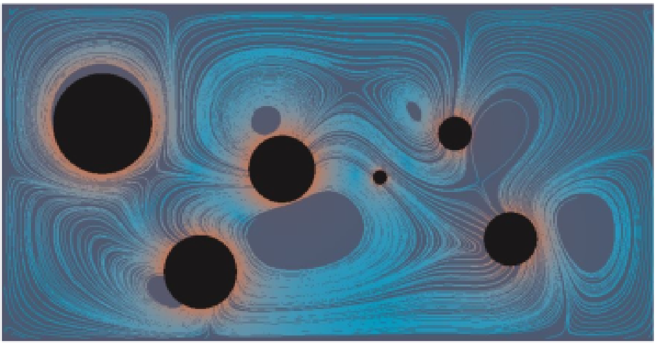
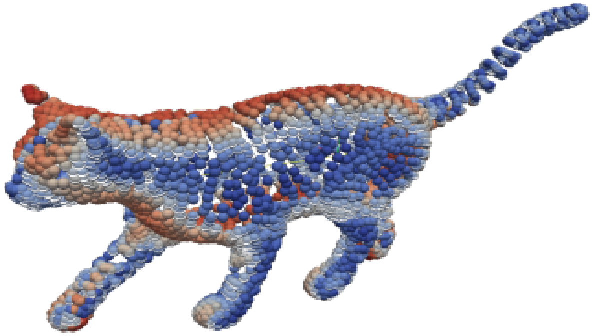
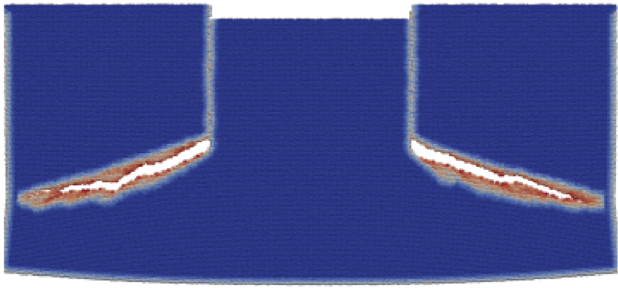
SAND2021-3212PE

Exceptional service in the national interest

Sandia National Laboratories







Convergent and structure preserving architectures for SciML

Nat Trask
Center for Computing Research
Sandia National Laboratories

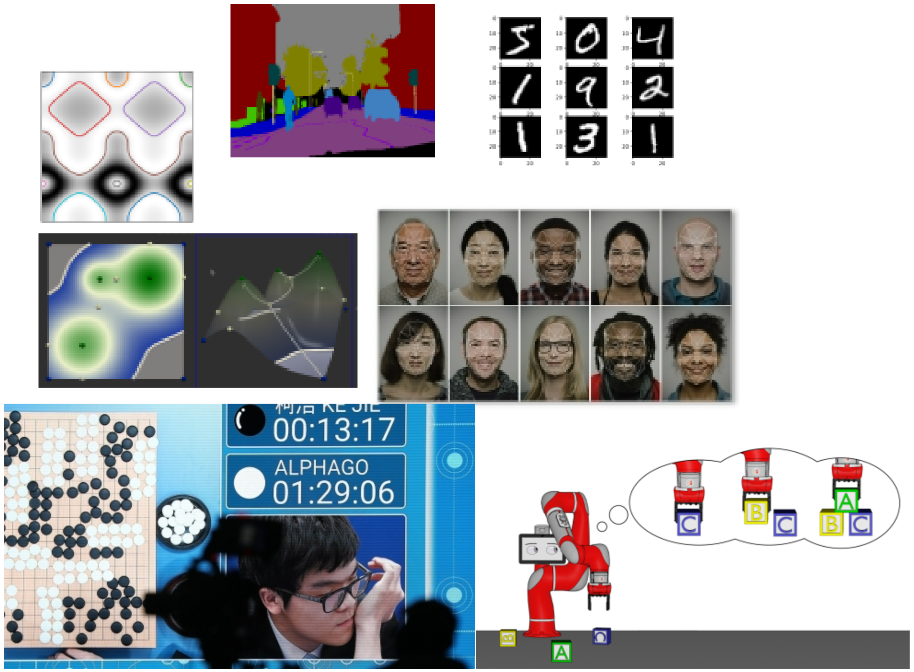
Center for Computing Research

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Scientific machine learning (SciML) vs traditional ML



(Some) traditional ML Tasks

Classification

Image/video processing

Natural language processing

Optimal control

(Some) traditional ML Tools

convNets/uNets/GNN for spatial data

RNN/resnets/LSTM for transient

GANs for distributions

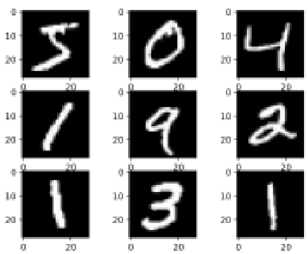
Broadly, much of ML is designed for qualitative comparisons and classification

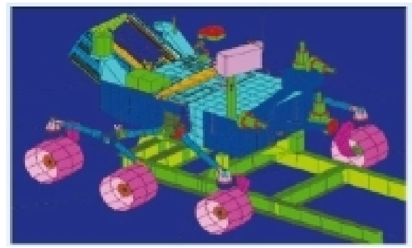
Architectures and training strategies tailored toward a given task

Generally, improved accuracy is “bad”, generalization is “good”

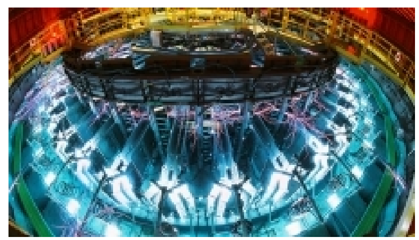
Image credit: google images, Microsoft.com, …

Different requirements for SciML





Complex geometries, physics-based interactions



Labor intensive, expensive + **small** data

Traditional mod+sim tasks
Constitutive modeling
PDE-based physics models
Dynamical systems
Inverse problems + UQ

Traditional tools for mod+sim
Approximation/FEM spaces
Variational principles
Geometric/algebraic structure

SciML requirements:
Small data, accuracy, stability, and uncertainty quantification

Can we embed into off-the-shelf ML to obtain hybrid SciML tools w/ guarantees?

Image credit: google images, techbriefs.com, forbes.com, sandia.gov

Our goal: structure preserving SciML

$$\underset{\xi}{\operatorname{argmin}} ||\mathcal{N}\mathcal{N} - \mathbf{u}_{\text{data}}||^2$$

”Black-box” ML
No physics + big data

$$\underset{\xi}{\operatorname{argmin}} ||\mathcal{N}\mathcal{N} - \mathbf{u}_{\text{data}}||^2 + \epsilon ||\mathbf{L}[\mathcal{N}\mathcal{N};\xi] - \mathbf{f}||^2$$

Physics-informed ML
Weak physics alleviate data requirements

$$\underset{\xi}{\operatorname{argmin}} ||\mathcal{N}\mathcal{N} - \mathbf{u}_{\text{data}}||^2 \quad \text{such that } \mathbf{L}[\mathcal{N}\mathcal{N};\xi] = \mathbf{f}$$

Structure preserving ML
Exact physics treatment independent of data

No domain expertise ⟷ Strong physical priors

Objective: Efficient machine learned surrogates that provide same **accuracy, stability** and **physical realizability** guarantees as traditional forward models **in small data limits**

Parametric high-dim surrogates for UQ + optimal design

For $u_\theta \in \mathbb{R}^d$ seek

$$\underset{\theta}{\text{argmin}}\, \mathcal{F}[u_\theta]$$

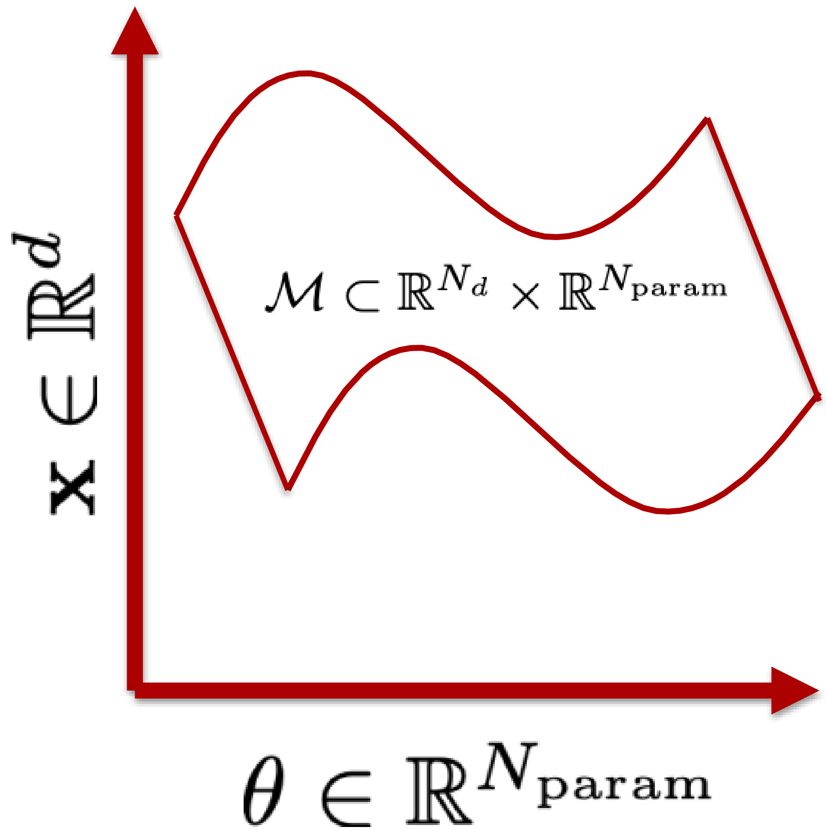
$$\text{s.t. } \mathcal{L}[u_\theta] = f$$

Examples:

- Model discovery/calibration
- Design optimization
- Uncertainty quantification

Promising tools:

- DNNs break c.o.d.
- SciML to enforce constraint



$$dim(\mathcal{M}) \ll N_d \times N_{\text{param}}$$

Outline for today

- **Main idea:** How to engineer traditional modeling and simulation requirements into deep learning frameworks
  - **Q1: Accuracy**
  - **Q2: Structure-preservation and stability**
- Some motivating applications across the laboratories
- A1: Realizing exponential convergence with POU-Nets
- A2: A data-driven exterior calculus for structure preservation

Simple pedagogical examples!

Some data-driven modeling (DDM) exemplars @ SNL

Practical requirements for using SciML in engineering

Extreme/high-risk scenarios require prediction guarantees!

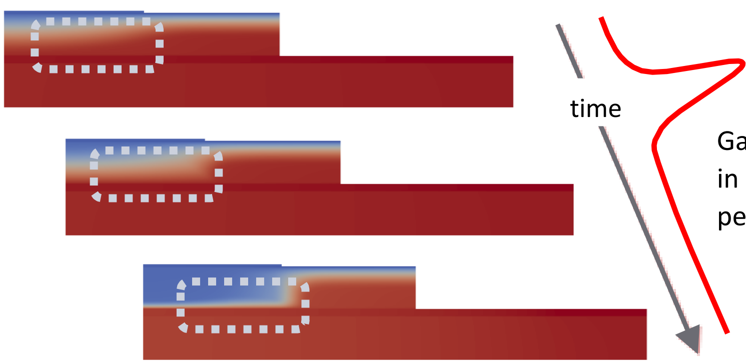
Data-driven models live in an ecosystem of production code

DDM1: Rapid radiation-hardened semiconductor design

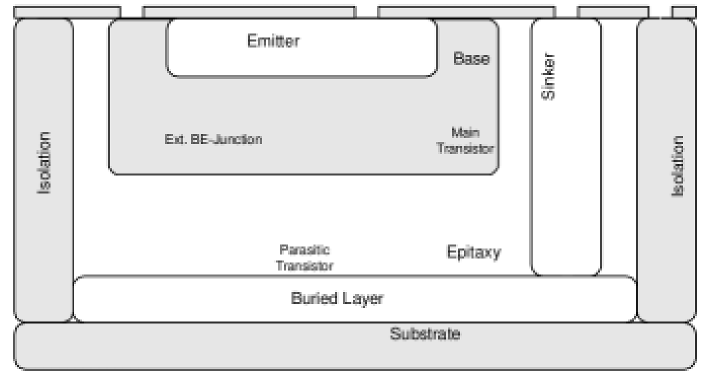
Decade to develop empirical circuit models for a given semiconductor device!

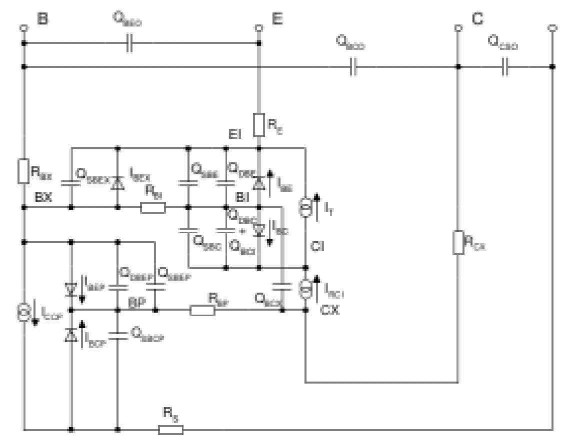
Assimilating new material/radiation effects requires O(1 month) turnaround vs years

DDM idea: Use high-fidelity drift-diffusion PDE model to train a cheap Xyce/DAE circuit model



Gaussian dose rate pulse in semiconductor, peak at 200 ns

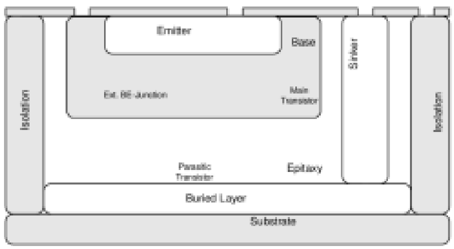




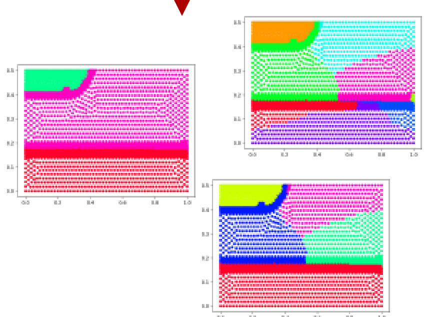
Top: PDE simulation of BJT device
Bottom: Empirical compact/circuit model
Left: Modeling challenge: impact of radiation on nominal device behavior

Gao, X., Huang, A., Trask, N. and Reza, S., Physics-Informed Graph Neural Network for Circuit Compact Model Development. In *2020 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)* (pp. 359-362). IEEE.

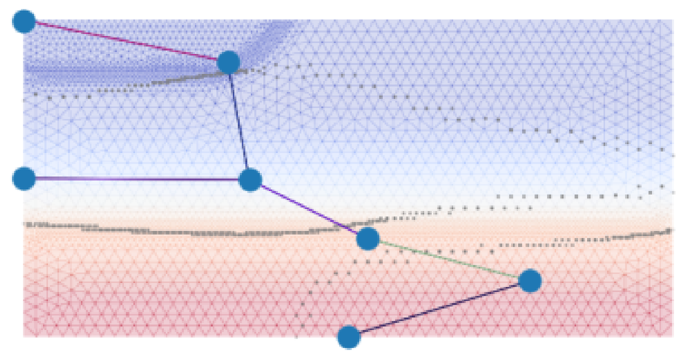
DDM1: Rapid radiation-hardened semiconductor design



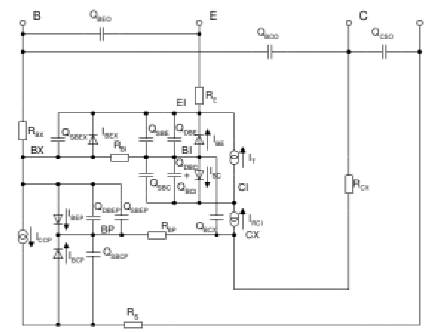
High-fidelity drift-diffusion PDE-based simulation



Partitioning into physics-informed subdomains



Learning data-driven graphical model for voltage-current relation

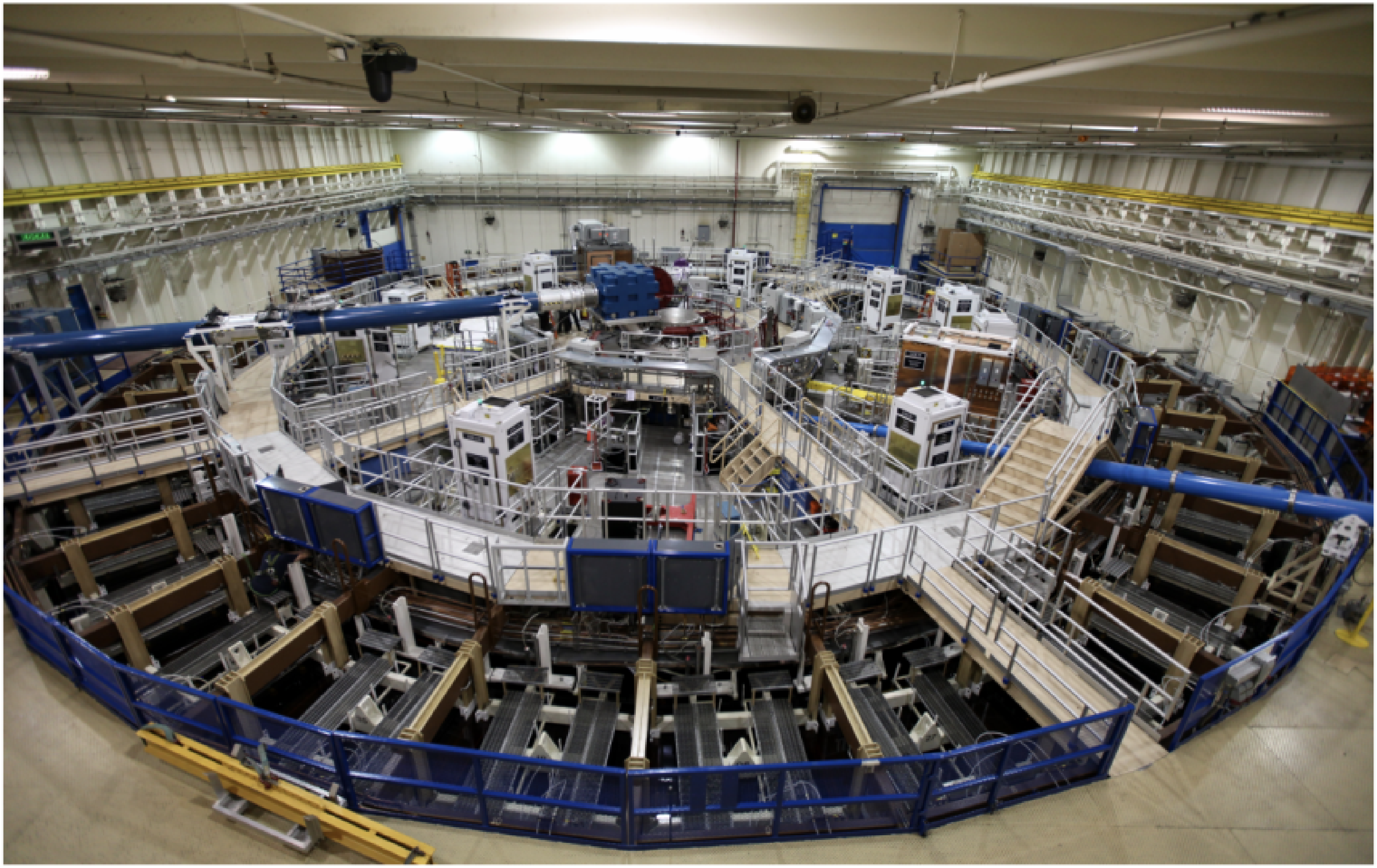


Result: robust surrogate embedded in production circuit simulator

- Data-driven partitioning to extract coarse partitioning of space
- Use exterior calculus ideas to fit control volume analysis to data
- **Result:** reduced order models with structure preservation + guaranteed stability properties that can **reliably be coupled to production circuit simulators**

DDM2: Shock magnetohydro experiments on Z-machine

Sandia National Laboratories



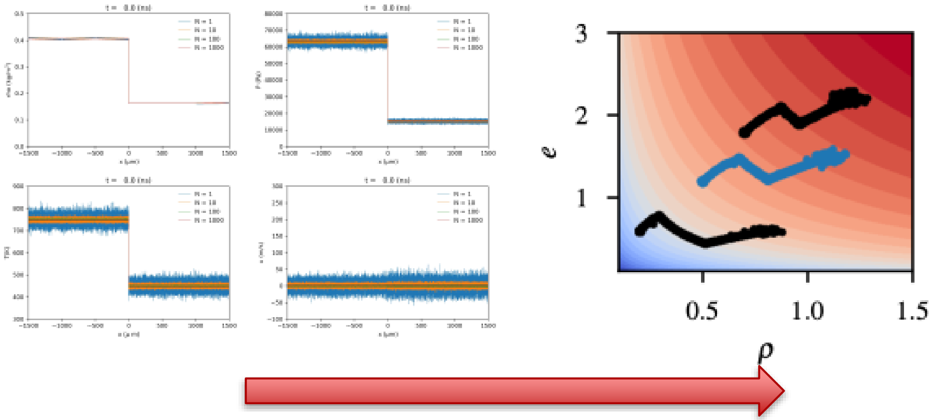
A pulsed power fusion facility for generating extreme environments for short times

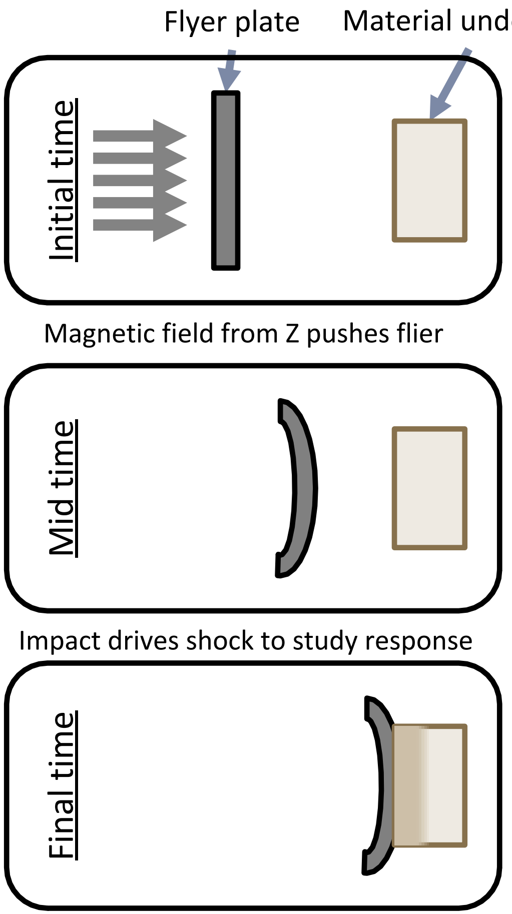
DDM2: Shock magnetohydro experiments on Z-machine

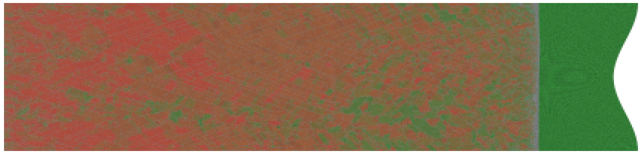
Discovery of material EOS
How to extract EOS under extreme conditions from shock response?

SciML fits black-box EOS to indirect state measurements, embedding thermo

Physics compatibility allows deployment into production codes



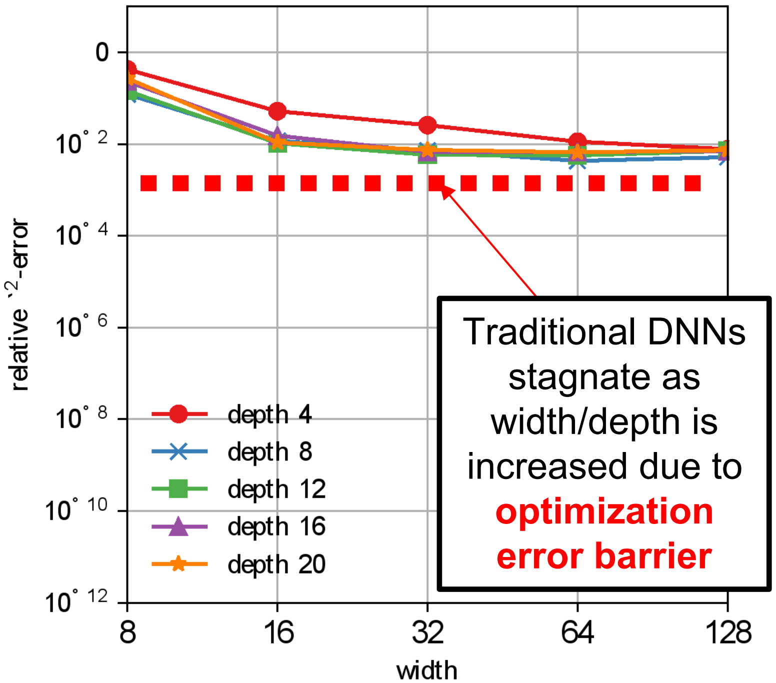




Synthetic data: MD simulations of shocked material

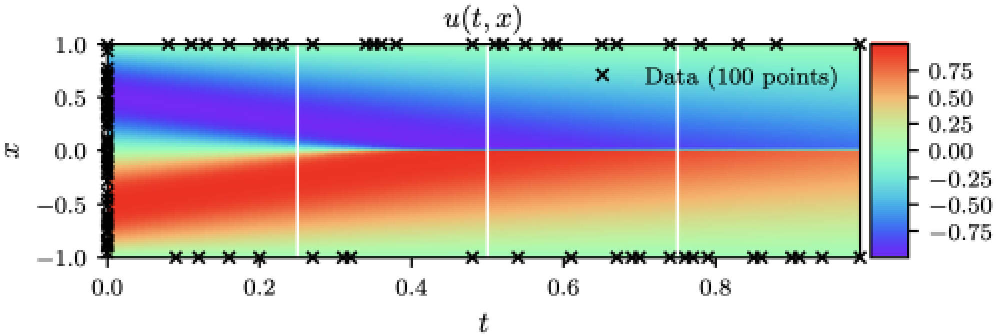
Patel, Ravi G., …, N.Trask, et al. "Thermodynamically consistent physics-informed neural networks for hyperbolic systems." *arXiv preprint arXiv:2012.05343* (2020).

Design requirements and issues for data-driven models



$$\mathcal{L} = \mathcal{L}_{data} + \epsilon_1 \mathcal{L}_{PDE}$$

$$+\epsilon_2 \mathcal{L}_{IC} + \epsilon_3 \mathcal{L}_{BC} + \epsilon_4 \mathcal{L}_{\text{conservation}}$$



#1: Optimization error effecting accuracy

#2: Stability, convergence guarantees, physics

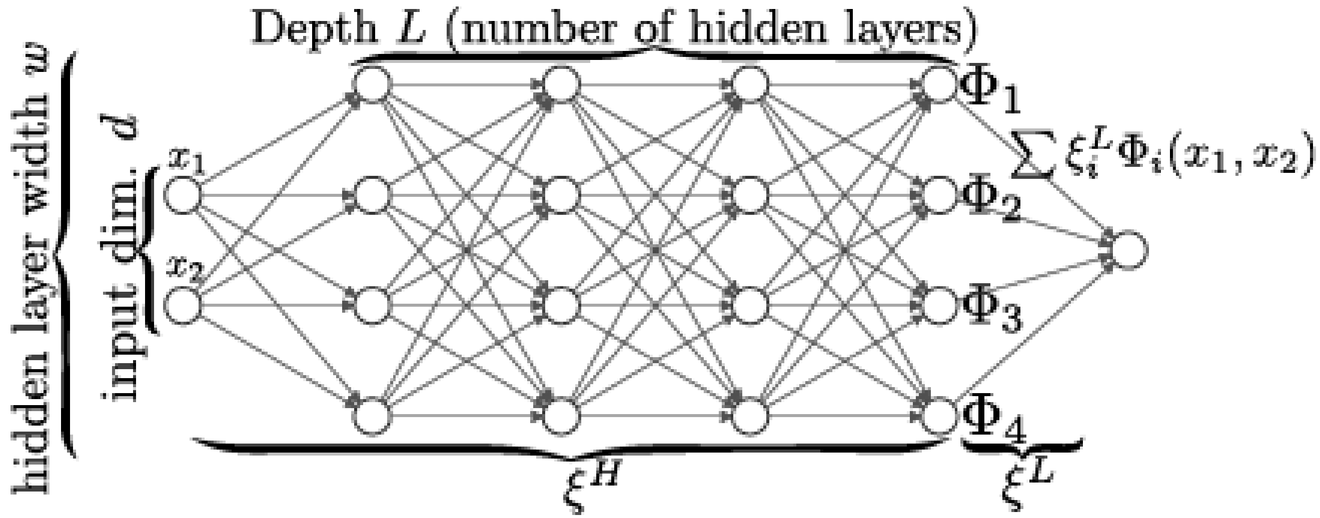
#3: How to fix while preserving high-dim scalability

Raissi, Maziar, Paris Perdikaris, and George Em Karniadakis. "Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations." *arXiv preprint arXiv:1711.10561* (2017).

What needs to be done to augment traditional ML to obtain trustworthy AI for SciML problems?

Part 1: How to build networks with convergence properties

What does a deep network actually do? Find a mesh!

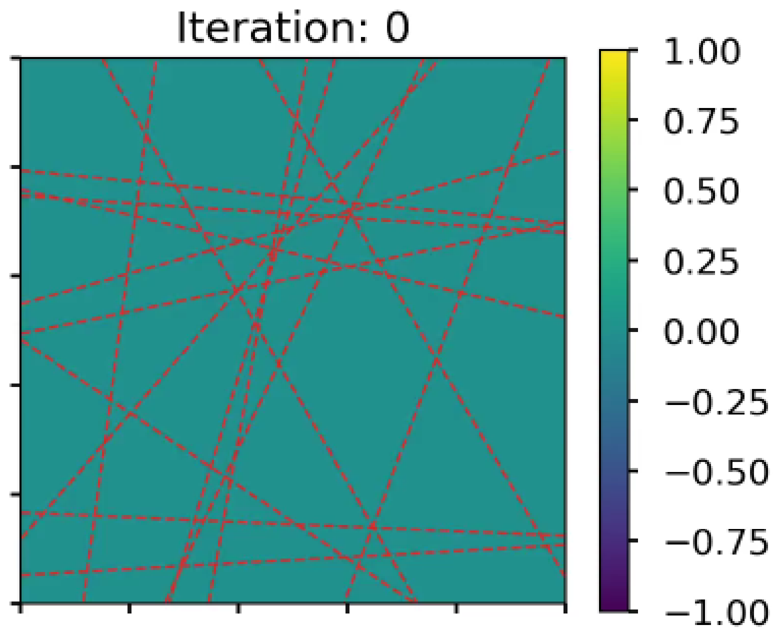


Much of folklore surrounding DNN accuracy related to **universal approximation theorem** giving convergence in infinite limits

To understand actual **convergence rates** lots of recent work provides existence proofs linking to FEM

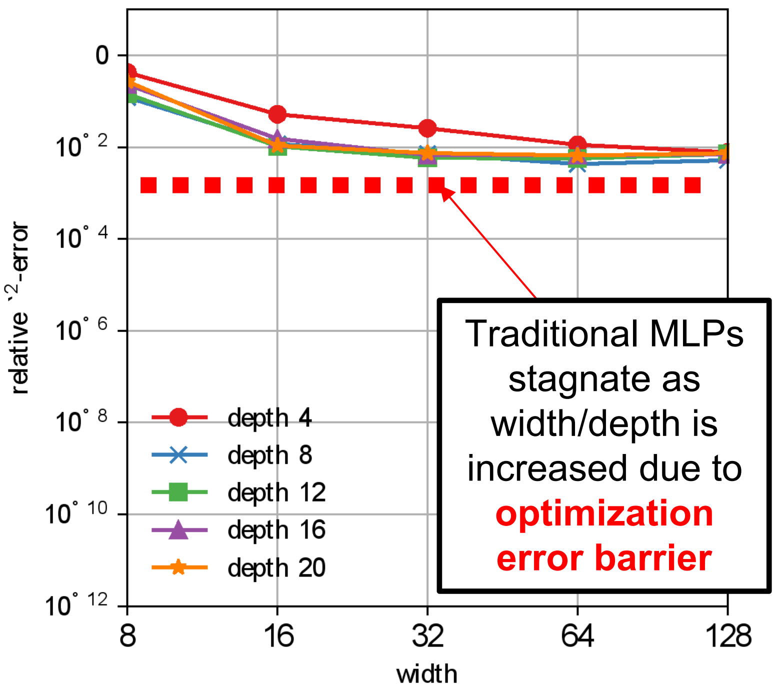
- Algebraic convergence w.r.t. width (Opschoor19)
- ReLU networks as piecewise linear FEM (He18)
- Convergence w.r.t. depth (Telgarsky15,Yarotsky17)

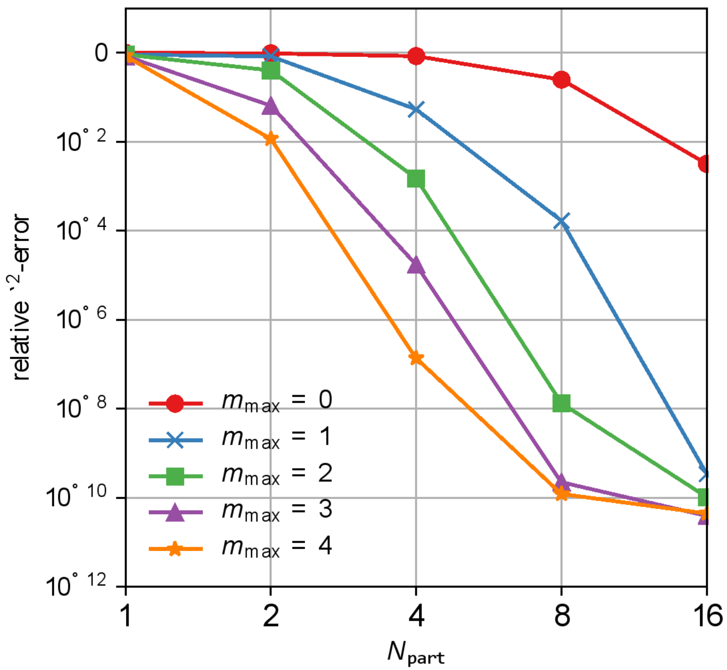
Cyr, E.C., Gulian, M.A., Patel, R.G., Perego, M. and Trask, N.A., 2020, August. Robust training and initialization of deep neural networks: An adaptive basis viewpoint. In *Mathematical and Scientific Machine Learning* (pp. 512-536). PMLR.



Breaking the optimization error barrier - POUnets

These analyses provide a best possible accuracy for a network – but can that be realized in practice when training with first-order optimizers?





Proposed architectures demonstrate algebraic convergence rates for smooth data

References from our group:

1. Cyr, Eric C., et al. "Robust training and initialization of deep neural networks: An adaptive basis viewpoint." *Mathematical and Scientific Machine Learning*. PMLR, (2020).
2. Patel, Ravi G., et al. "A block coordinate descent optimizer for classification problems exploiting convexity." *arXiv preprint arXiv:2006.10123* (2020). Accepted to AAAI-MLPS
3. **Lee, Kookjin, et al. "Partition of unity networks: deep hp-approximation." *arXiv preprint arXiv:2101.11256* (2021) accepted to AAAI-MLPS**

DNNs may emulate traditional approximation spaces

Opschoor et al have established that DNNs may emulate a broad class of approximations: nodal FEM, free-knot splines, spectral approximation, RBFs

Emulation of monomials

Proposition 4.2. *For each $n \in \mathbb{N}_0$ and each polynomial $v \in \mathbb{P}_n([-1,1])$, such that $v(x) = \sum_{\ell=0}^{n} \tilde{v}_\ell x^\ell$, for all $x \in [-1,1]$ with $C_0 := \sum_{\ell=2}^{n} |\tilde{v}_\ell|$, there exist NNs $\{\Phi_\beta^v\}_{\beta \in (0,1)}$ with input dimension one and output dimension one which satisfy*

$$\begin{aligned}
\left\| v - \mathrm{R}(\Phi_\beta^v) \right\|_{W^{1,\infty}(\hat{I})} &\leq \beta, \\
\mathrm{R}(\Phi_\beta^v)(0) &= v(0), \\
L(\Phi_\beta^v) &\leq C_L(1+\log_2(n))\log_2(C_0/\beta) + \tfrac{1}{3}C_L(\log_2(n))^3 + C(1+\log_2(n))^2, \\
M(\Phi_\beta^v) &\leq 4C_M n \log_2(C_0/\beta) + 8C_M n \log_2(n) + 4C_L(1+\log_2(n))^2\log_2(C_0/\beta) + C(1+n), \\
M_{\mathrm{fi}}(\Phi_\beta^v) &\leq 4\log_2(n) + 4, \\
M_{\mathrm{la}}(\Phi_\beta^v) &\leq 4n + 2
\end{aligned}$$

if $C_0 > \beta$. If $C_0 \leq \beta$ the same estimates hold, but with C_0 replaced by 2β.

Emulation of piecewise polynomial space

Proposition 5.1. *For all $\boldsymbol{p} = (p_i)_{i \in \{1,\dots,N\}} \subset \mathbb{N}$, all partitions $\mathcal{T}$ of $I = (0,1)$ into N open, disjoint, connected subintervals and for all $v \in S_{\boldsymbol{p}}(I,\mathcal{T})$, for $0 < \varepsilon < 1$ exist NNs $\{\Phi_\varepsilon^{v,\mathcal{T},\boldsymbol{p}}\}_{\varepsilon \in (0,1)}$ such that for all $1 \leq q' \leq \infty$ holds*

$$\begin{aligned}
\left\| v - \mathrm{R}\left(\Phi_\varepsilon^{v,\mathcal{T},\boldsymbol{p}}\right) \right\|_{W^{1,q'}(I)} &\leq \varepsilon |v|_{W^{1,q'}(I)}, \\
L\left(\Phi_\varepsilon^{v,\mathcal{T},\boldsymbol{p}}\right) &\leq C_L(1+\log_2(p_{\max}))\left(2p_{\max} + \log_2(1/\varepsilon)\right) + C_L\log_2(1/\varepsilon) + C\left(1+\log_2^3(p_{\max})\right), \\
M\left(\Phi_\varepsilon^{v,\mathcal{T},\boldsymbol{p}}\right) &\leq 8C_M\sum_{i=1}^{N} p_i^2 + 4C_M\log_2(1/\varepsilon)\sum_{i=1}^{N} p_i + \log_2(1/\varepsilon)\,C\left(1+\sum_{i=1}^{N}\log_2^2(p_i)\right) \\
&\quad + C\left(1+\sum_{i=1}^{N} p_i\log_2^2(p_i)\right) \\
&\quad + 2N\left(C_L(1+\log_2(p_{\max}))\left(2p_{\max}+\log_2(1/\varepsilon)\right) + C\left(1+\log_2^3(p_{\max})\right)\right), \\
M_{\mathrm{fi}}\left(\Phi_\varepsilon^{v,\mathcal{T},\boldsymbol{p}}\right) &\leq 6N, \\
M_{\mathrm{la}}\left(\Phi_\varepsilon^{v,\mathcal{T},\boldsymbol{p}}\right) &\leq 2N + 2.
\end{aligned}$$

In addition, it holds that $\mathrm{R}\left(\Phi_\varepsilon^{v,\mathcal{T},\boldsymbol{p}}\right)(x_j) = v(x_j)$ for all $j \in \{0,\dots,N\}$, where $\{x_j\}_{j=0}^{N}$ are the nodes of $\mathcal{T}$.

Opschoor, J.A., Petersen, P.C. and Schwab, C., 2020. Deep ReLU networks and high-order finite element methods. *Analysis and Applications*, *18*(05), pp.715-770.

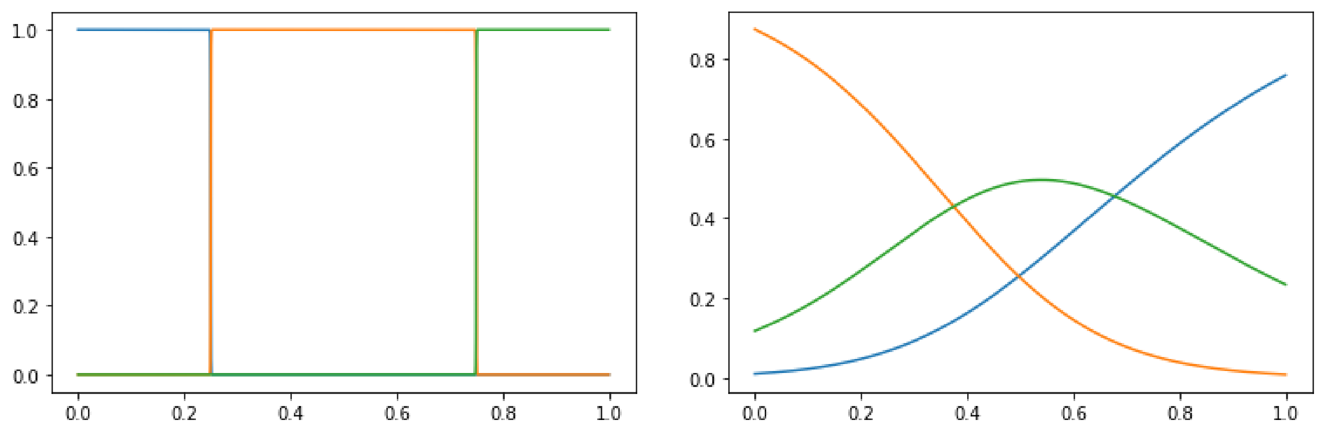
Partition of unity

Key role:
Localizing approximation

Definition: *Partition of unity (POU)*
A collection of functions $\{\phi_i\}_{i=1,\dots,N}$ satisfying

- $\phi_i > 0$
- $\sum_i \phi_i = 1$

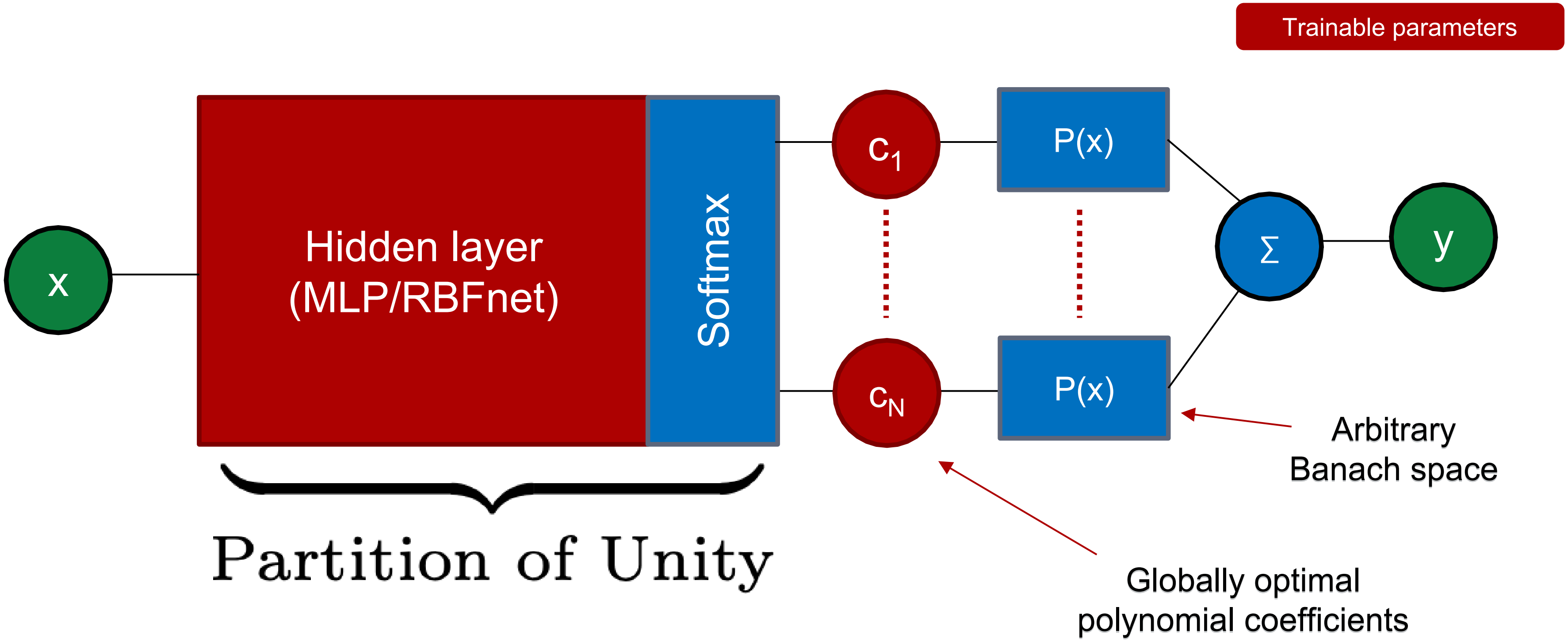
Example:
Consider a partition of $\Omega \subset \mathbb{R}^d$ into disjoint cells $\Omega = \bigcup_i C_i$. Then the indicator functions $\phi_i(x) = \mathbb{1}_{C_i}(x)$ form a POU.



POU corresponding to Cartesian mesh, and another with non-disjoint supports

POU-Net

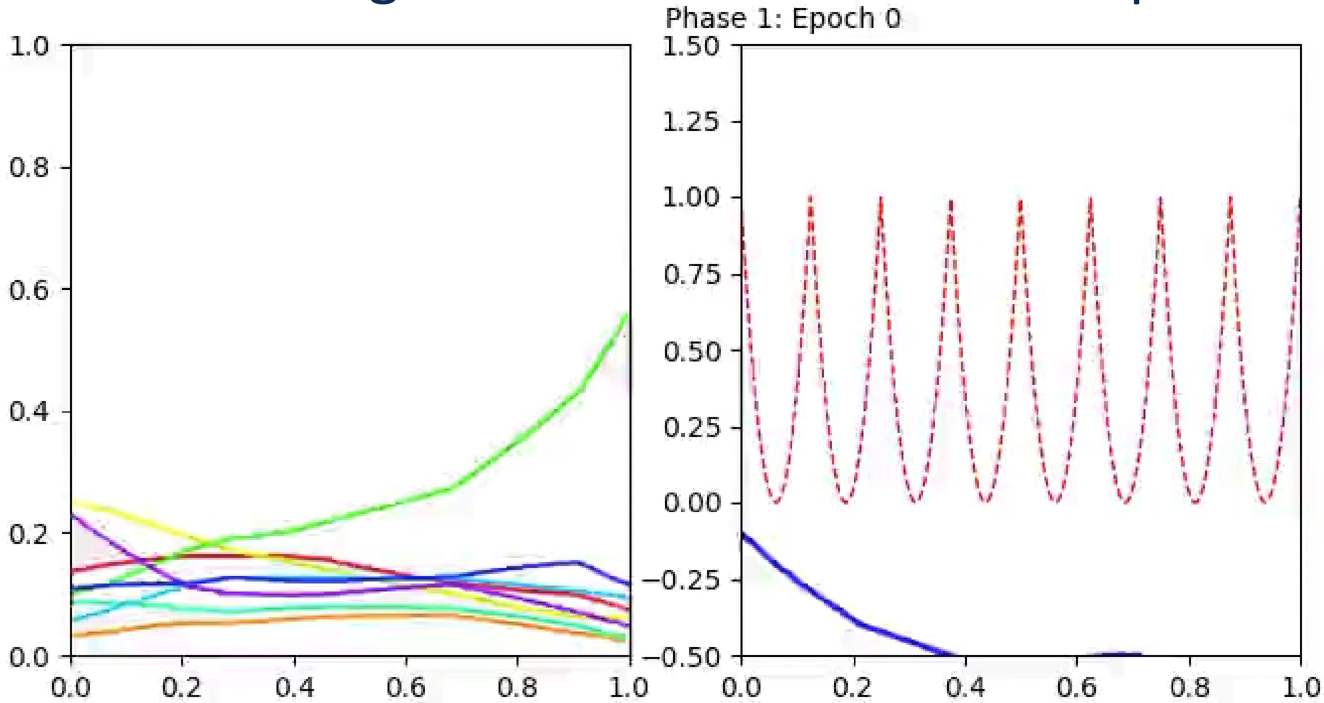
Main idea: rather than emulate POU + monomials, build them directly into architecture



Training:

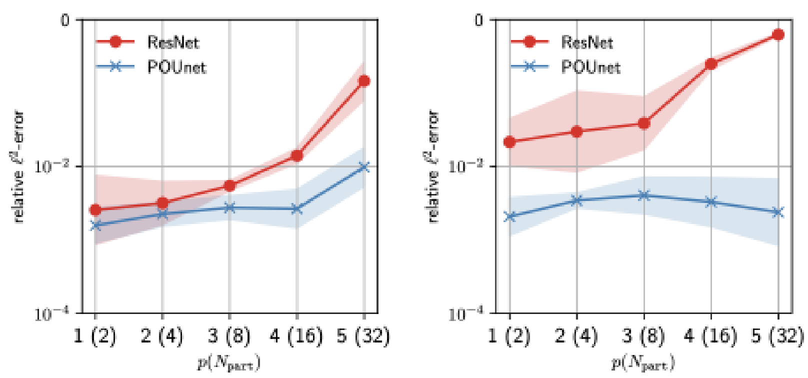
1. Solve weighted least squares for **globally optimal** coefficients
2. Apply gradient update to adjust partition

A “meshfree” generation of a traditional hp-FEM space



Using ResNets for POUs allow discontinuities in partitions

Top left: Evolution of partitions on unit interval
Top right: Optimal reconstruction (blue) of piecewise quadratic space (red)
Bottom right: Convergence vs ResNet



(a) Triangular waves (b) Quadratic waves

An "aspirational" error estimate

Theorem 1. *Consider an approximant y_{POU} of the form* (1) *with $V = \pi_m(\mathbb{R}^d)$. If $y(\cdot) \in C^{m+1}(\Omega)$ and $\boldsymbol{\xi}^*, \mathbf{c}^*$ solve* (3) *to yield the approximant y^*_{POU}, then*

$$\|y^*_{POU} - y\|^2_{\ell_2(\mathcal{D})} \leq C_{m,y} \max_{\alpha} diam\left(supp(\phi^{\boldsymbol{\xi}}_{\alpha})\right)^{m+1} \quad (4)$$

where $\|y^*_{POU} - y\|_{\ell_2(\mathcal{D})}$ denotes the root-mean-square norm over the training data pairs in $\mathcal{D}$,

$$\|y^*_{POU} - y\|_{\ell_2(\mathcal{D})} = \sqrt{\frac{1}{N_{data}} \sum_{(\mathbf{x},y)\in\mathcal{D}} \left(y^*_{POU}(\mathbf{x}) - y(\mathbf{x})\right)^2},$$

and

$$C_{m,y} = \|y\|_{C^{m+1}(\Omega)}.$$

Proof. For each α, take $q_\alpha \in \pi_m(\mathbb{R}^d)$ to be the mth order Taylor polynomial of $y(\cdot)$ centered at any point of $\text{supp}(\phi^{\boldsymbol{\xi}}_\alpha)$. Then for all $\mathbf{x} \in \text{supp}(\phi^{\boldsymbol{\xi}}_\alpha)$,

$$|q_\alpha(\mathbf{x}) - y(\mathbf{x})| \leq C_{m,y}\, \text{diam}\left(\text{supp}(\phi^{\boldsymbol{\xi}}_\alpha)\right)^{m+1}. \quad (5)$$

Define the approximant $\widetilde{y}_{\text{POU}} = \sum_{\alpha=1}^{N_{\text{part}}} \phi^{\boldsymbol{\xi}}_\alpha(\mathbf{x}) q_\alpha(\mathbf{x})$, which is of the form (1) and represented by feasible $(\boldsymbol{\xi}, \mathbf{c})$. Then by definition of y^*_{POU} and (3), we have

$$\begin{aligned}
\|y^*_{\text{POU}}(\mathbf{x}) - y(\mathbf{x})\|^2_{\ell_2(\mathcal{D})} &\leq \|\widetilde{y}_{\text{POU}}(\mathbf{x}) - y(\mathbf{x})\|^2_{\ell_2(\mathcal{D})} \\
&= \left\| \sum_{\alpha=1}^{N_{\text{part}}} \phi^{\boldsymbol{\xi}}_\alpha(\mathbf{x}) q_\alpha(\mathbf{x}) - y(\mathbf{x}) \sum_{\alpha=1}^{N_{\text{part}}} \phi^{\boldsymbol{\xi}}_\alpha(\mathbf{x}) \right\|^2_{\ell_2(\mathcal{D})} \\
&= \left\| \sum_{\alpha=1}^{N_{\text{part}}} \phi^{\boldsymbol{\xi}}_\alpha(\mathbf{x}) \left(q_\alpha(\mathbf{x}) - y(\mathbf{x})\right) \right\|^2_{\ell_2(\mathcal{D})}.
\end{aligned}$$

For each $\mathbf{x} = \mathbf{x}_i \in \mathcal{D}$, if $\mathbf{x} \in \text{supp}(\mathcal{D})$, then we apply (5); otherwise, the summand $\phi^{\boldsymbol{\xi}}_\alpha(\mathbf{x}) \left(q_\alpha(\mathbf{x}) - y(\mathbf{x})\right)$ vanishes. So

$$\begin{aligned}
&\|y^*_{\text{POU}}(\mathbf{x}) - y(\mathbf{x})\|^2_{\ell_2(\mathcal{D})} \\
&\leq \left\| \sum_{\alpha=1}^{N_{\text{part}}} C_{m,y}\, \text{diam}\left(\text{supp}(\phi^{\boldsymbol{\xi}}_\alpha)\right)^{m+1} \phi^{\boldsymbol{\xi}}_\alpha(\mathbf{x}) \right\|^2_{\ell_2(\mathcal{D})} \\
&\leq C_{m,y} \max_\alpha \text{diam}\left(\text{supp}(\phi^{\boldsymbol{\xi}}_\alpha)\right)^{m+1} \left\| \sum_{\alpha=1}^{N_{\text{part}}} \phi^{\boldsymbol{\xi}}_\alpha(\mathbf{x}) \right\|^2_{\ell_2(\mathcal{D})} \\
&\leq C_{m,y} \max_\alpha \text{diam}\left(\text{supp}(\phi^{\boldsymbol{\xi}}_\alpha)\right)^{m+1}.
\end{aligned}$$

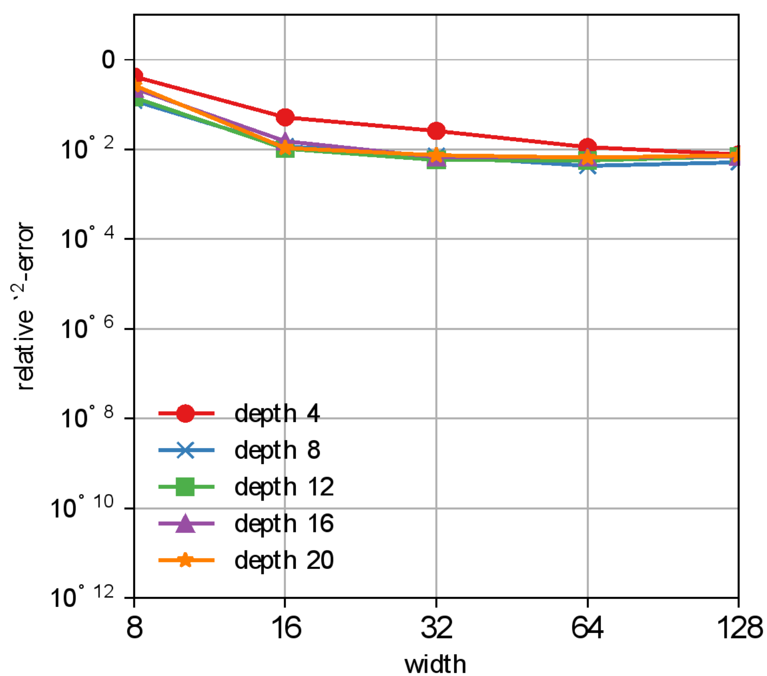
- If reconstructing with polynomials, and **POU with compact support** is found, we realize hp-convergence for smooth functions **independent of dimension**
- Prompts questions for how to promote sparsity in POU parameterization + training (see paper)

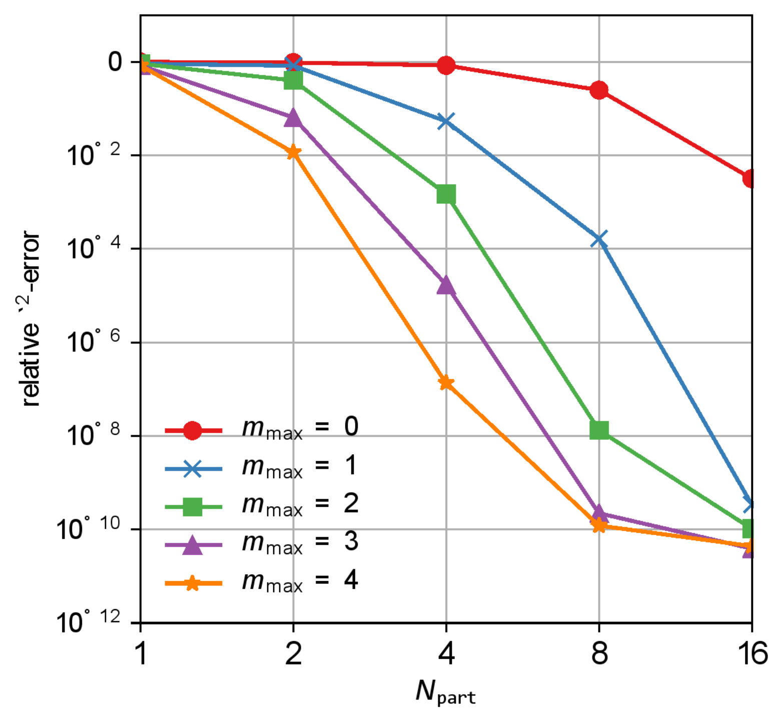
Lee, Kookjin, et al. "Partition of unity networks: deep hp-approximation." *arXiv preprint arXiv:2101.11256* (2021) accepted to AAAI-MLPS

hp-convergence for smooth functions

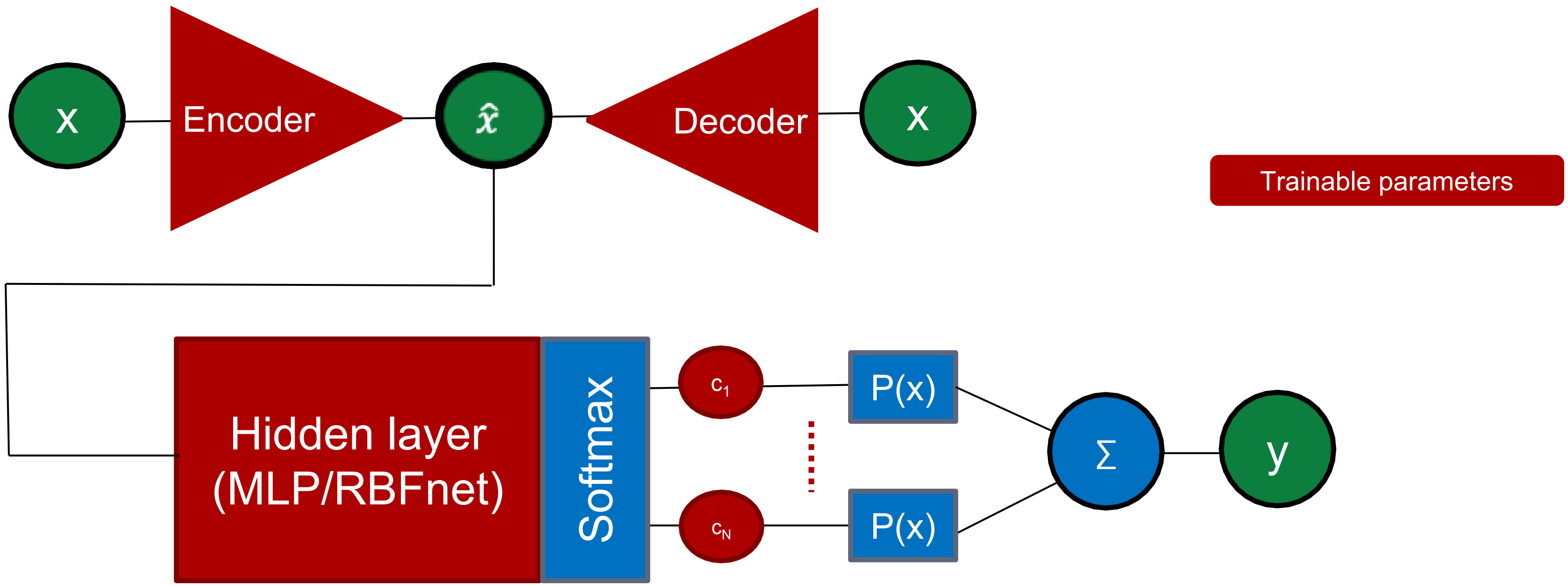
$$\mathcal{M} \subset \{x = 0, y \in [-1, 1]\} \bigcup \{y = 0, x \in [-1, 1]\}$$

$$f(x, y) = \sin(2\pi x)\ \sin(2\pi y)$$



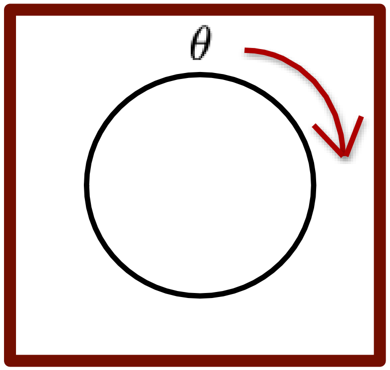


High-dimensional approximation

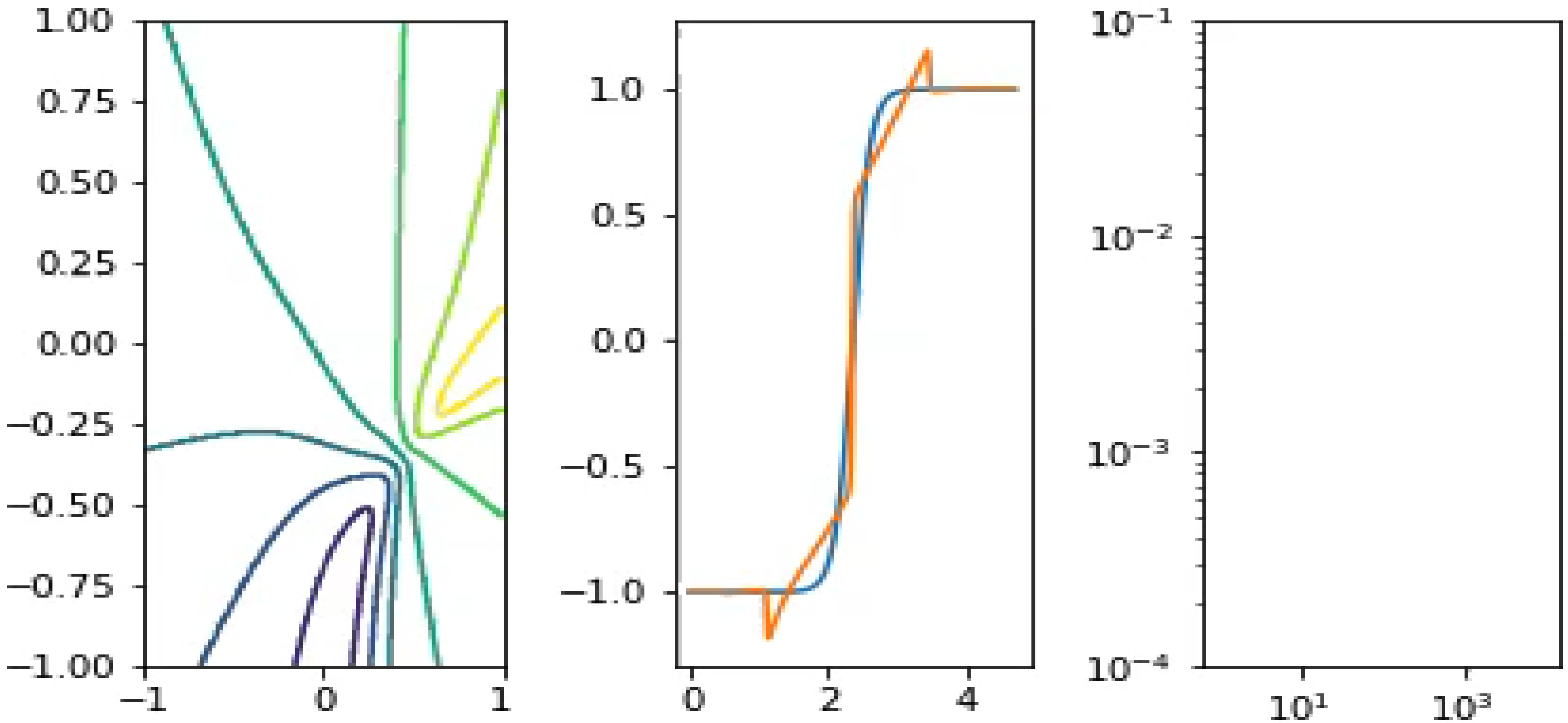


IDEA:
Discovery of latent dimensional coordinates
"Mesh" underlying manifold

EXAMPLE:
"Find" theta coordinate and mesh manifold



Example – discovering a mesh of the unit sphere



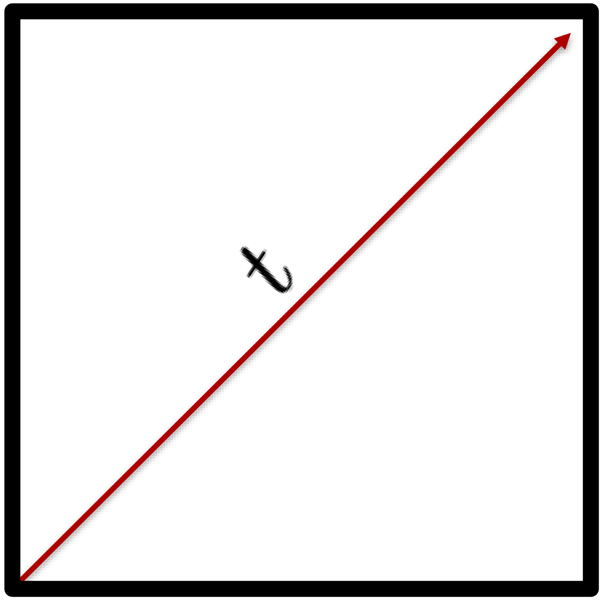
Isocontours of i^{th} partition

Target (blue)
Regressed (orange)

RMS loss

Linear basis
Data sampled from θ in [0,1.5π]

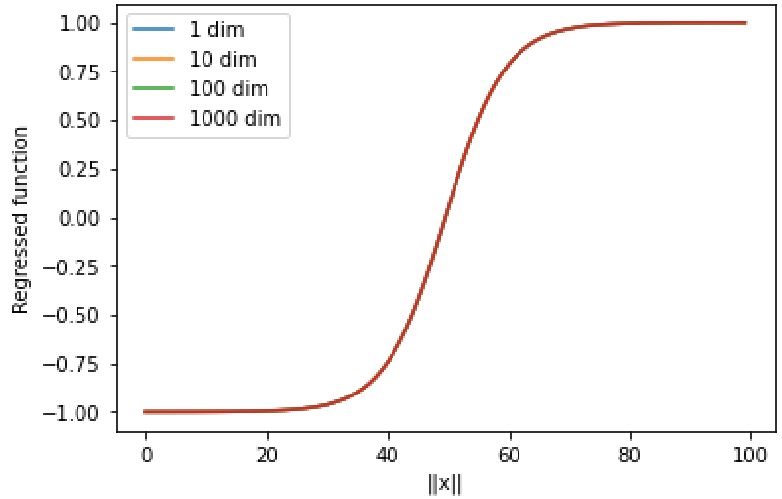
Example – breaking curse of dimensionality

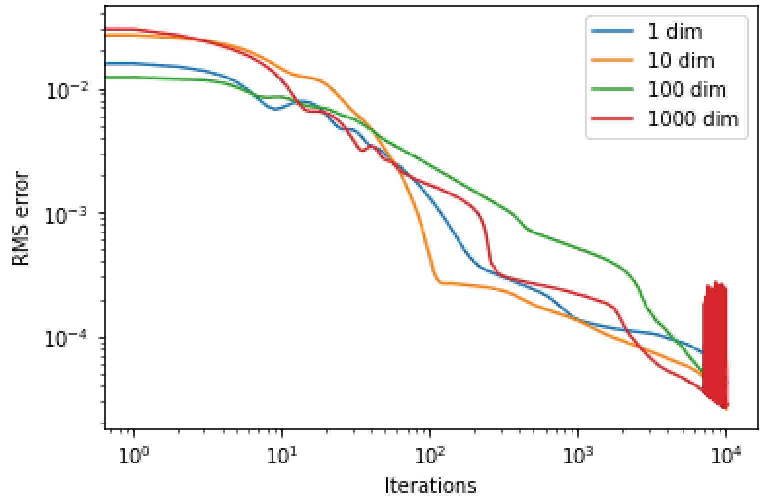


$$\mathcal{M} \subset [0,1]^d$$

$$\Gamma_{\mathcal{M}}(t) = < 0, \ldots, 0 > + t^2 < 1, \ldots, 1 >$$

$$f(t) = tanh\left(10\left(\frac{t}{\sqrt{d}} - \frac{1}{2}\right)\right)$$





What needs to be done to augment traditional ML to obtain trustworthy AI for SciML problems?

Part 1: How to build networks with convergence properties

Part 2: How to preserve structure related to physics-compatibility, stability, and well-posedness

What are physics compatible discretizations for PDEs?

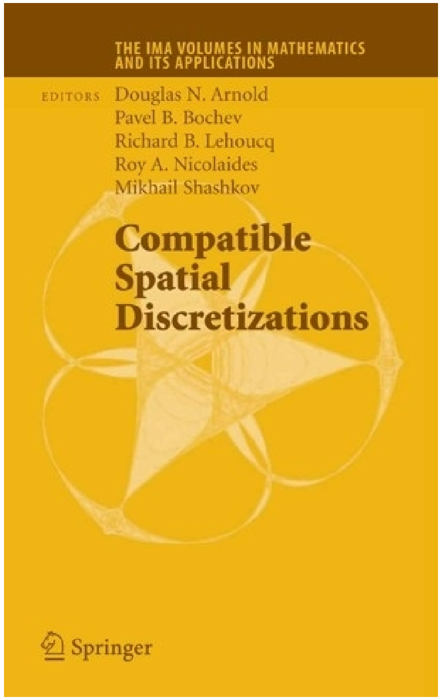
Sandia National Laboratories

Methods for solving PDEs which:

Use generalized Stokes theorems to approximate differential operators

Preserve topological structure in governing equations

Mimic properties of continuum operators (thus sometimes called **mimetic discretizations**)



Arnold, D. N., Bochev, P. B., Lehoucq, R. B., Nicolaides, R. A., & Shashkov, M. (Eds.). (2007). *Compatible spatial discretizations* (Vol. 142). Springer Science & Business Media.

Two key ingredients:

<u>1: A topological structure</u>

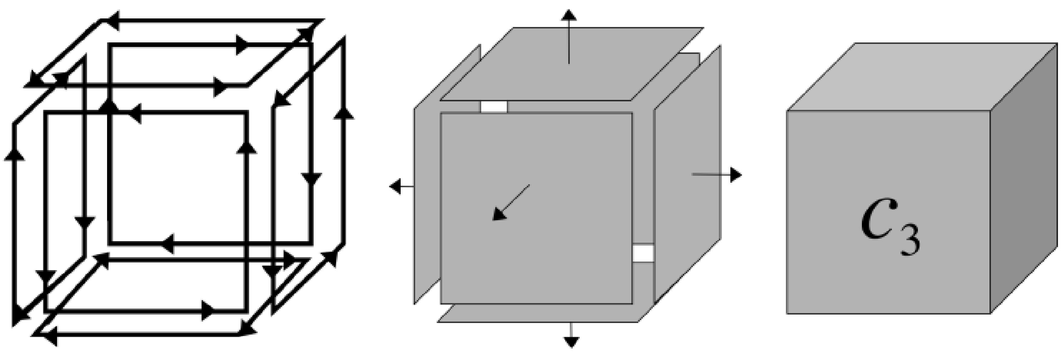
In PDE discretization this is a mesh, with boundary operators linking cells, faces, edges, and nodes

We will use a graph as an inexpensive low-dimensional mesh surrogate

<u>2: Metric information</u>

Measures associated with mesh entities, ensuring discrete exterior derivatives converge to div/grad/curl

Graphs are purely topological with no natural metric, we will use ML to extract metric information from data



$$0 \leftarrow \partial\partial c_3 \xleftarrow{\partial} \partial c_3 \xleftarrow{\partial} c_3$$

$$\nabla \cdot \mathbf{u} = \frac{1}{\mu(C)} \sum_{f \in \partial C} \int_f \mathbf{u} \cdot d\mathbf{A}$$

The ingredients to the discrete exterior calculus

Chain complex

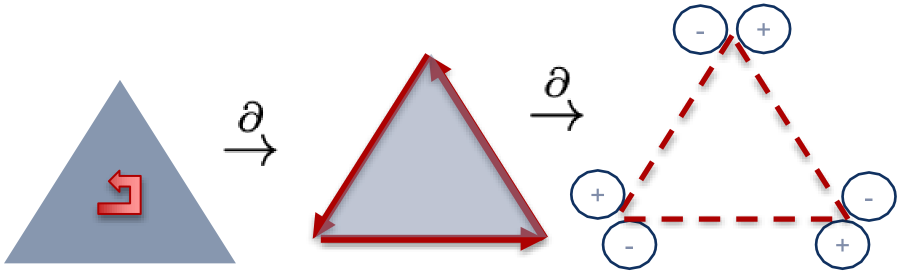
$$C_0 \xleftarrow{\partial_0} C_1 \xleftarrow{\partial_1} C_2 \xleftarrow{\partial_2} C_3$$

Cochain complex

$$C^0 \xrightarrow{d_0} C^1 \xrightarrow{d_1} C^2 \xrightarrow{d_2} C^3$$

Codifferentials

$$C^0 \underset{d_0^\star}{\overset{d_0}{\rightleftarrows}} C^1 \underset{d_1^\star}{\overset{d_1}{\rightleftarrows}} C^2 \underset{d_2^\star}{\overset{d_2}{\rightleftarrows}} C^3$$



$$\int_\omega du = \int_{\partial\omega} u$$

$$(v, d_k^\star u)_k = (d_k v, u)_{k+1}$$

Compatible PDE

K+1-simplices as chains
Stokes theorem give cochains
L2-adjoints give codifferentials

Combinatorial Hodge

K-cliques as chains
Graph div/grad/curl give cochains
Use data to obtain codifferentials

A data-driven exterior calculus (DDEC)

$$
\begin{array}{ccccccccccc}
C^0 & \xleftarrow{\mathsf{d}_0^*} & C^1 & \xleftarrow{\mathsf{d}_1^*} & C^2 & \xleftarrow{\mathsf{d}_2^*} & C^3 & \xleftarrow{\mathsf{d}_3^*} & \cdots & \xleftarrow{\mathsf{d}_{d-1}^*} & C^d \\
\downarrow \mathbf{D}_0^{-1} & & \downarrow \mathbf{D}_1^{-1} & & \downarrow \mathbf{D}_2^{-1} & & \downarrow \mathbf{D}_3^{-1} & & & & \downarrow \mathbf{D}_d^{-1} \\
C^0 & \underset{\delta_0}{\overset{\delta_0^*}{\rightleftarrows}} & C^1 & \underset{\delta_1}{\overset{\delta_1^*}{\rightleftarrows}} & C^2 & \underset{\delta_2}{\overset{\delta_2^*}{\rightleftarrows}} & C^3 & \underset{\delta_3}{\overset{\delta_3^*}{\rightleftarrows}} & \cdots & \underset{\delta_{d-1}}{\overset{\delta_0^*}{\rightleftarrows}} & C^d \\
\downarrow \mathbf{B}_0 & & \downarrow \mathbf{B}_1 & & \downarrow \mathbf{B}_2 & & \downarrow \mathbf{B}_3 & & & & \downarrow \mathbf{B}_d \\
C^0 & \xrightarrow{\mathsf{d}_0} & C^1 & \xrightarrow{\mathsf{d}_1} & C^2 & \xrightarrow{\mathsf{d}_2} & C^3 & \xrightarrow{\mathsf{d}_3} & \cdots & \xrightarrow{\mathsf{d}_{d-1}} & C^d
\end{array}
$$

Idea: Take graph calculus and introduce learnable inner products

$$(x, y)_{\mathbf{B}_k} = x^\intercal \mathbf{B}_k y$$

$$(x, y)_{\mathbf{D}_k} = x^\intercal \mathbf{D}_k y$$

to find data-driven exterior calculus operators that inherit the structure of graph exterior calculus

What does all this give you?

$$
\begin{array}{ccccccccc}
C^0 & \xleftarrow[\mathbf{d}_0^*]{} & C^1 & \xleftarrow[\mathbf{d}_1^*]{} & C^2 & \xleftarrow[\mathbf{d}_2^*]{} & C^3 & \xleftarrow[\mathbf{d}_3^*]{} \cdots \xleftarrow[\mathbf{d}_{d-1}^*]{} & C^d \\
\downarrow \mathbf{D}_0^{-1} & & \downarrow \mathbf{D}_1^{-1} & & \downarrow \mathbf{D}_2^{-1} & & \downarrow \mathbf{D}_3^{-1} & & \downarrow \mathbf{D}_d^{-1} \\
C^0 & \underset{\delta_0}{\overset{\delta_0^*}{\leftrightarrows}} & C^1 & \underset{\delta_1}{\overset{\delta_1^*}{\leftrightarrows}} & C^2 & \underset{\delta_2}{\overset{\delta_2^*}{\leftrightarrows}} & C^3 & \underset{\delta_3}{\overset{\delta_3^*}{\leftrightarrows}} \cdots \underset{\delta_{d-1}}{\overset{\delta_0^*}{\leftrightarrows}} & C^d \\
\downarrow \mathbf{B}_0 & & \downarrow \mathbf{B}_1 & & \downarrow \mathbf{B}_2 & & \downarrow \mathbf{B}_3 & & \downarrow \mathbf{B}_d \\
C^0 & \xrightarrow{\mathbf{d}_0} & C^1 & \xrightarrow{\mathbf{d}_1} & C^2 & \xrightarrow{\mathbf{d}_2} & C^3 & \xrightarrow{\mathbf{d}_3} \cdots \xrightarrow{\mathbf{d}_{d-1}} & C^d
\end{array}
$$

- Differential operators which locally and globally conserve fluxes, circulations, potentials
- Invertible Hodge Laplacians $\Delta_k = d^\star_{k+1} d_{k+1} + d_k d^\star_{k+1}$
- Exact sequence properties $d_{k+1} d_k = d^\star_k d^\star_{k+1} = 0$
- Hodge decomposition $u = d^* \alpha + d\beta + \gamma$
- Corollary: treatment of nontrivial null-spaces in electromagnetism

Theorems…

Theorem 3.1. *The discrete derivatives* d_k *in* (11) *form an exact sequence if the simplicial complex is exact, and in particular* $\mathsf{d}_{k+1} \circ \mathsf{d}_k = 0$. *In* $\mathbb{R}^3$, *we have* $CURL_h \circ GRAD_h = DIV_h \circ CURL_h = 0$.

Theorem 3.2. *The discrete derivatives* d_k^* *in* (11) *form an exact sequence of the simplicial complex is exact, and in particular* $\mathsf{d}_k^* \circ \mathsf{d}_{k+1}^* = 0$. *In* $\mathbb{R}^3$, $DIV_h^* \circ CURL_h^* = CURL_h^* \circ GRAD_h^* = 0$.

Theorem 3.3 (Hodge Decomposition). *For* C^k, *the following decomposition holds*

$$C^k = \operatorname{im}(\mathsf{d}_{k-1}) \bigoplus\nolimits_k \ker(\Delta_k) \bigoplus\nolimits_k \operatorname{im}(\mathsf{d}_k^*), \tag{17}$$

where $\bigoplus_k$ *means the orthogonality with respect to the* $(\cdot,\cdot)_{\mathbf{D}_k \mathbf{B}_k^{-1}}$*-inner product.*

Theorem 3.4 (Poincaré inequality). *For each* k, *there exists a constant* $c_{P,k}$ *such that*

$$\|\mathbf{z}_k\|_{\mathbf{D}_k \mathbf{B}_k^{-1}} \le c_{P,k} \|\mathsf{d}_k \mathbf{z}_k\|_{\mathbf{D}_{k+1} \mathbf{B}_{k-1}^{-1}}, \quad \mathbf{z}_k \in \operatorname{im}(\mathsf{d}_k^*),$$

and another constant $c_{P,k}^*$ *such that*

$$\|\mathbf{z}_k\|_{\mathbf{D}_k \mathbf{B}_k^{-1}} \le c_{P,k}^* \|\mathsf{d}_{k-1}^* \mathbf{z}_k\|_{\mathbf{D}_{k-1} \mathbf{B}_{k-1}^{-1}}, \quad \mathbf{z}_k \in \operatorname{im}(\mathsf{d}_{k-1}).$$

Thus, for $\mathbf{u}_k \in C^k$, *we have*

$$\inf_{\mathbf{h}_k \in \ker(\Delta_k)} \|\mathbf{u}_k - \mathbf{h}_k\|_{\mathbf{D}_k \mathbf{B}_k^{-1}} \le C \left(\|\mathsf{d}_k \mathbf{u}_k\|_{\mathbf{D}_{k+1} \mathbf{B}_{k+1}^{-1}} + \|\mathsf{d}_{k-1}^* \mathbf{u}_k\|_{\mathbf{D}_{k-1} \mathbf{B}_{k-1}^{-1}} \right),$$

where constant $C > 0$ *only depends on* $c_{P,k}$ *and* $c_{P,k}^*$.

Theorem 3.5 (Invertibility of Hodge Laplacian). *The* k^{th}*-order Hodge Laplacian* Δ_k *is positive-semidefinite, with the dimension of its null-space equal to the dimension of the corresponding homology* $H^k = \ker(\mathsf{d}_k)/\operatorname{im}(\mathsf{d}_{k-1})$.

Details: Trask, Nathaniel, Andy Huang, and Xiaozhe Hu. "Enforcing exact physics in scientific machine learning: a data-driven exterior calculus on graphs." *arXiv preprint arXiv:2012.11799* (2020).

Using DDEC to discover structure preserving surrogates

$$\nabla \cdot \mathbf{F} = f$$

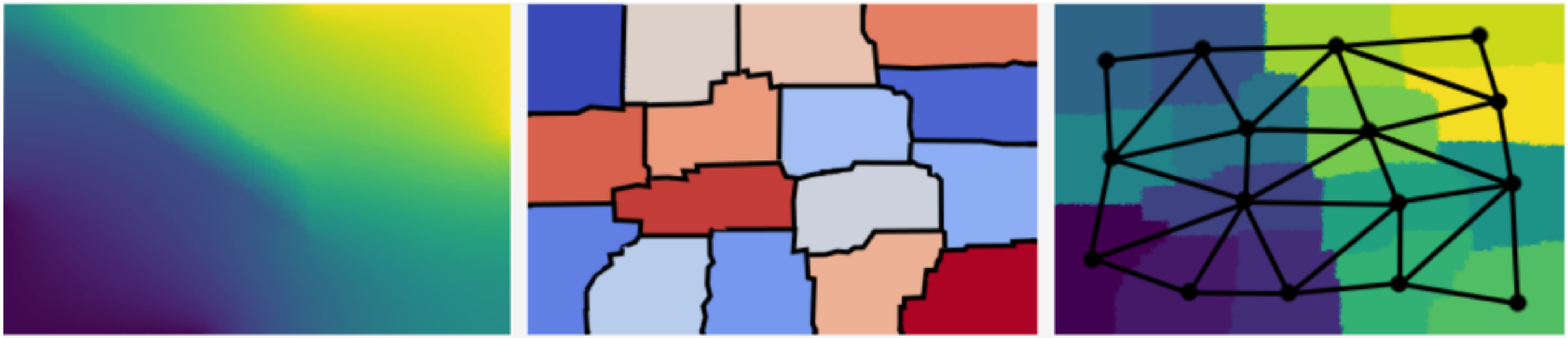
$$\mathbf{F} + \kappa \nabla \phi = 0$$

Structure preserving trainable exterior derivatives

$$d_0^\star \mathbf{F} = f$$

$$\mathbf{F} + \xi d_0 \phi + \mathcal{N}_\eta(\phi) = 0$$

Black box NN flux



High-fidelity PDE solution

Apply graph-cut to coarse-grain chain complex

Average over partitions to obtain training data

General optimization problem

Sandia National Laboratories

Fluxes: $$\mathbf{w}_{k+1} = \mathrm{d}_k \mathbf{u}_k + \epsilon \mathcal{N}\mathcal{N}(\mathrm{d}_k \mathbf{u}_k; \xi),$$

Conservation: $$\mathrm{d}_{k-1}\ \mathrm{d}^*_{k-1} \mathbf{u}_k + \mathrm{d}^*_k \mathbf{w}_{k+1} = \mathbf{f}_k.$$

$$\Rightarrow \quad a(\mathbf{v}, \mathbf{u}; \mathbf{B}, \mathbf{D}) + N_{\mathbf{v}}[\mathbf{u}; \xi] = b(\mathbf{v})$$

Invertible bilinear form w/ metric params

Nonlinear perturbation with DNN params

If we can fit the model to data while imposing equality constraint, then during training we restrict to manifold of solvable models preserving physics

$$\underset{\mathbf{B},\mathbf{D},\xi}{\mathrm{argmin}}, ||\mathbf{w} - \mathbf{w}_{\mathrm{data}}||^2$$

$$\text{such that } \mathcal{L}[\mathbf{w}, \mathbf{u}; \mathbf{B}, \mathbf{D}, \xi] = 0$$

Is PDE constraint well posed?

$$a(\mathbf{v}, \mathbf{u}; \mathbf{B}, \mathbf{D}) + N_{\mathbf{v}}[\mathbf{u}; \xi] = b(\mathbf{v})$$

Theorem 3.6. *The equation* (24) *has at least one solution* $\mathbf{u}_k \in \mathbb{V}$ *satisfies*

$$\|\mathbf{u}_k\| \leq \frac{\|\mathbf{f}\|}{(C_p - C_N)}. \tag{26}$$

Theorem 3.7. *If* $\frac{C_{\nabla N}\|\mathbf{f}\|}{C_p(C_p - C_N)} < 1$, *then the equation* (24) *has at most one solution in* $\mathbb{V}$.

A unique solution exists if the Hodge-Laplacian is sufficiently large relative to the nonlinear part, following standard elliptic PDE arguments

- Poincare constant easily estimated from matrix eigenvalues
- Lipschitz constant on nonlinearity straightforward for DNNs

Solvability constraint could be enforced during training if desired

“PDE”-constrained optimization

$$\mathbf{L}_{\mathbf{u},\lambda,\mathbf{B},\mathbf{D},\xi} = ||\mathbf{w} - \mathbf{w}_{\text{data}}||^2 + \lambda^\intercal \mathcal{L}[\mathbf{w},\mathbf{u};\mathbf{B},\mathbf{D},\xi]$$

$$\mathcal{L}[\mathbf{w},\mathbf{u};\mathbf{B},\mathbf{D},\xi] = 0$$

An iterative algorithm guaranteeing exact enforcement of physics at each iteration:

- Solve forward problem with current model parameters

$$\mathbf{w},\mathbf{u} \leftarrow \nabla_\lambda \mathbf{L}_{\mathbf{u},\lambda,\mathbf{B},\mathbf{D},\xi} = 0$$

- Solve adjoint problem with current forward solution

$$\lambda \leftarrow \nabla_{\mathbf{u}} \mathbf{L}_{\mathbf{u},\lambda,\mathbf{B},\mathbf{D},\xi} = 0$$

- Apply gradient descent to update model

$$\mathbf{B},\mathbf{D},\xi \leftarrow \nabla_{\mathbf{B},\mathbf{D},\xi} \mathbf{L}_{\mathbf{u},\lambda,\mathbf{B},\mathbf{D},\xi} = 0$$

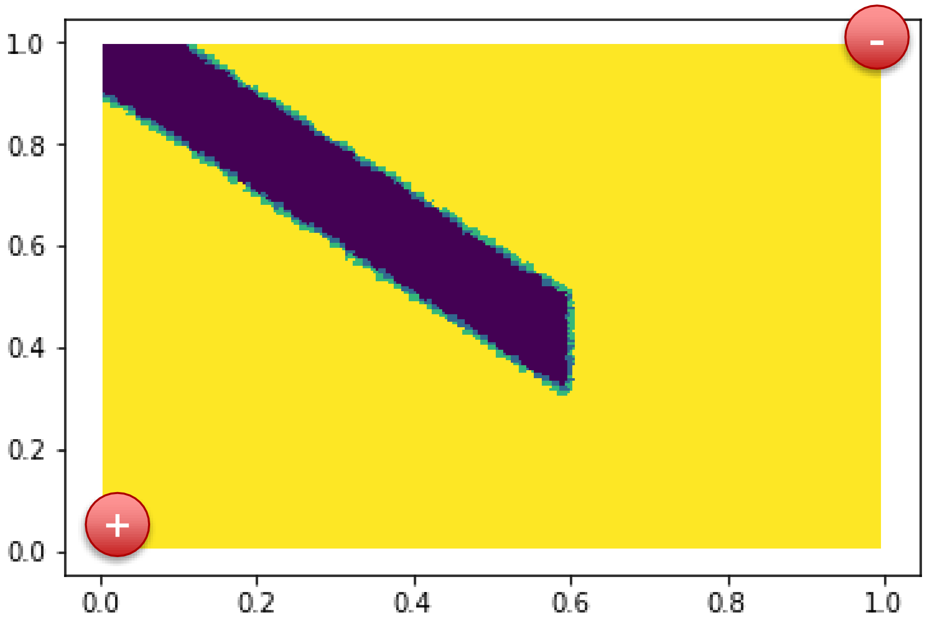
Back to Darcy...

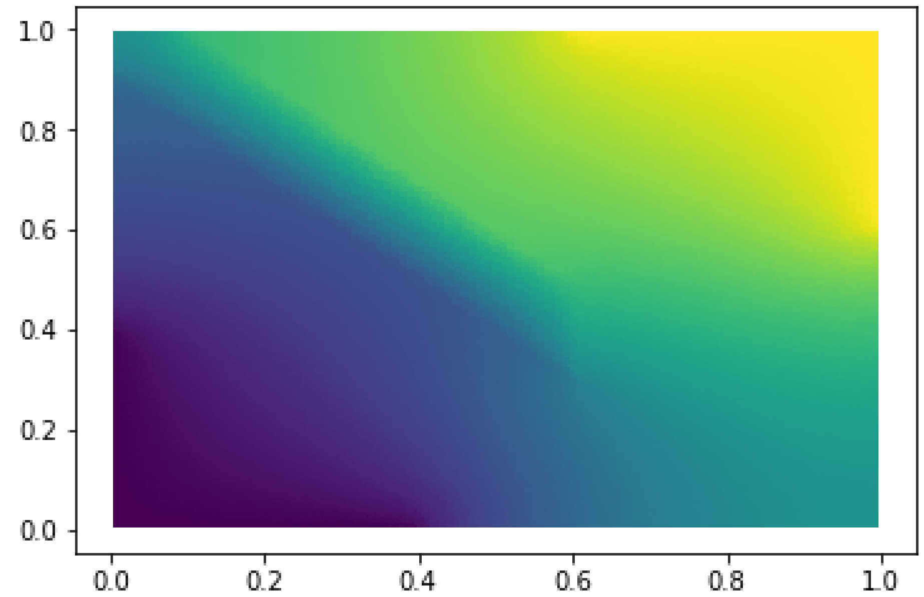
$$\nabla \cdot \mathbf{F} = f$$

$$\mathbf{F} + \kappa \nabla \phi = 0$$

$$d_0^{\star} \mathbf{F} = f$$

$$\mathbf{F} + \xi d_0 \phi + \mathcal{N}_\eta(\phi) = 0$$





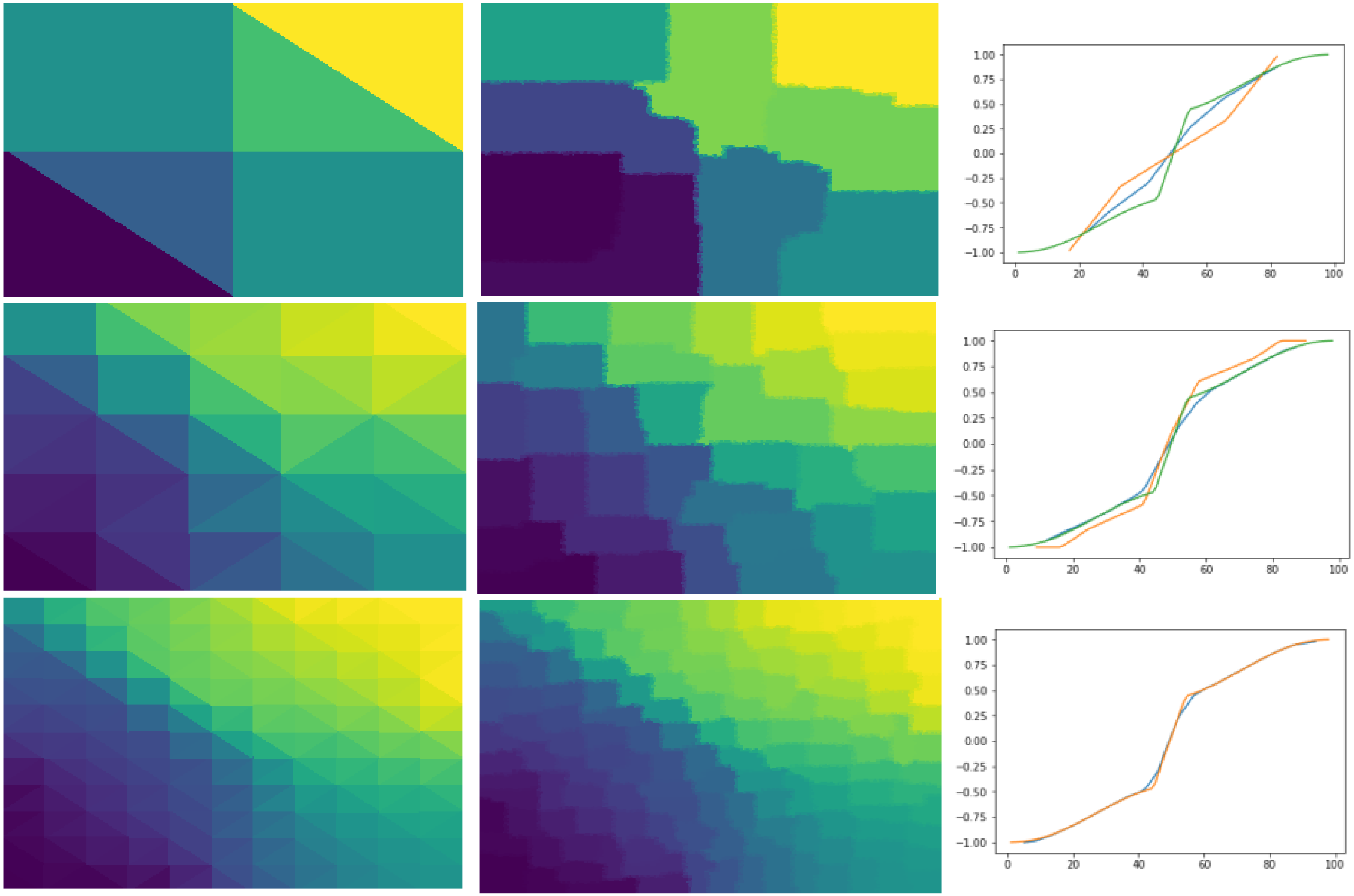
Comparison to traditional covolume: improved accuracy at low resolution

Sandia National Laboratories

N = 2^2

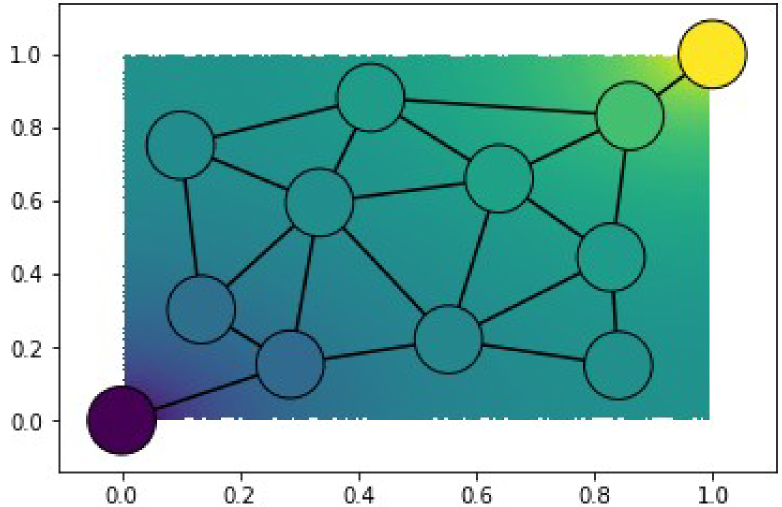
N = 5^2

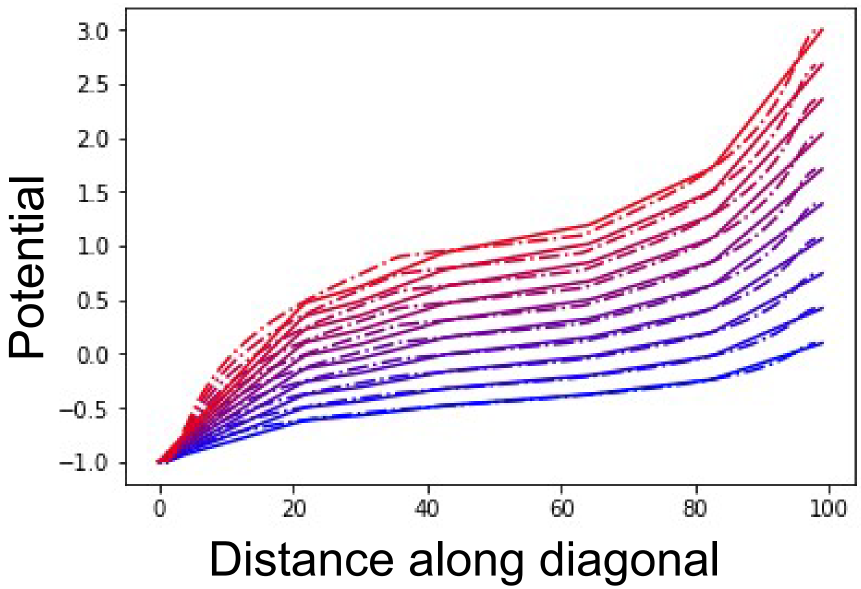
N = 10^2

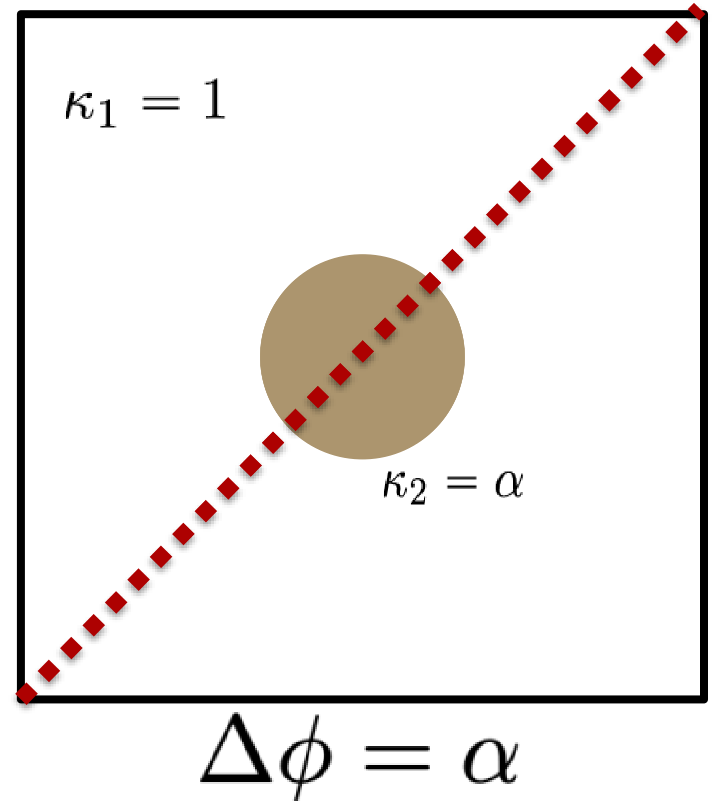


Comparison of pressure for same # DOF for FVM (left) and DDEC (center)
Right: profile along diagonal shows better fit to solution (green) by DDEC (blue) vs FVM (orange)

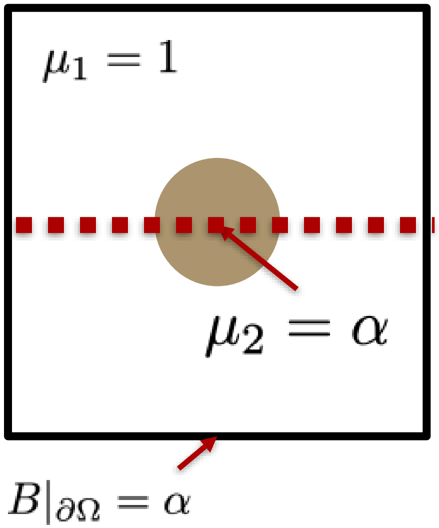
Nonlinear Darcy: potential profile across diagonal







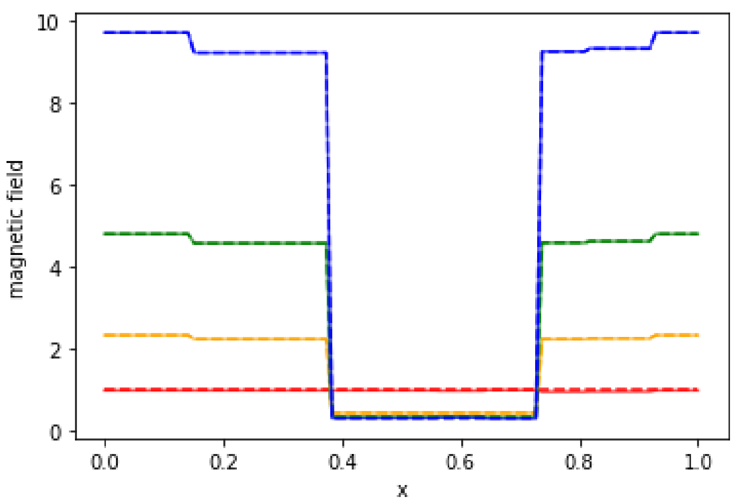
The rest of the de Rham complex - magnetostatics

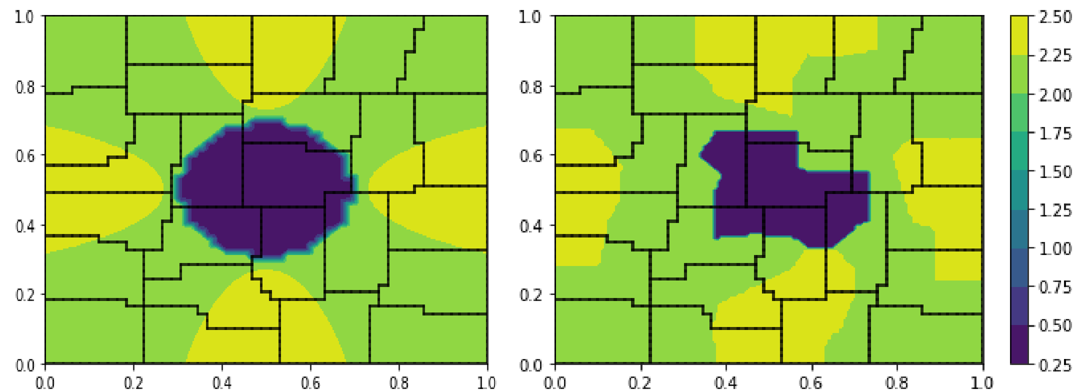


$$\nabla \times \frac{1}{\mu} \nabla \times \mathbf{u} = \mathbf{f}$$

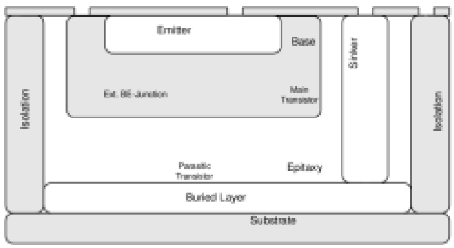
$$\nabla \cdot \mathbf{u} = 0$$

Extracted surrogate:
Is exactly div free
Provides sharp interfaces
Conserves circulation
Guaranteed solvable
Generalizes to other BCs

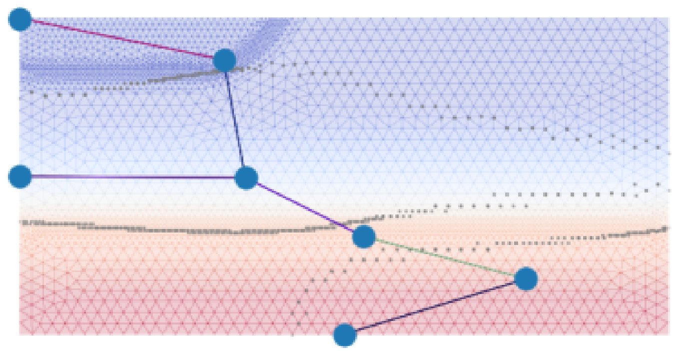




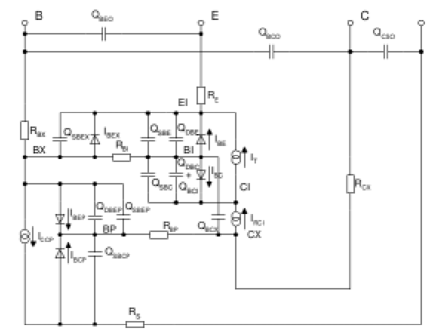
Recall DDM example: semiconductor surrogates



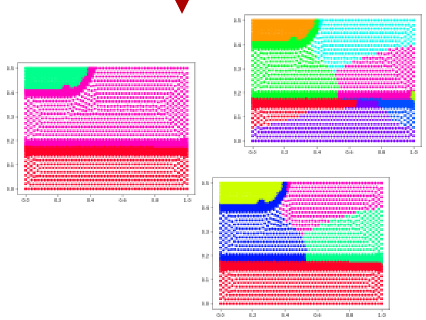
High-fidelity drift-diffusion PDE-based simulation



Learning data-driven graphical model for voltage-current relation



Result: robust surrogate embedded in production circuit simulator



Partitioning into physics-informed subdomains

- Data-driven partitioning to extract coarse partitioning of space
- Use exterior calculus ideas to fit control volume analysis to data
- **Result:** reduced order models with structure preservation + guaranteed stability properties that can **reliably be coupled to production circuit simulators**

Acknowledgements

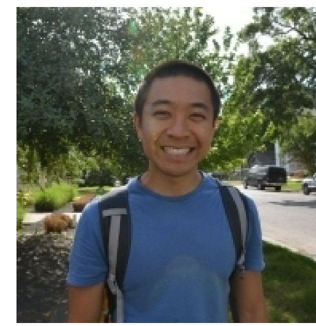
Funding sources:
- DOE early career
- Philms (ASCR MMICCs center)
- SNL LDRD
- ASC
- John von Neumann postdoc

Collaborators:
- Graph exterior calculus
  - **Xiaozhe Hu (Tufts)**
- Semiconductor work in 1300
  PIRAMID LDRD
  Huang, X. Gao, S. Reza
- Z-machine + shock physics
  Kris Beckwith, Patrick Knapp
- Combustion Research Facility
  Jackie Chen, MK Lee (8300)
- Subsurface fracture networks
  Jeffrey Hyman (LANL)



Postdocs: Ravi Patel, Mamikon Gulian, Kookjin Lee





Staff: Eric Cyr, Mitch Wood, Andy Huang

Several new projects – please contact for postdoc/collaboration opportunities (natrask@sandia.gov)

References

Highlighted publications

1. You, Huaiqian, et al. "Data-driven learning of nonlocal physics from high-fidelity synthetic data." *Computer Methods in Applied Mechanics and Engineering, AI special issue* (2021)
2. Patel, Ravi G., et al. "A physics-informed operator regression framework for extracting data-driven continuum models." *Computer Methods in Applied Mechanics and Engineering, AI special issue* (2021)
3. **Lee, Kookjin, et al. "Partition of unity networks: deep hp-approximation."** ***arXiv preprint arXiv:2101.11256*** **(2021).**
4. **Trask, Nathaniel, Andy Huang, and Xiaozhe Hu. "Enforcing exact physics in scientific machine learning: a data-driven exterior calculus on graphs."** ***arXiv preprint arXiv:2012.11799*** **(2020).**
5. Patel, Ravi G., et al. "Thermodynamically consistent physics-informed neural networks for hyperbolic systems." *arXiv preprint arXiv:2012.05343* (2020).
6. Cyr, Eric C., et al. "Robust training and initialization of deep neural networks: An adaptive basis viewpoint." *Mathematical and Scientific Machine Learning*. PMLR, (2020).
7. Patel, Ravi G., et al. "A block coordinate descent optimizer for classification problems exploiting convexity." *arXiv preprint arXiv:2006.10123* (2020).
8. Gao, Xujiao, et al. "Physics-Informed Graph Neural Network for Circuit Compact Model Development." *2020 International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*. IEEE (2020)
9. Huang, Andy, et al. "Greedy Fiedler Spectral Partitioning for Data-driven Discrete Exterior Calculus." 2021 AAAI-MLPS Conferences (under review)
10. Trask, Nathaniel, et al. "GMLS-Nets: A framework for learning from unstructured data." NeurIPs proceedings (2019)

Open source software

- GMLS-nets: learning from unstructured data through meshfree approximation (https://github.com/rgp62/gmls-net)
- MOR-Physics: Modal Operator Regression for physics discovery (https://github.com/rgp62/MOR-Physics)