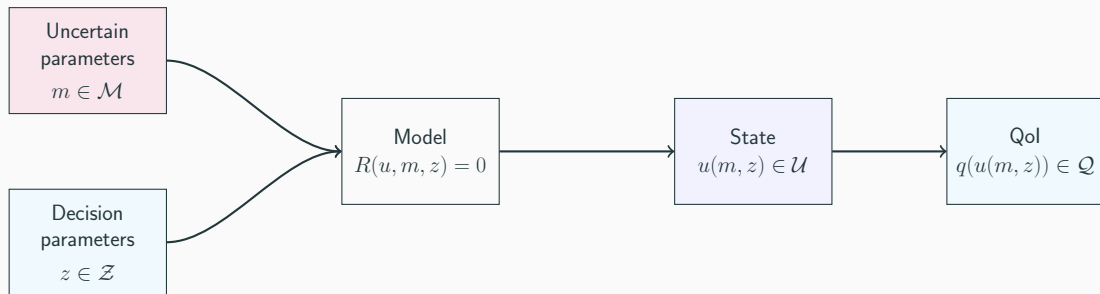
Boyuan (John) Yao
CSEM PhD student

1. Motivation: Statistical decision making for complex physical systems
2. (Derivative-Informed) Neural Operators Theory and Algorithms
3. Numerical Examples

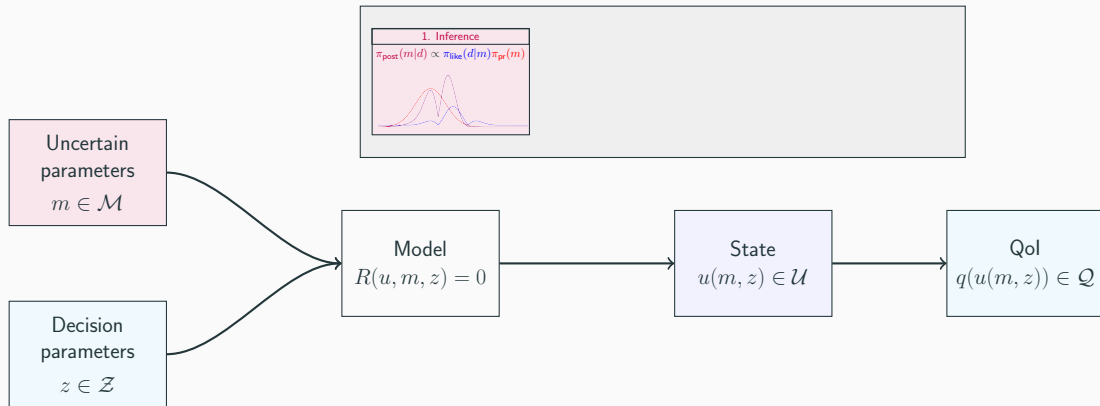
Motivation: Statistical Decision Making for Complex Systems



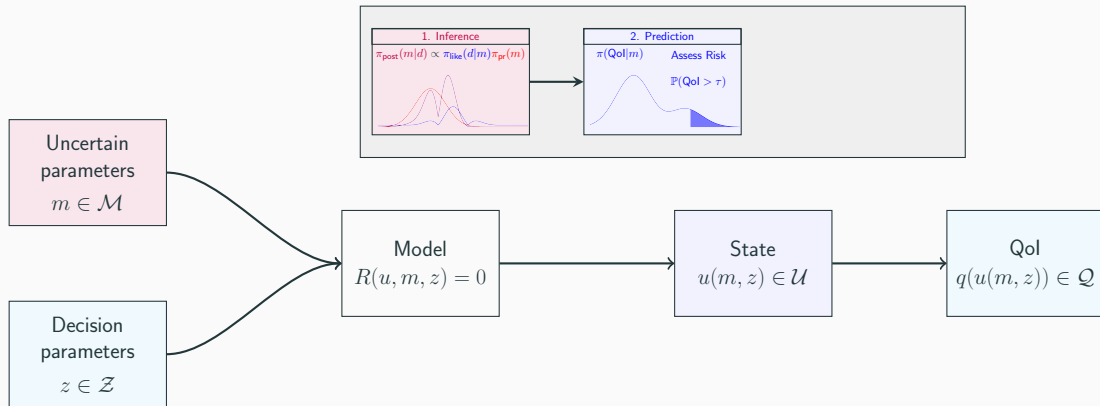
Statistical Decision Making for Complex Systems



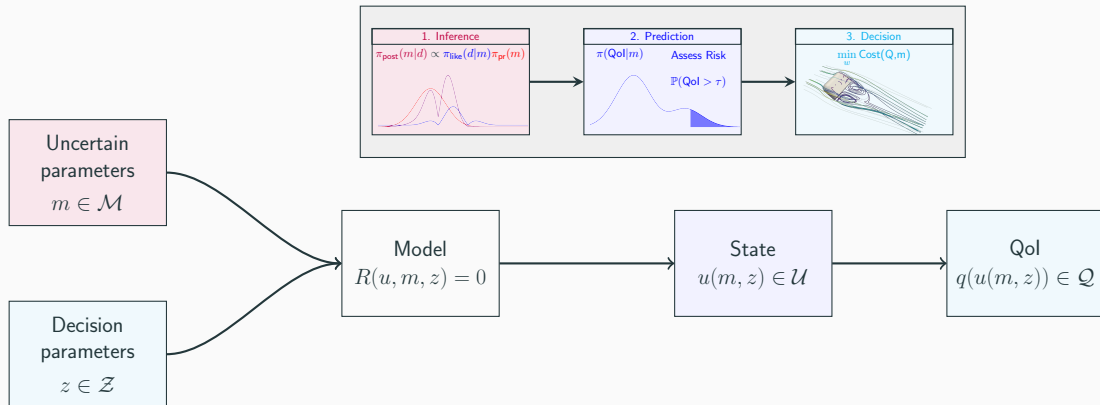
Statistical Decision Making for Complex Systems



Statistical Decision Making for Complex Systems

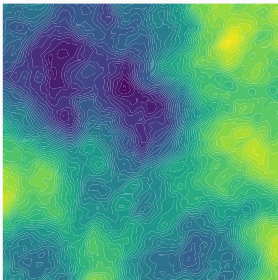


Statistical Decision Making for Complex Systems

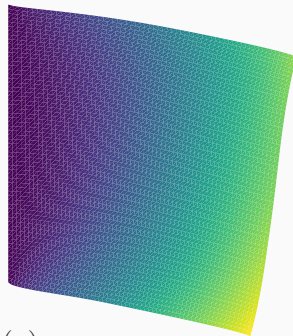


Example I: Deformation of Elastic Body

Material property m



Solution $u(m, z)$

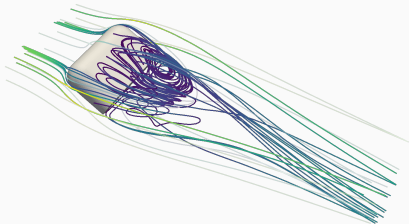


$$\nabla \cdot \sigma(u, m) + f = 0$$

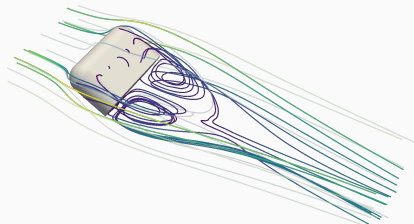
$$\sigma(u, m) \cdot n = t(z)$$

Traction boundary condition $t(z)$

Example II: Optimization of Fluid Flow



Uncontrolled



Controlled

m : Random inflow velocity

z : Suction boundary condition

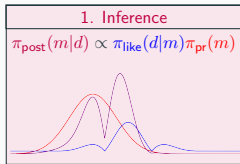
$u(m, z)$: Velocity & pressure fields

$$\nabla \cdot v = 0$$

$$\rho(v \cdot \nabla)v - \mu \nabla^2 v + \nabla p = f$$

$$+\text{Inflow}(m), \quad \text{Control}(z) \quad \text{BCs}$$

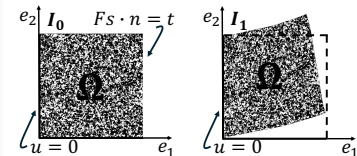
Model Calibration via Inverse Problems



Given real world observational data $\mathbf{d}$, find model parameters m , so that the model is predictive.

$$\min_m \frac{1}{2} \|Bu(m) - \mathbf{d}\|^2 + \mathcal{P}(m)$$

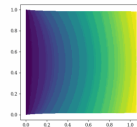
Given images of deformed elastic solids, can we determine their material properties?¹



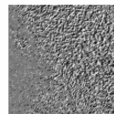
Iteration: 0



Displacement



$I_0(x) - I_1(u(x))$

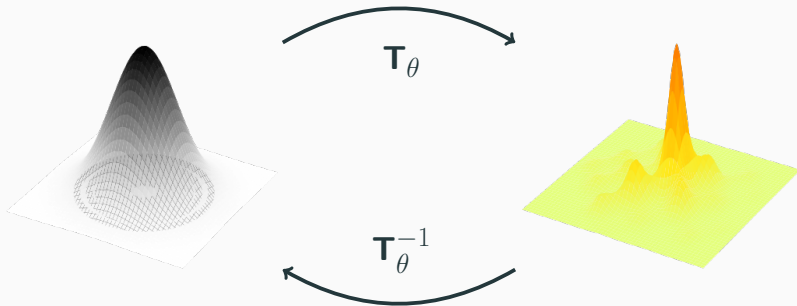


¹J. Kirchhoff, D. Luo, T. O'Leary-Roseberry, *et al.*, "Inference of Heterogeneous Material Properties via Infinite-Dimensional Integrated DIC,"

Transport Map Variational (Bayesian) Inference

Learn an invertible map $\mathbf{T} \in \mathcal{T}$ such that if $z \sim \pi_{\text{prior}}$ then $x = \mathbf{T}(z) \sim \pi_{\text{post}}$

$$\min_{\mathbf{T} \in \mathcal{T}} \mathcal{D}_{\text{KL}}(\mathbf{T}_{\#}\pi_{\text{prior}} || \pi_{\text{post}})$$

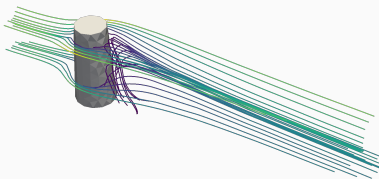
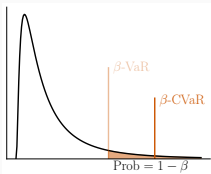


PDE-Constrained Stochastic Optimization

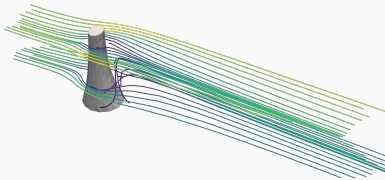
Given calibrated uncertainty $m \sim \mu$, make **risk-optimal** decisions (design/control)

$$\min_z \rho_{m \sim \mu}[Q(u(m, z))] + \mathcal{P}(z)$$

where the risk measure ρ_m specifies a risk preference e.g., Mean, **CVaR**



Initial shape



Optimal shape (credit Xindi Gong)

Statistical decision making often formulated as optimization

$$\text{E.g., } \min_m f(m) = F(u(m))$$

$$\text{or } \min_z f(z) = \rho_{m \sim \mu}[Q(m, z)]$$

Solved via

$$m_{k+1} = m_k - \alpha_k P_k \nabla f(m_k)$$

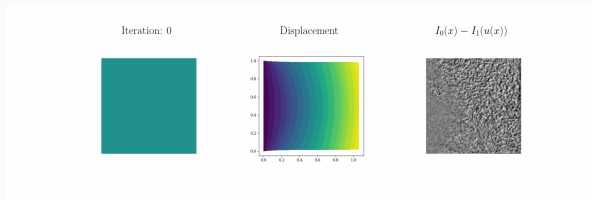
$$z_{k+1} = z_k - \alpha_k P_k \nabla f(z_k)$$

until

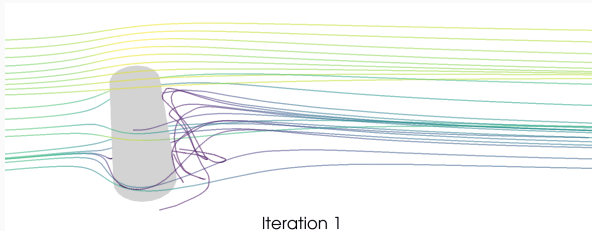
$$\nabla f(m^*) = 0$$

$$\nabla f(z^*) = 0$$

Inverse Problems



PDE-constrained optimization



Surrogates are required at scale, need gradient consistency

Algorithmic Complexity = $\mathcal{O}(\text{Sampling} \times \text{Iterations} \times \text{Model Evaluations})$

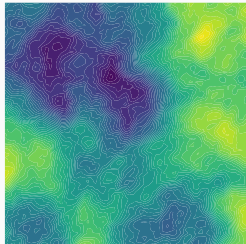
- Need surrogates to overcome the computational costs
- Additionally require that the **surrogate-approximated gradients** converge to true gradients

$$\nabla f_w \rightarrow \nabla f$$

Derivative-Informed Operator Learning

Surrogates for Implicitly Defined Maps on Function Spaces

Input function $m \in \mathcal{M}$



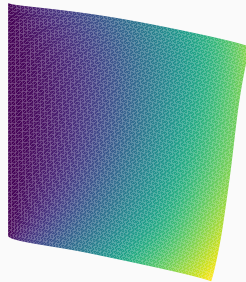
Distributed $m \sim \mu$

E.g., $\mu = \mathcal{N}(\overline{m}, \mathcal{C})$

$\mathcal{M}, \mathcal{U}$ separable Hilbert spaces.



Output function $u(m) \in \mathcal{U}$



$m \mapsto u(m)$ such that $R(u, m) = 0 \in \mathcal{U}'$

Notation for inner products and norms

- Hilbert spaces $\mathcal{M}, \mathcal{U}$ are equipped with inner products $\langle \cdot, \cdot \rangle_{\mathcal{M}}$ and $\langle \cdot, \cdot \rangle_{\mathcal{U}}$, respectively.

$$\text{E.g., } \langle w, v \rangle_{\mathcal{U}} = \int_{\Omega} w(x)v(x)dx$$

- These inner products induce norms

$$\text{E.g., } \|w\|_{\mathcal{U}} = \sqrt{\langle w, w \rangle_{\mathcal{U}}}$$

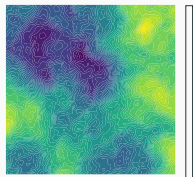
- For linear operators $A : \mathcal{M} \rightarrow \mathcal{U}$, Hilbert–Schmidt norm

$$\|A\|_{HS} = \sqrt{\sum_{i=1}^{\infty} \|Ae_i\|_{\mathcal{U}}^2}$$

where $\{e_i\}_{i=1}^{\infty}$ is an orthonormal basis for $\mathcal{M}$, i.e., $\langle e_i, e_j \rangle_{\mathcal{M}} = \delta_{ij}$

Neural Operator Learning (L^2_μ Formulation)

Input function $m \in \mathcal{M}$



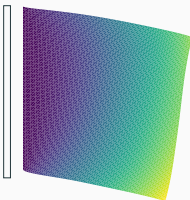
Distributed $m \sim \mu$

Neural Operator

$$u_w : \mathcal{M} \rightarrow \mathcal{U}$$

L^2_μ training formulation

Output function $u(m) \in \mathcal{U}$



$$\min_w \mathbb{E}_{m \sim \mu} [\|u(m) - u_w(m)\|_{\mathcal{U}}^2]$$

- Reduced basis architectures (e.g., PCANet²)

- Fourier Neural Operator³

- DeepONets⁴

²J. S. Hesthaven and S. Ubbiali, “Non-intrusive reduced order modeling of nonlinear problems using neural networks,” *Journal of Computational Physics*, vol. 363, pp. 55–78, Jun. 2018. DOI: 10.1016/j.jcp.2018.02.037K. Bhattacharya, B. Hosseini, N. B. Kovachki, et al., “Model reduction and neural networks for parametric PDEs,” *SMAI Journal of Computational Mathematics, Volume 7*, 2021

³Z. Li, N. Kovachki, K. Azizzadenesheli, et al., “Fourier neural operator for parametric partial differential equations,” *arXiv preprint, arXiv.2010.08895*, 2021. DOI: 10.48550/arXiv.2010.08895. eprint: 2010.08895

⁴L. Lu, P. Jin, G. Pang, et al., “DeepONet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators,” *Nature Machine Intelligence*, 2021

Motivation: Optimization for input parameter.

Consider a simple optimization problem for the input parameter m .

$$\min_m f(m) = F(u(m)), \text{ e.g., } \frac{1}{2} \|u(m) - u^{\text{target}}\|_{\mathcal{U}}^2$$

Solved by gradient methods:

$$\text{Find } m^\star \text{ such that } \nabla f(m^\star) = 0$$

Notation: D : infinite-dimensional (Fréchet) derivatives, ∇ : gradients (via Riesz R.T.).

Approximate solution using a surrogate

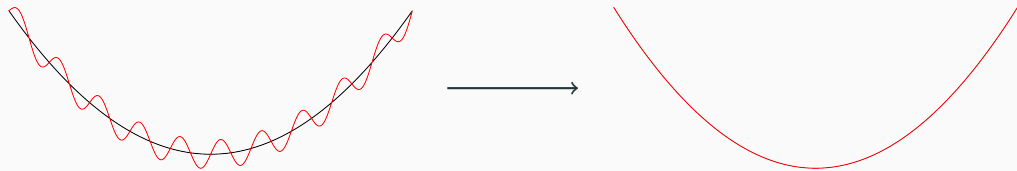
Consider the surrogate approximated optimization problem

$$\min_m f_w(m) = F(u_w(m)), \text{ e.g., } \frac{1}{2} \|u_w(m) - u^{\text{target}}\|_{\mathcal{U}}^2$$

How close is the surrogate approximated solution to the truth?

Approximate solution using a surrogate

- Neural network approximation invariably introduces nonconvexity into f_w
- NN approximation theory often establishes uniform convergence on compact sets.
- This guarantees convergence of minimization problems asymptotically (epiconvergence)



There are no practical guarantees pre-asymptotically!

Approximate solution using a surrogate

Consider the surrogate approximated optimization problem

$$\min_m f_w(m) = F(u_w(m)), \text{ e.g., } \frac{1}{2} \|u_w(m) - u^{\text{target}}\|_{\mathcal{U}}^2$$

How close is the surrogate approximated solution to the truth?

Bound of gradient error (informal)

Assume $Du(m), Du_w(m)$ bounded in e.g., Hilbert–Schmidt norm, locally bounded F .

$$\|\nabla f(m) - \nabla f_w(m)\|_{\mathcal{M}}^2 \leq C_1 \left(\|u(m) - u_w(m)\|_{\mathcal{U}}^2 + \|Du(m) - Du_w(m)\|_{HS(\mathcal{M}, \mathcal{U})}^2 \right)$$

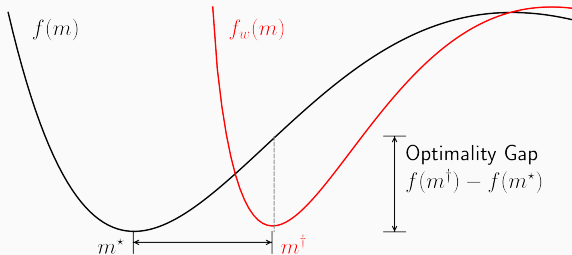
Optimization using a surrogate leads to systematic error

Find $\nabla f_w(m^\dagger) = 0$ instead of find $\nabla f(m^\star) = 0$

Bound of optimality gap

Suppose $f(m)$ is strongly convex in a ball containing $m^\star$ and $m^\dagger$, then we have

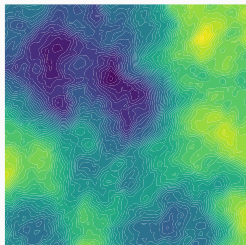
$$f(m^\dagger) - f(m^\star) \leq C_2 \left(\|u(m^\dagger) - u_w(m^\dagger)\|_{\mathcal{U}}^2 + \|Du(m^\dagger) - Du_w(m^\dagger)\|_{HS(\mathcal{M}, \mathcal{U})}^2 \right)$$



**Control surrogate Jacobian to
control optimization error**

Derivative-Informed Neural Operators (H_μ^1 formulation⁵)

Input function $m \in \mathcal{M}$

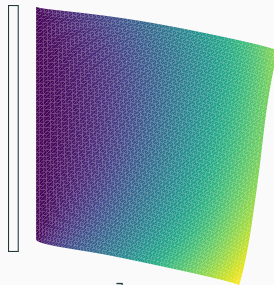


Derivative-Informed
Neural Operator

$$u_w : \mathcal{M} \rightarrow \mathcal{U}$$

H_μ^1 training formulation

Output function $u(m) \in \mathcal{U}$



$$\min_w \mathbb{E}_{m \sim \mu} \left[\|u(m) - u_w(m)\|_{\mathcal{U}}^2 + \|Du(m) - Du_w(m)\|_{HS(\mathcal{M}, \mathcal{U})}^2 \right]$$

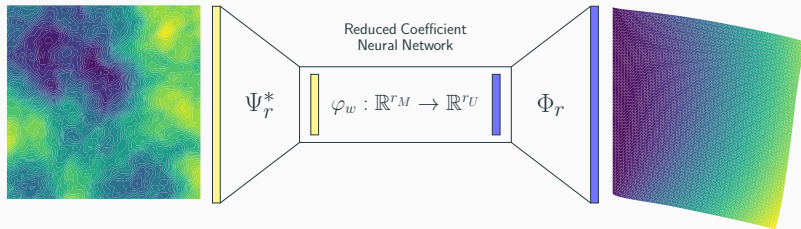
Are NOs universal approximations in H_μ^1 ?
How do we make it work in practice?

⁵T. O'Leary-Roseberry, P. Chen, U. Villa, et al., "Derivative-informed neural operator: An efficient framework for high-dimensional parametric derivative learning," *Journal of Computational Physics*, vol. 496, p. 112555, 2024

Reduced Basis Architecture H_μ^1 UA

Input function $m \in \mathcal{M}$

Output function $u(m) \in \mathcal{U}$



- Empirical approximation of reduced bases
- E.g., PCA, derivative-informed subspaces etc.
- Restrictions on the activation function (smooth, ReLU-like activation functions)

Theorem 10 in Luo et al, informal

For any $\epsilon > 0$, $r, N \in \mathbb{N}$ there exists an RBNO $u_w = \Phi_r \circ \varphi_w \circ \Psi_r^*$ with d_L layers simlult. satisfying

$$\begin{aligned} \mathbb{E}_N \left[\|u - u_w\|_{L_\mu^2}^2 \right] &\leq \underbrace{\epsilon_w}_{NN} + \underbrace{K_{\mathcal{M}}^0 \left(\sum_{j \geq r} \lambda_j^{\text{in}} + \mathcal{E}_{\mathcal{M}}(N, r) \right)}_{\text{input basis truncation + sampling}} + \underbrace{K_{\mathcal{U}}^0 \left(\sum_{j \geq r} \lambda_j^{\text{out}} + \mathcal{E}_{\mathcal{U}}(N, r) \right)}_{\text{output basis truncation + sampling}} + \underbrace{\mathcal{E}_u(N)}_{\text{sampling}} \\ \mathbb{E}_N \left[\|u - u_w\|_{H_\mu^1}^2 \right] &\leq \underbrace{\epsilon_w}_{NN} + \underbrace{K_{\mathcal{M}}^1 \left(\sum_{j \geq r} \lambda_j^{\text{in}} + \mathcal{E}_{\mathcal{M}}(N, r) \right)}_{\text{input basis truncation + sampling}} + \underbrace{K_{\mathcal{U}}^1 \left(\sum_{j \geq r} \lambda_j^{\text{out}} + \mathcal{E}_{\mathcal{U}}(N, r) \right)}_{\text{output basis truncation + sampling}} \end{aligned}$$

consequently, we can pick r, N, w such that

$$\|u - u_w\|_{H_\mu^1}^2 = \mathbb{E}_{m \sim \mu} \left[\|u(m) - u_w(m)\|_{\mathcal{U}}^2 + \|Du(m) - Du_w(m)\|_{HS(\mathcal{M}, \mathcal{U})}^2 \right] < \epsilon$$

⁶D. Luo, T. O'Leary-Roseberry, P. Chen, et al., "Dimension reduction for derivative-informed operator learning: An analysis of approximation

Efficient Training Algorithms

Derivative of PDE solution is a linear operator, given by **implicit function theorem**

$$\mathcal{L}(\mathcal{M}, \mathcal{U}) \ni Du(m) = - \underbrace{\left[\frac{\partial R(u, m)}{\partial u} \right]^{-1}}_{A^{-1} \text{ for linear PDEs}} \frac{\partial R(u, m)}{\partial m}$$

Efficient Matrix-Free Approximation of $Du(m)$

- Can approximate $Du(m)$ via its application (or adjoint) on functions.
- Can be done on random vectors, or on targeted subspaces
- Each application further amortizes linear algebra (e.g., $A^{-1}, \partial_u R^{-1}$)
- Derivative data costs **can be marginal** relative to forward solves.

Orthonormal bases (ONBs)

For a Hilbert space, $\mathcal{M}$, ONB $\{\psi_i\}_{i=1}^{\infty}$ spans $\mathcal{M}$, and

$$\langle \psi_i, \psi_j \rangle_{\mathcal{M}} = \delta_{ij}$$

Generalizes canonical bases e_i for $\mathbb{R}^d$ to function spaces

Computational difficulty with derivative training

$$\|Du(m) - Du_w(m)\|_{HS}^2 = \begin{cases} \sum_{i=1}^{\infty} \|(Du(m) - Du_w(m))\psi_i\|_{\mathcal{U}}^2, & \{\psi_i\}_{i=1}^{\infty} \text{ an ONB for } \mathcal{M} \\ \sum_{i=1}^{\infty} \|(Du(m)^* - Du_w(m)^*)\phi_i\|_{\mathcal{M}}^2, & \{\phi_i\}_{i=1}^{\infty} \text{ an ONB for } \mathcal{U} \end{cases}$$

is infeasible to evaluate.

Solution: subspace training

We can overcome this difficulty by approximating the Jacobian norm in subspaces. If Φ_r and Ψ_r are chosen wisely then:

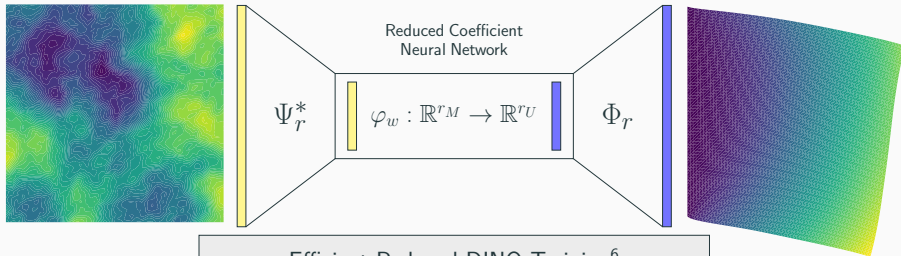
$$\begin{aligned}\|Du(m) - Du_w(m)\|_{HS}^2 &\approx \sum_{i \leq r_U, j \leq r_M} \langle \phi_i, (Du(m) - Du_w(m)) \psi_j \rangle_{\mathcal{U}}^2 \\ &= \|\Phi_i^*(Du(m) - Du_w(m))\Psi_r\|_F^2\end{aligned}$$

- This applies to any neural operator.
- Jacobian of neural network can be computed via forward or reverse mode AD.

Reduced Basis Architectures Lead to Efficient DINO Formulation

Input function $m \in \mathcal{M}$

Output function $u(m) \in \mathcal{U}$



Efficient Reduced DINO Training⁶

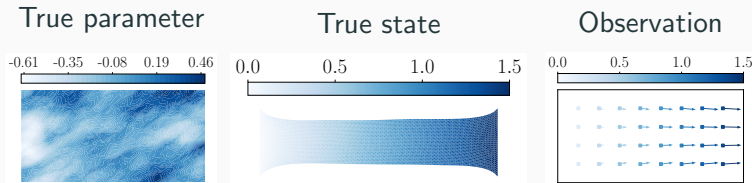
$$\mathbf{g}(m) = \Phi_r^* \circ u(m)$$

$$\nabla \mathbf{g}(m) = \Phi_r^* \circ Du(m) \circ \Psi_r$$

$$\min_w \mathbb{E}_{m \sim \mu} [\|\varphi_w - \mathbf{g}\|_2^2 + \|\nabla \varphi_w - \nabla \mathbf{g}\|_F^2]$$

⁶T. O'Leary-Roseberry, P. Chen, U. Villa, et al., "Derivative-informed neural operator: An efficient framework for high-dimensional parametric derivative learning," *Journal of Computational Physics*, vol. 496, p. 112555, 2024

Hyperelasticity Parameter to Observable Problem



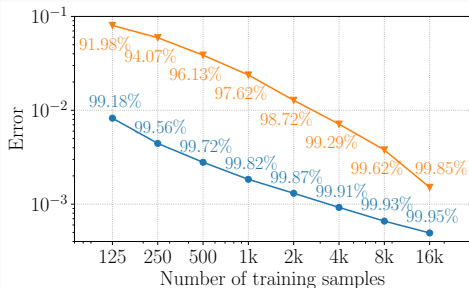
Cost-accuracy comparison of construction

- L^2_μ and H^1_μ training
- Reduced Basis architecture.

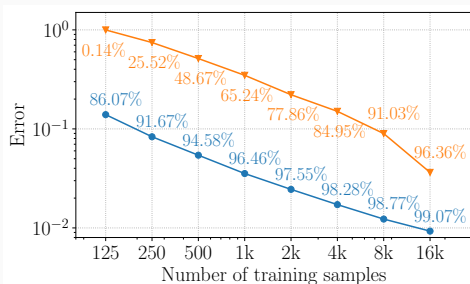
Comparison of generalization errors for L_μ^2 , H_μ^1 training

—●— DINO (H_μ^1 training) —▼— NO (L_μ^2 training)

Observable generalization error



Jacobian generalization error

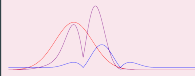


DINO training improves generalization accuracies per-unit compute.

Bayesian Inference with DINO's

1. Inference

$$\pi_{\text{post}}(m|d) \propto \pi_{\text{like}}(d|m)\pi_{\text{pr}}(m)$$



Bayesian Inverse Problems: Statistically Infer m from Observations

- Observations

$$d = Bu(m) + \epsilon = q(m) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \Gamma_{\text{noise}})$$

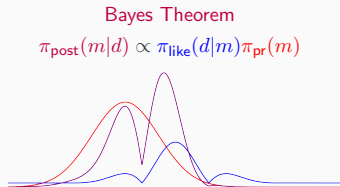
B is the observation operator, ϵ is Gaussian additive noise, q is the parameter-to-observable map

- Gaussian Prior $\mu = \mathcal{N}(\bar{m}, \mathcal{C}_{\text{prior}})$

Bayes Theorem

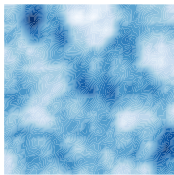
$$d\mu_{\text{post}}(m|d) \propto \underbrace{\exp(-\Phi(m; d))}_{\text{the likelihood}} d\mu_{\text{prior}}(m)$$

$$\text{where } \Phi(m; d) \propto \frac{1}{2} \|q(m) - d\|_{\Gamma_{\text{noise}}^{-1}}^2$$

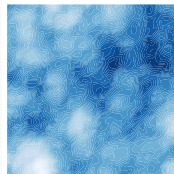


Bayesian Inference: Move Prior Samples to Obey Posterior

Prior Sample



Posterior Sample



Methods

- Markov Chain Monte Carlo
- Variational inference

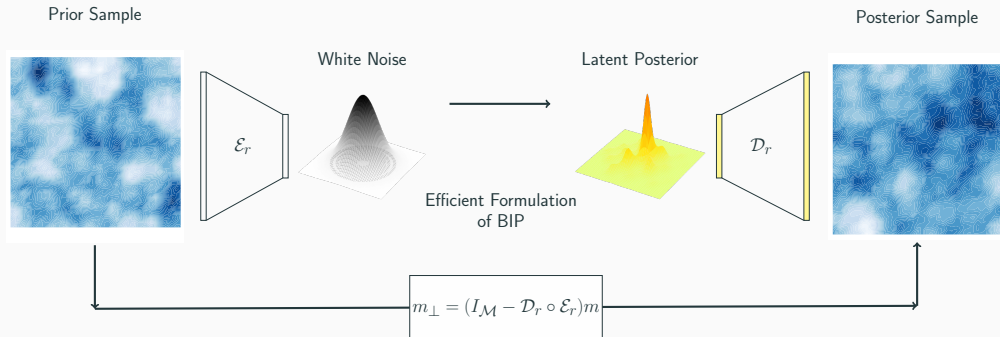
Difficulties

- High dimensionality
- Likelihood evaluations (PDE solves)

Formally intractable for high-dimensional expensive models.

Tractable Bayesian Inference via Subspace Decomposition

Subspace decomposition leads to efficient formulation of Bayesian inverse problems



When does it lead to a good approximation of the original problem?

Bayes: $d\mu_{\text{post}}(m|d) \propto \exp(-\Phi(m; d)) d\mu_{\text{prior}}(m)$

- Approximate likelihood potential $\Phi : \mathcal{M} \rightarrow \mathbb{R}$ by ridge function

$$\widehat{\Phi} : \text{Im}(\mathcal{P}) \rightarrow \mathbb{R}, \quad \mathcal{P} = \mathcal{D}_r \circ \mathcal{E}_r, \quad \widehat{\Phi} \circ \mathcal{D}_r : \mathbb{R}^r \rightarrow \mathbb{R}$$

- Defines approximate posterior $\widehat{\mu}_{\text{post}}$

Subspace Logarithmic Sobolev Inequality⁷

Using the derivative-informed subspace, there exists a ridge function potential such that

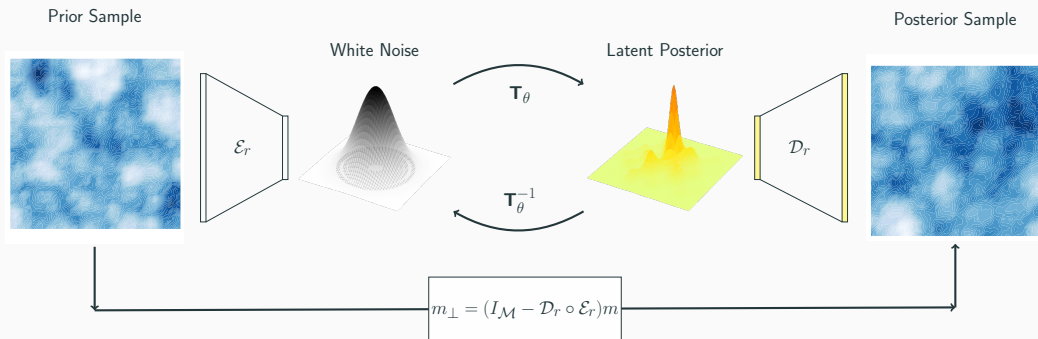
$$\mathbb{E}_d \left[\mathcal{D}_{\text{KL}}(\mu_{\text{post}} || \widehat{\mu}_{\text{post}}) \right] \leq \frac{1}{2} \sum_{i \geq r} \lambda_i^{\text{AS}}$$

⁷T. Cui and O. Zahm, “Data-free likelihood-informed dimension reduction of bayesian inverse problems,” *Inverse Problems*, vol. 37, no. 4,

Transport Map Variational Inference with Lazy Maps⁸

Learn an invertible map $\mathbf{T} \in \mathcal{T}$ such that if $z \sim \pi_{\text{prior}}$ then $x = \mathbf{T}(z) \sim \pi_{\text{post}}$

$$\min_{\mathbf{T} \in \mathcal{T}} \mathcal{D}_{\text{KL}}(\mathbf{T}_{\#} \pi_{\text{prior}} || \pi_{\text{post}})$$



Full map is then $T_\theta = \mathcal{D}_r \circ \mathbf{T}_\theta \circ \mathcal{E}_r + (I_{\mathcal{M}} - \mathcal{P})$

⁸M. Brennan, D. Bigoni, O. Zahm, et al., "Greedy inference with structure-exploiting lazy maps," *Advances in Neural Information Processing*

Transport Map Construction with PDE is expensive

$\mathbf{T}_\theta$, parametrized by $\theta \in \mathbb{R}^{d_\theta}$ constructed via expensive stochastic optimization

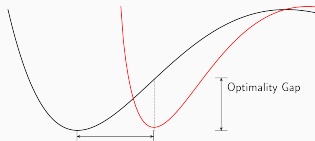
$$\min_{\theta} \mathcal{L}^d(\theta) = \mathcal{D}_{\text{KL}}(T_\theta \mu || \mu_{\text{post}}^d)$$

- Intractable when using PDE.
- Surrogate necessary but leads to errors.

Transport Map Optimization Bounds (Informal)⁹

$$\mathbb{E}_d \left[\left\| \nabla_{\theta} \mathcal{L}^d(\theta) - \nabla_{\theta} \widehat{\mathcal{L}}^d(\theta) \right\| \right] \leq C_1 \mathbb{E}_{m \sim \mu} \left[\|f_w - g\|_2^2 + \|\nabla f_w - \nabla g\|_F^2 \right]^{\frac{1}{2}}$$

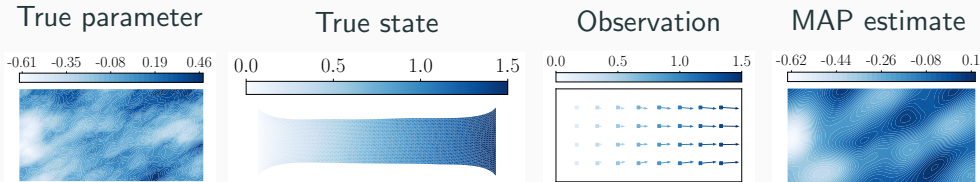
$$\mathbb{E}_d \left[\sqrt{\mathcal{L}^d(\theta^\dagger) - \mathcal{L}^d(\theta^*)} \right] \leq C_2 \mathbb{E}_{m \sim \mu} \left[\|f_w - g\|_2^2 + \|\nabla f_w - \nabla g\|_F^2 \right]^{\frac{1}{2}}$$



⁹L. Cao, J. Chen, M. Brennan, *et al.*, “LazyDINO: Fast, Scalable, and Efficiently Amortized Bayesian Inversion via Structure-Exploiting and

Surrogate-Driven Measure Transport,” *To be submitted, Journal of Machine Learning Research, arXiv preprint arXiv:2411.12726*, 2024

Hyperelastic material inverse problem with Gaussian random field prior



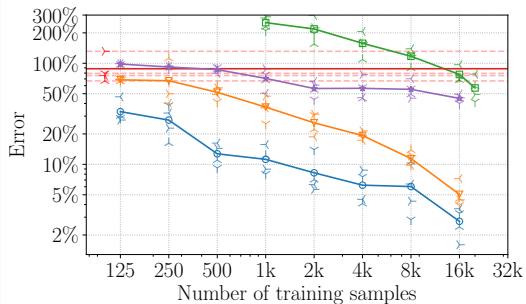
Comparison of methods on cost-accuracy basis

We compare LazyDINOs with

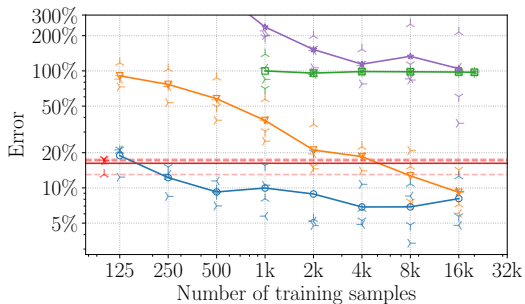
- LazyNO (using L^2_{μ} NO)
- LazyMaps (using PDE)
- neural posterior estimation (NPE)
- Laplace approximation

LazyDINO produces superior mean and covariance estimates

Posterior Mean Error



Posterior Covariance Error



Cost of PtO *amortized* over 4 BIPs

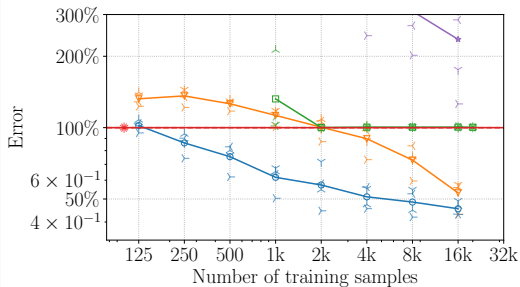
- LazyDINO (Ours)
- ▽— LazyN0
- ★— NPE

Cost of PtO *repeated* over 4 BIPs

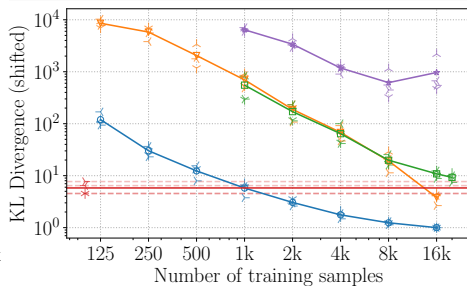
- LazyMaps
- Laplace approximation

LazyDINO produces superior skewness and KL divergence errors

Posterior Skewness Error



Shifted forward KL divergence error



Cost of PtO *amortized* over 4 BIPs

- LazyDINO (Ours)
- ▽— LazyNO
- ▲— NPE

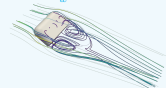
Cost of PtO *repeated* over 4 BIPs

- LazyMaps
- *— Laplace approximation

Stochastic PDE-constrained optimization with DINOs

3. Decision

$\min_w \text{Cost}(Q,m)$



PDE-constrained optimization under uncertainty

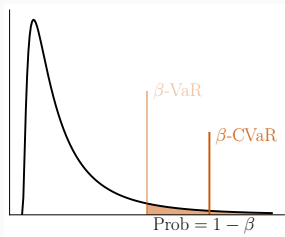
Once uncertainty is quantified in complex systems, we often want to make **risk-optimal decisions** with respect to (finite-dimensional) optimization variable z .

$$\min_z F(z) = \rho_{m \sim \nu}[Q(u(m, z))] \text{ s.t. } R(u, m, z) = 0$$

- Q is an **optimization objective** (e.g., drag)
- ρ is a **risk measure** on Q

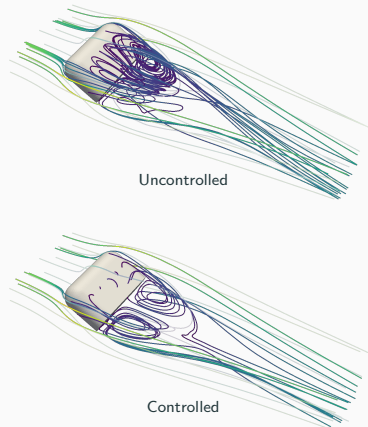
E.g., Mean: $\mathbb{E}_{\nu_m}[Q]$

$\text{CVaR}_\beta[Q]$

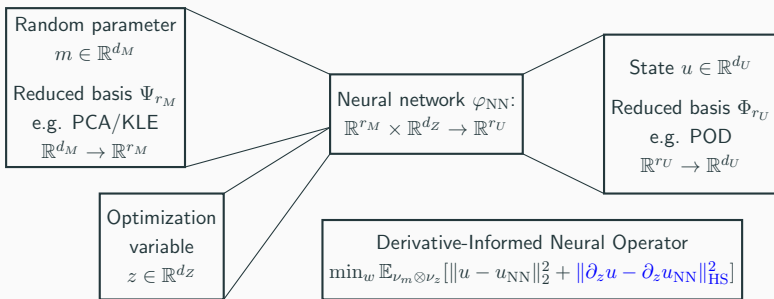


- Such problems are solved using **gradient-based** optimization methods

E.g., $z_{k+1} = z_k - \alpha_k \nabla_z F(z)$ until convergence



$$\text{MR-DINO} : u_{\text{NN}}(m, z) = \Phi_{r_U} \varphi_w(\Psi_{r_M}^T m, z) + b$$

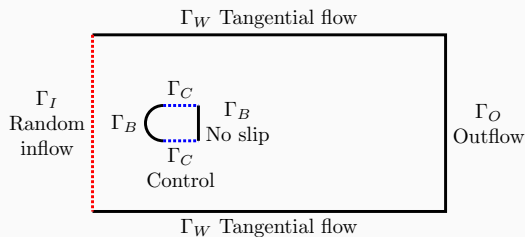


MR-NO : Same architecture, but without derivative training

¹⁰ D. Luo, T. O'Leary-Roseberry, P. Chen, *et al.*, "Efficient PDE-constrained optimization under high-dimensional uncertainty using derivative-informed neural operators," *SIAM Journal on Scientific Computing*, arXiv preprint arXiv:2305.20053, 2025

Numerical example: flow control around bluff body

$$\begin{aligned}
 -\nu \Delta \mathbf{u} + \nabla p + (\mathbf{u} \cdot \nabla) \mathbf{u} &= 0 & x \in \Omega \\
 \nabla \cdot \mathbf{u} &= 0 & x \in \Omega \\
 \mathbf{u} - \mathbf{u}_\infty &= 0 & x \in \Gamma_{in} \\
 \mathbf{u} &= 0 & x \in \Gamma_B \setminus \Gamma_C \\
 \mathbf{u} - \phi(z) \mathbf{n} &= 0 & x \in \Gamma_C \\
 \mathbf{u} \cdot \mathbf{n} = 0, \quad T(u, p) \mathbf{n} \cdot \mathbf{t} &= 0 & x \in \Gamma_W \\
 T(\mathbf{u}, p) \mathbf{n} &= 0 & x \in \Gamma_O
 \end{aligned}$$



Uncertain inlet flow profile: $\mathbf{u}_\infty = e^m \mathbf{e}_1$ where $m \sim \mathcal{N}(\bar{m}, \mathcal{C})$

Control flow profile over bluff body edge, parameterized by B-splines

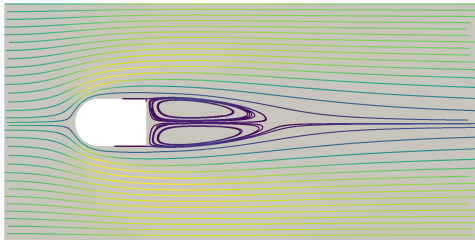
$$\phi(z)(x) = \sum_{i=1}^N z_i \phi_i(x) \mathbf{n}$$

Viscous dissipation obj. $:= 2\nu \int_{\Omega} \text{sym}(\nabla \mathbf{u}) : \text{sym}(\nabla \mathbf{u}) \, dx$, **Tracking obj.** $:= \int_{D_o} \|\mathbf{u} - \mathbf{u}_{\text{target}}\|^2 \, dx$

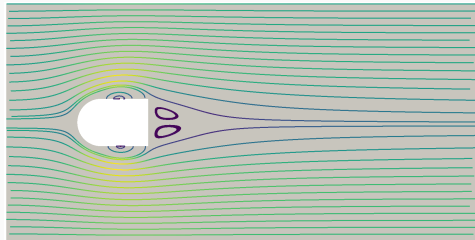
Penalization: L^2 norm of the controlled velocity $\mathcal{P}(z) := \alpha \int_{\Gamma_C} |\phi(z)|^2 \, ds$

Optimal flow control

Viscous dissipation minimization: $\min_z \text{CVaR}_\beta[Q](z) + \mathcal{P}(z)$ with $\beta = 0.95$



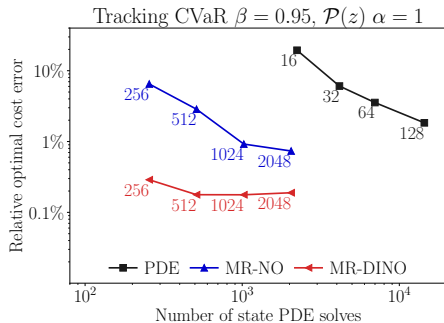
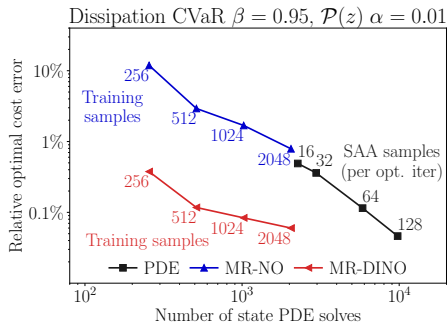
Uncontrolled



Optimal control

Flow field with no control and optimal control z_{NN}^* by MR-DINO with 256 training samples

Cost-accuracy comparison for OUU



Relative optimal cost errors (CVaR + penalty) computed at the optimal controls

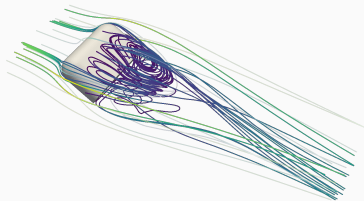
MR-DINO up to **10× more sample efficient** than SAA with PDE for the viscous dissipation objective and over **100× more sample efficient** for tracking objective

A 3D Navier–Stokes flow control problem with MR-DINO

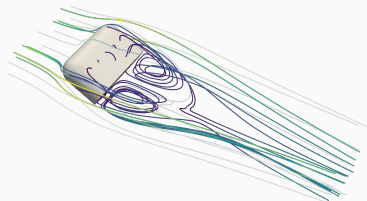
Optimization of viscous dissipation with CVaR, $\beta = 0.95$ with $L^2(\Gamma_C)$ penalty

MR-DINO trained with **448 samples**: **1.5%** L^2_μ testing error

$10^7 \times$ faster than PDE solves, each solve taking **30mins**, using **48 CPU cores**.



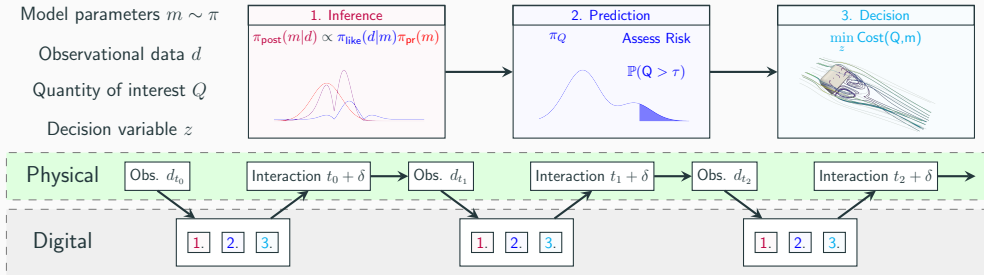
Uncontrolled



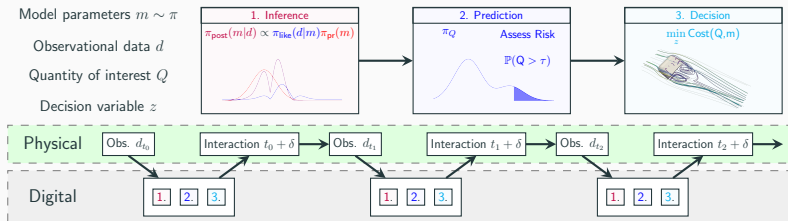
Controlled

Streamlines for a sample of the flow field with a random control (left) and an optimal control (right). The SAA optimization problem with MR-DINO is solved in **53 seconds**.

Towards Digital Twins



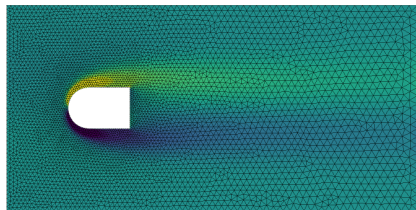
DT-Ready Surrogates



- DTs require surrogates ready to rapidly **infer**, **predict** and **optimize** in sequence.
- Must be parametrized by inference and decision parameter, and accurately capture system dynamics.

Derivative-Informed Operator Learning for Transient Systems

with Lianghao Cao, Dingcheng Luo, **Boyuan (John) Yao** and **Graham Pash**



NOs for parametrized temporal systems

$$\frac{\partial u}{\partial t} + R(u, m, z) = 0$$

Preliminary results: random inflow

PDE



H_{μ}^1 HMM



L_{μ}^2 HMM



- Reduced basis approach
- Hidden memory model (HMM)

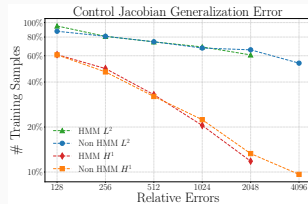
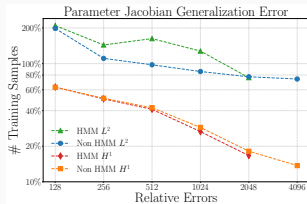
- L_{μ}^2 training
- Derivative training H_{μ}^1

Preliminary results: control problem

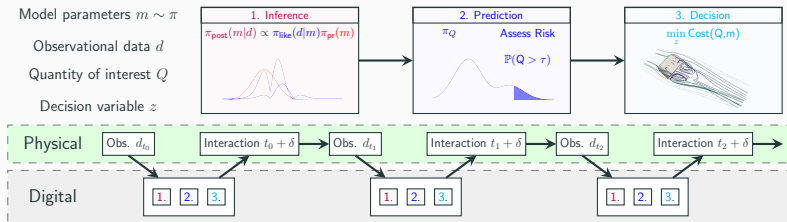
Uncontrolled



Controlled (with PDE)



Additional thoughts



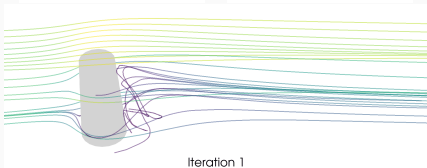
- DA / Inference task can save surrogate from divergence over a fixed window.
- Distributional shift will be a huge issue.

Thanks for your attention!

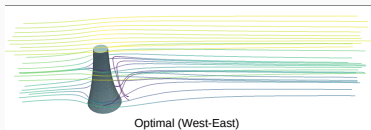
Papers



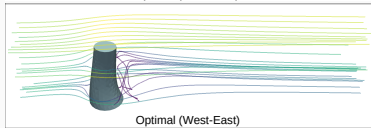
Code



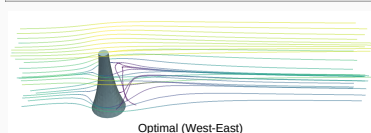
PDE



L^2_μ NO



H^1_μ NO



Thanks to NSF for support: RISE Award 2425922, OAC Award 2313033, DMS Award 2324643

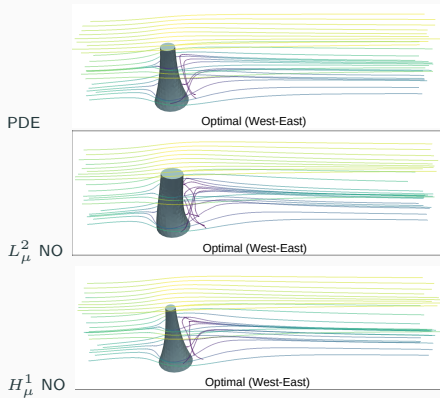
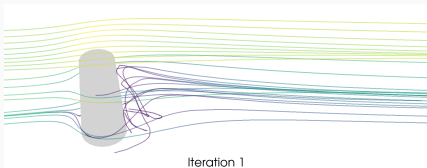
Upcoming preprints

Shape Derivative-Informed Neural Operators

with Omar Ghattas, **Xindi Gong**, Dingcheng Luo, Ruanui Nicholson

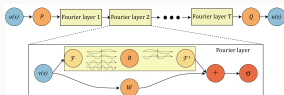
- Parametrized shape optimization
- Inversion for unknown geometry
- Shape design under uncertainty etc.

PDE-constrained optimization



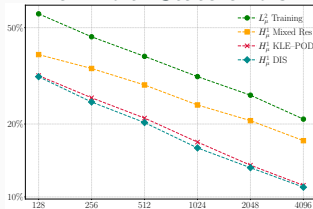
(Derivative-Informed) Fourier Neural Operators

with **Boyuan (John) Yao**, Lianghao Cao, Omar Ghattas, Nikola Kovachki, and Dingcheng Luo

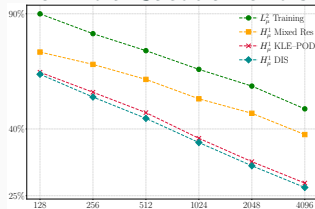


credit: Zongyi Li

Helmholtz State errors



Helmholtz Jacobian errors



FNO C^1 UA on compact sets (informal)

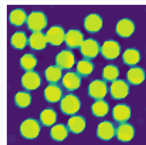
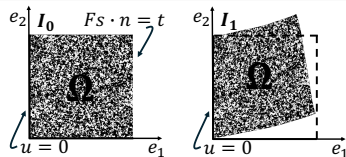
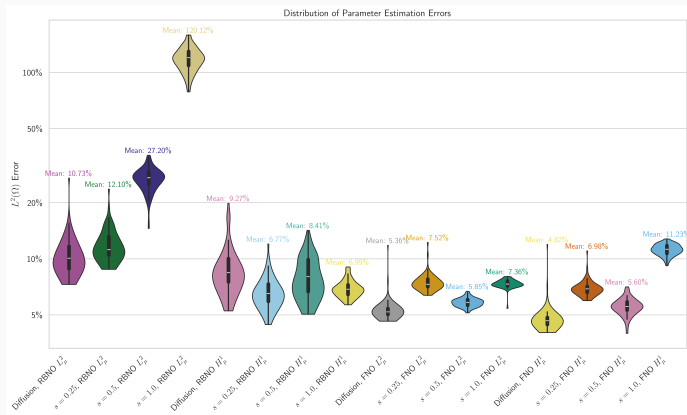
Let $s, s', \delta \geq 0$, and let $\mathcal{M} = H^s(\mathbb{T}^d; \mathbb{R}^d)$, $\mathcal{M}_\delta = H^{s+\delta}(\mathbb{T}^d; \mathbb{R}^d)$ and $\mathcal{U} = H^{s'}(\mathcal{T}^d; \mathbb{R}^d)$. Suppose $u : \mathcal{M} \rightarrow \mathcal{U}$ and $Du : \mathcal{M} \rightarrow HS(\mathcal{M}_\delta, \mathcal{U})$ is continuous. Then for any $K \subset \subset \mathcal{M}$ and $\epsilon > 0$ there exists cts. diff. FNO u_w s.t.

$$\max \left\{ \sup_{m \in \mathcal{M}} \|u(m) - u_w(m)\|_{\mathcal{U}}, \sup_{m \in \mathcal{M}} \|Du(m) - Du_w(m)\|_{HS(\mathcal{M}_\delta, \mathcal{U})} \right\} \leq \epsilon$$

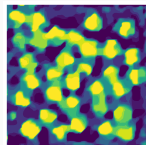
Material inversion using DIC + NOs

a study of architectures, formulations and distributions

with **Joseph Kirchhoff**, **Boyuan (John) Yao**, Josh Chen, Dingcheng Luo and Omar Ghattas



composite
structure



DIFNO +
Diffusion