This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

SAND2023-04587C

Sandia National Laboratories

Exceptional service in the national interest

Machine-Learned Finite Element Exterior Calculus for Linear and Nonlinear Problems

Jonas A. Actor, Nat Trask, Andy Huang
Postdoctoral Appointee
Computational Mathematics - 01442
Sandia National Laboratories
jaactor@snl.gov

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering 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.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & 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.

Setup: Inverse Problems

Given: **data** { high-fidelity model, expensive, compute-intensive, observational } capturing physics laws

Goal: recover **model** that describes **data** while preserving underlying physics laws

Preserving Physics in Machine Learning Models

Black Box NN

learn model through **data exclusively**

$$\min_{\xi} \left\| NN_{\xi} - u_{data} \right\|_2^2$$

Physics-Informed NN

learn model **close to physics** through penalty

$$\min_{\xi} \left\| NN_{\xi} - u_{data} \right\|_2^2 \quad + \quad \lambda \left\| \boldsymbol{L}\left[NN_{\xi}\right] - \boldsymbol{f} \right\|_2^2$$

Structure-Preserving NN

learn model that **preserves physics exactly**

$$\min_{\xi} \left\| NN_{\xi} - u_{data} \right\|_2^2 \quad \text{such that} \quad \boldsymbol{L}\left[\mathrm{NN}_{\xi}\right] = \boldsymbol{f}$$

“Learn from data”

“Solve from physics”

Why Learn a Chain Complex?

Guarantee physics constraints

Divergence-free, gauge-free constraints appear naturally in homology of chain complex

Learn new metrics

Chain complexes can be built for any manifold, with bespoke metric information

Build with mesh-free construction

Build *k*-forms using Whitney Forms

Construct with exact integration

Avoid variational crimes due to collocation, quadrature, etc.



Example of a Chain Complex

The **de Rham complex** encodes how exterior calculus relates volumes to boundary fluxes.

$$C^\infty(\Omega) \underset{\text{div}^*}{\overset{\text{grad}}{\rightleftarrows}} [C^\infty(\Omega)]^3 \underset{\text{curl}^*}{\overset{\text{curl}}{\rightleftarrows}} [C^\infty(\Omega)]^3 \underset{\text{grad}^*}{\overset{\text{div}}{\rightleftarrows}} C^\infty(\Omega)$$

Stokes' Theorem

$$\int_\Omega \nabla \cdot \boldsymbol{u} = \int_{\partial\Omega} \boldsymbol{u} \cdot \vec{n}$$

Exact Sequence Property

$$\nabla \times (\nabla u) = 0$$

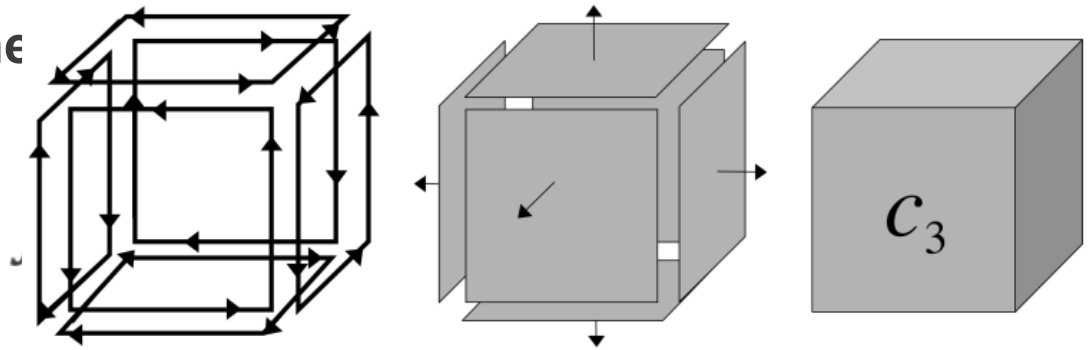
Example of a Chain Complex

The **de Rham complex** encodes how exterior calculus relates volumes to boundary fluxes.

$$C^0 \underset{\delta_0^*}{\overset{\delta_0}{\rightleftarrows}} C^1 \underset{\delta_1^*}{\overset{\delta_1}{\rightleftarrows}} C^2 \underset{\delta_2^*}{\overset{\delta_2}{\rightleftarrows}} C^3$$

- **Stokes' The** ty

$$\int_\Omega \delta_0^* \boldsymbol{u} =$$



Learning a Chain Complex

Need to define and **parameterize**:

$$C^0 \underset{\delta_0^*}{\overset{\delta_0}{\rightleftarrows}} C^1 \underset{\delta_1^*}{\overset{\delta_1}{\rightleftarrows}} C^2 \underset{\delta_2^*}{\overset{\delta_2}{\rightleftarrows}} C^3$$

- (discrete) spaces of *k*-forms $C^0, C^1, \ldots, C^k$
- differentials in chain complex $\delta_0, \delta_1, \ldots, \delta_k$

Defining *k*-forms

Build a **partition of unity** $\{\psi_i\}_{i=1,\dots,N}$ parameterized by **trainable parameters** $\{\theta_j\}_{j=1,\dots,M}$

A partition of unity (POU) maintains the following characteristics:

1. $0 \leq \psi_i(x) \leq 1$ for every point $x \in \Omega$
2. $\sum_i \psi_i = 1$

Examples: softmax, probability distributions, **B-splines**



Defining our Partition of Unity

1. Define fine-scale trainable **spline knots** $\{t_k\}_{k=1,\dots,K}$

2. Define B1-splines $\widehat{\psi_k}(x) = \begin{cases} \frac{x - t_{k-1}}{t_k - t_{k-1}} & \text{if } x \in [t_{k-1}, t_k] \\ 1 - \frac{x - t_k}{t_{k+1} - t_k} & \text{if } x \in [t_k, t_{k+1}] \\ 0 & \text{else} \end{cases}$

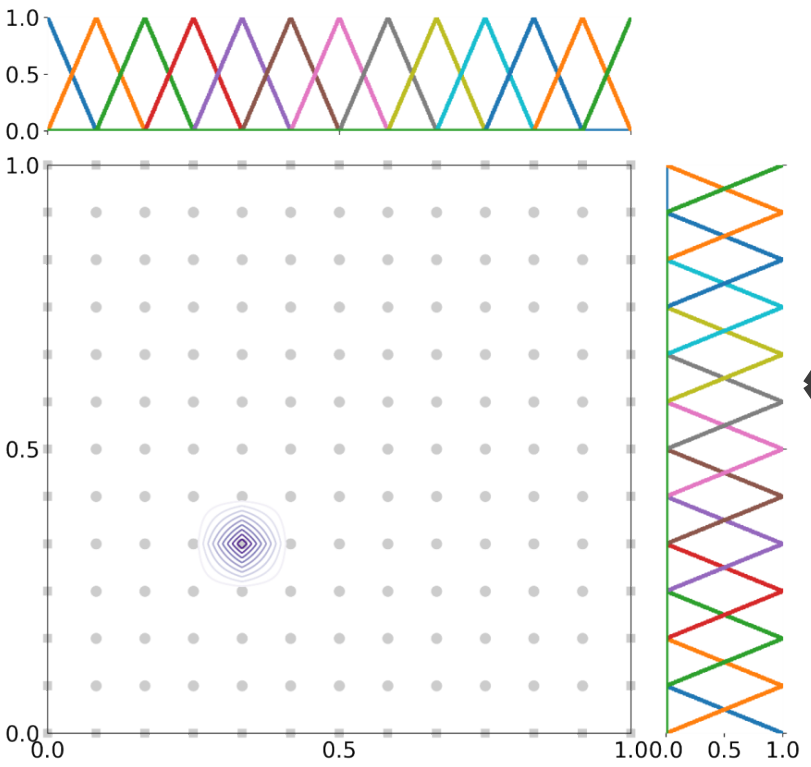
3. Define trainable **convex combination matrix** $W \in \mathbb{R}^{K \times N}$, constrained so that

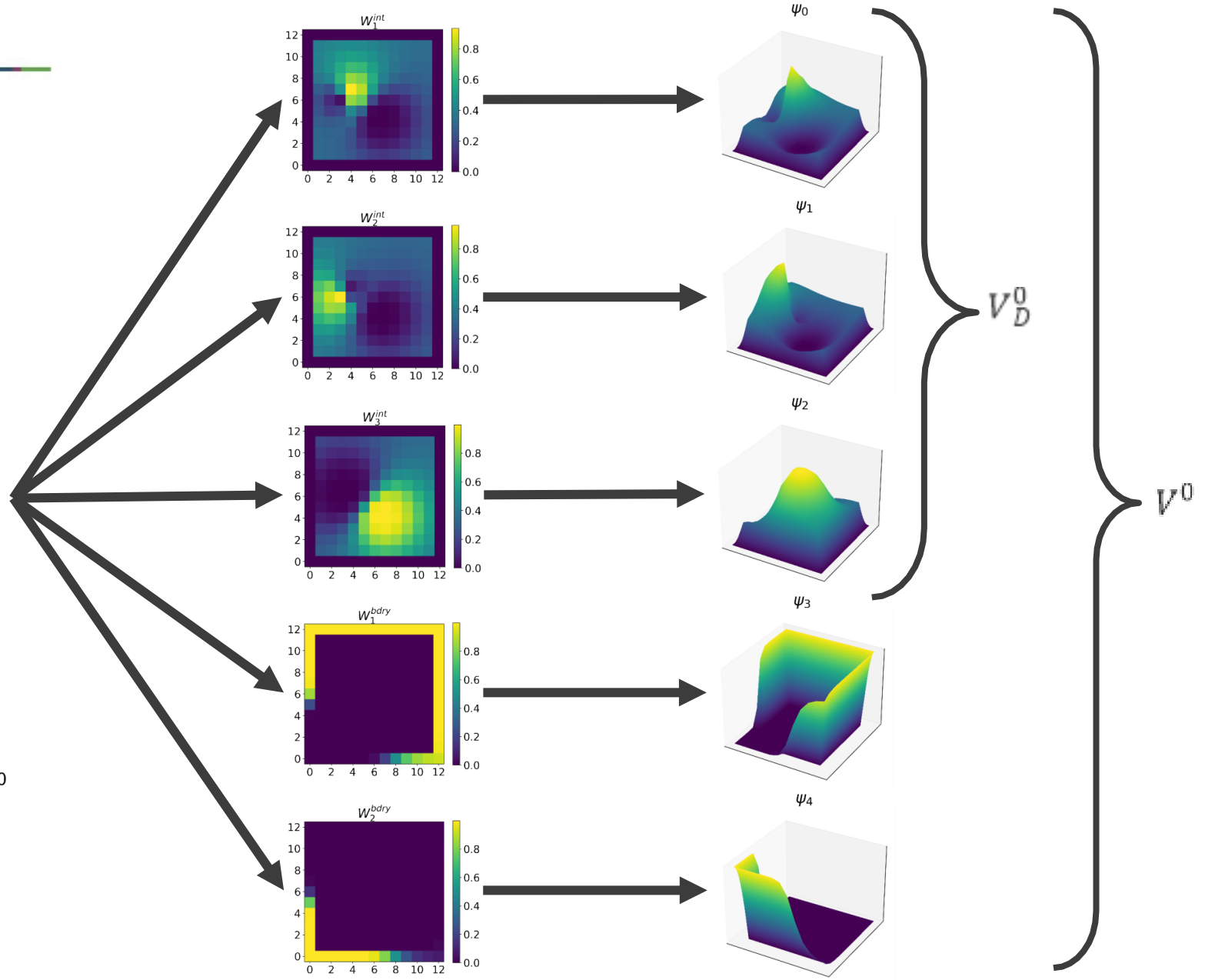
$$0 \le W_{k,n} \le 1 \quad \forall\, k, n$$

$$\sum_n W_{k,n} = 1 \quad \forall\, k$$

4. Define POU functions $\psi_i(x) = \sum_k W_{k,i}\ \widehat{\psi_k}(x)$

An Illustration

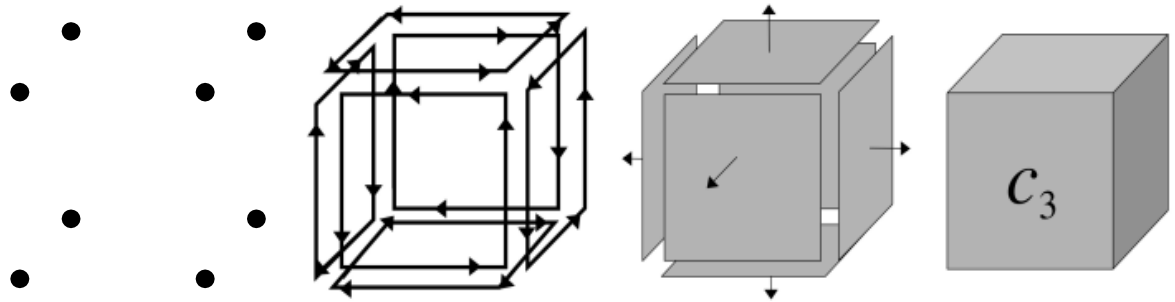




Defining Whitney *k*-forms

Define each space of *k*-forms C^k:

- $C^0 = \text{span}\,\{\psi_i\}_{i=1,\dots,N}$ = **our POUs**
- $C^1 = \text{span}\,\{\psi_{ij} = \psi_i \nabla\psi_j - \psi_j \nabla\psi_i\}_{i,j=1,\dots,N}$
- $C^k = \text{span}\,\{\psi^k_{j_0,j_1,\dots,j_k}\}_{j_0,j_1,\dots,j_k=1,\dots,N}$



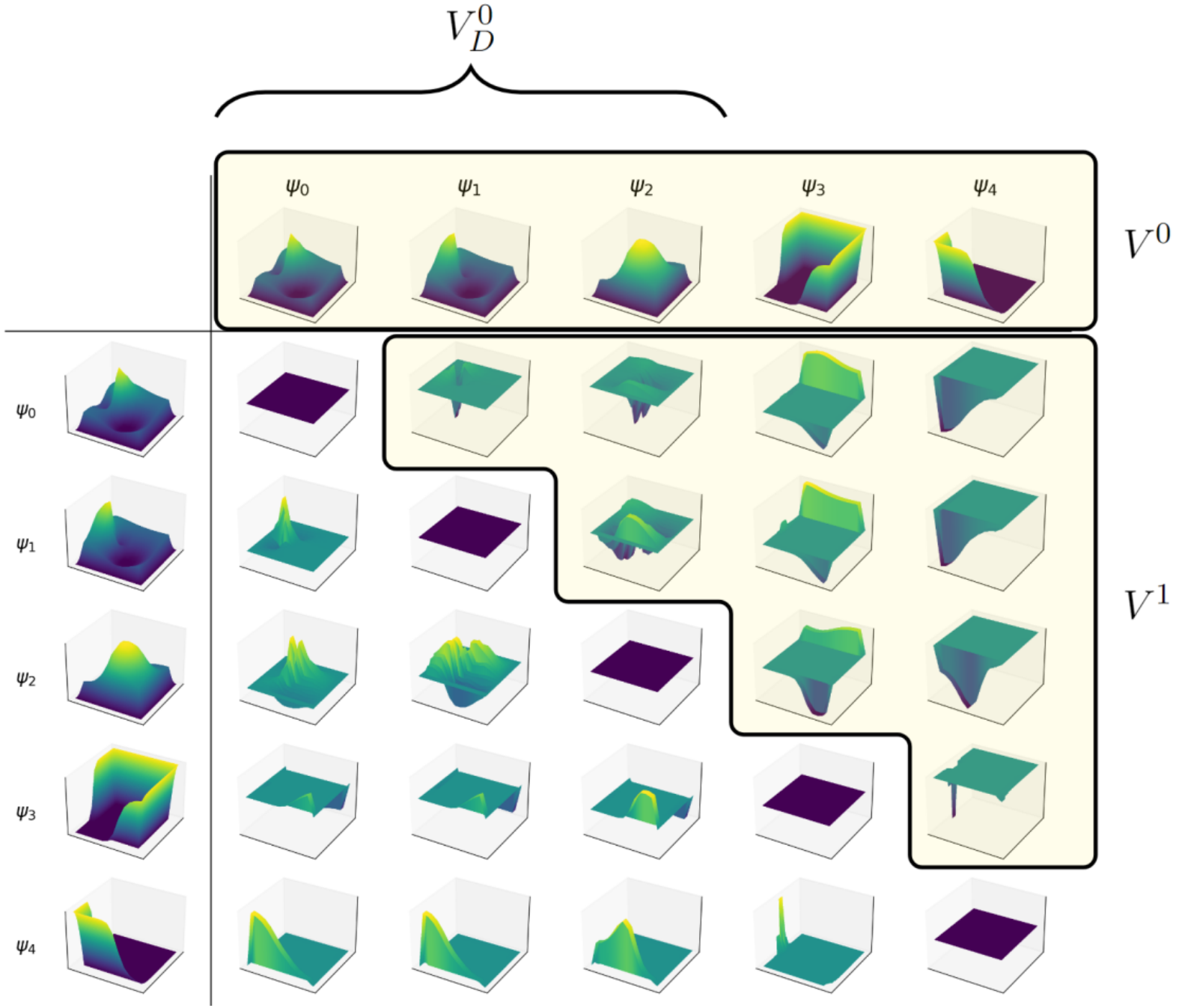
$$\psi^k_{j_0,j_1,\dots,j_k} = k! \sum_{i=0}^{k} (-1)^i \psi_{j_i}\, \mathrm{d}\psi_{j_0} \wedge \dots \wedge \mathrm{d}\psi_{j_{i-1}} \wedge \mathrm{d}\psi_{j_{i+1}} \wedge \dots \wedge \mathrm{d}\psi_{j_k}$$

By construction, $\forall \psi_i \in C^0, \;\; \nabla\psi_i \in C^1$, i.e. our spaces are **compatible**.

An Illustration

Using 0-forms from earlier,

$$\psi_{ij} = \psi_{i\nabla}\psi_j - \psi_{j\nabla}\psi_i$$



Defining Metricized Exterior Derivatives

Initial δ_k is the corresponding discrete exterior calculus operator on a complete graph

$$\mathrm{DIV}\,\psi_{ij} = \delta_0^*\psi_{ij} = \psi_j - \psi_i$$

New δ_k warps DEC operator by multiplying on left and right with positive diagonal **trainable metric tensors** B_k, D_k:

$$\begin{array}{ccccccccccc}
C^0 & \xleftarrow{\mathbf{d}_0^*} & C^1 & \xleftarrow{\mathbf{d}_1^*} & \delta_0 \longmapsto B_1\delta_0 B_0^{-1} & \xleftarrow{\mathbf{d}_3^*} & \cdots & \xleftarrow{\mathbf{d}_{d-1}^*} & C^d \\
\downarrow \mathbf{D}_0^{-1} & & \downarrow \mathbf{D}_1^{-1} & & \delta_0^* \longmapsto D_0^{-1}\delta_0^* D_1^{-1} & & & & \downarrow \mathbf{D}_d^{-1} \\
C^0 & \underset{\delta_0}{\overset{\delta_0^*}{\leftrightarrows}} & C^1 & \underset{\delta_1}{\overset{\delta_1^*}{\leftrightarrows}} & C^2 & \underset{\delta_2}{\overset{\delta_2^*}{\leftrightarrows}} & C^3 & \underset{\delta_3}{\overset{\delta_3^*}{\leftrightarrows}} & \cdots & \underset{\delta_{d-1}}{\overset{\delta_0^*}{\leftrightarrows}} & C^d \\
\downarrow \mathbf{B}_0 & & \downarrow \mathbf{B}_1 & & \downarrow \mathbf{B}_2 & & \downarrow \mathbf{B}_3 & & & & \downarrow \mathbf{B}_d \\
C^0 & \xrightarrow{\mathbf{d}_0} & C^1 & \xrightarrow{\mathbf{d}_1} & C^2 & \xrightarrow{\mathbf{d}_2} & C^3 & \xrightarrow{\mathbf{d}_3} & \cdots & \xrightarrow{\mathbf{d}_{d-1}} & C^d
\end{array}$$

Building Exact Physics FEEC System

Strong form

Find $(p, \boldsymbol{F}) \in C^0 \times C^1$ such that

$$
\begin{aligned}
\boldsymbol{F} - \nabla p &= \boldsymbol{0} && \text{in } \Omega \\
\nabla \cdot \boldsymbol{F} &= f && \text{in } \Omega \\
\boldsymbol{F} \cdot \overrightarrow{n} &= g_N && \text{on } \Gamma_N \\
p &= g_D && \text{on } \Gamma_D
\end{aligned}
$$

Finite dimensional variational problem

Find $p = \sum_i p_i \psi_i$ and $\boldsymbol{F} = \sum_{ij} F_{ij} \psi_{ij}$ such that $\forall\ \psi_a \in C_0^0,\ \psi_{ab} \in C^1$,

$$
\begin{aligned}
(\boldsymbol{F}, \psi_{ab}) - (\nabla p, \psi_{ab}) &= (\nabla g_D, \psi_{ab}) \\
-(\boldsymbol{F}, \nabla \psi_a) &= (f, q) + (g_N, q)_{\Gamma_N}
\end{aligned}
$$

Building Exact Physics FEEC System

Discrete System
Solve the linear system $\begin{bmatrix} M & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} F \\ p \end{bmatrix} = \begin{bmatrix} L \\ R \end{bmatrix}$ valent system $\begin{bmatrix} M^{-1} & \delta_0 \\ \delta_0^* & 0 \end{bmatrix} \begin{bmatrix} MF \\ p \end{bmatrix} = \begin{bmatrix} M^{-1}L \\ R \end{bmatrix}$

Introduce metrics
Change metrics for exterior derivatives

$$\delta_0 \longmapsto B_1 \delta_0 B_0^{-1}$$
$$\delta_0^* \longmapsto D_0^{-1} \delta_0^* D_1$$

Learnable system
Solve the new linear system $\begin{bmatrix} M^{-1} & B_1 \delta_0 B_0^{-1} \\ D_0^{-1} \delta_0^* D_1 & 0 \end{bmatrix} \begin{bmatrix} MF \\ p \end{bmatrix} = \begin{bmatrix} M^{-1}L \\ R \end{bmatrix}$



Building Exact Physics FEEC System

Machine learning FEEC problem

Solve the optimization problem

$$\min_{\substack{\text{POU parameters } \boldsymbol{\theta}, \\ \text{metrics } D_0, D_1, B_0, B_1}} \left\| p_{\text{data}} - \sum_i p_i \psi_i \right\|_2^2 + \left\| F_{\text{data}} - \sum_{ij} F_{ij} \psi_{ij} \right\|_2^2$$

$$\text{such that} \quad \begin{bmatrix} M^{-1} & B_1 \delta_0 B_0^{-1} \\ D_0^{-1} \delta_0^* D_1 & 0 \end{bmatrix} \begin{bmatrix} MF \\ p \end{bmatrix} = \begin{bmatrix} M^{-1} L \\ R \end{bmatrix}$$

Structure-Preserving NN learn model that **preserves physics exactly**

$$\min_{\xi} \left\| NN_\xi - u_{data} \right\|_2^2 \quad \text{such that} \quad \boldsymbol{L}[\mathrm{NN}_\xi] = \boldsymbol{f}$$





Examples

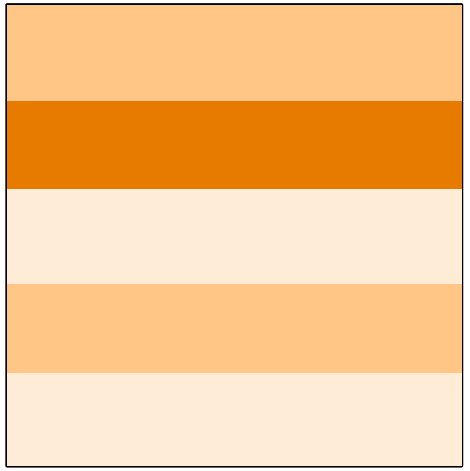
Examples 1&2: Mixed-Formulation Poisson Problems

Problem Statement

Find $(p, \boldsymbol{F}) \in C^0 \times C^1$ such that

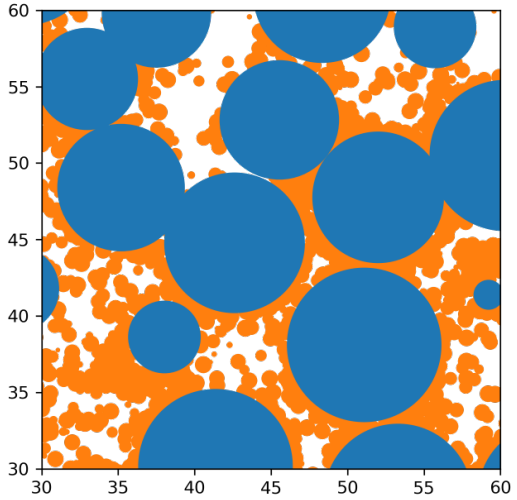
$$
\begin{aligned}
\boldsymbol{F} - \nabla p &= \boldsymbol{0} \quad \text{in } \Omega \\
\nabla \cdot (\kappa \boldsymbol{F}) &= f \quad \text{in } \Omega \\
\boldsymbol{F} \cdot \vec{n} &= g_N \quad \text{on } \Gamma_N \\
p &= g_D \quad \text{on } \Gamma_D
\end{aligned}
$$

Problem 1
Five Strip Problem



$$
\begin{aligned}
f &= 0 \\
\Gamma_N &= \partial\Omega \\
g_N &= \begin{cases} -1 & x = 0 \\ 1 & x = 1 \\ 0 & y = 0,1 \end{cases}
\end{aligned}
$$

Problem 2
Battery Problem



$$
\begin{aligned}
f &= 0 \\
\Gamma_D &= \partial\Omega \\
g_D &= 1 - x
\end{aligned}
$$

Battery data courtesy of Scott Roberts, 01513

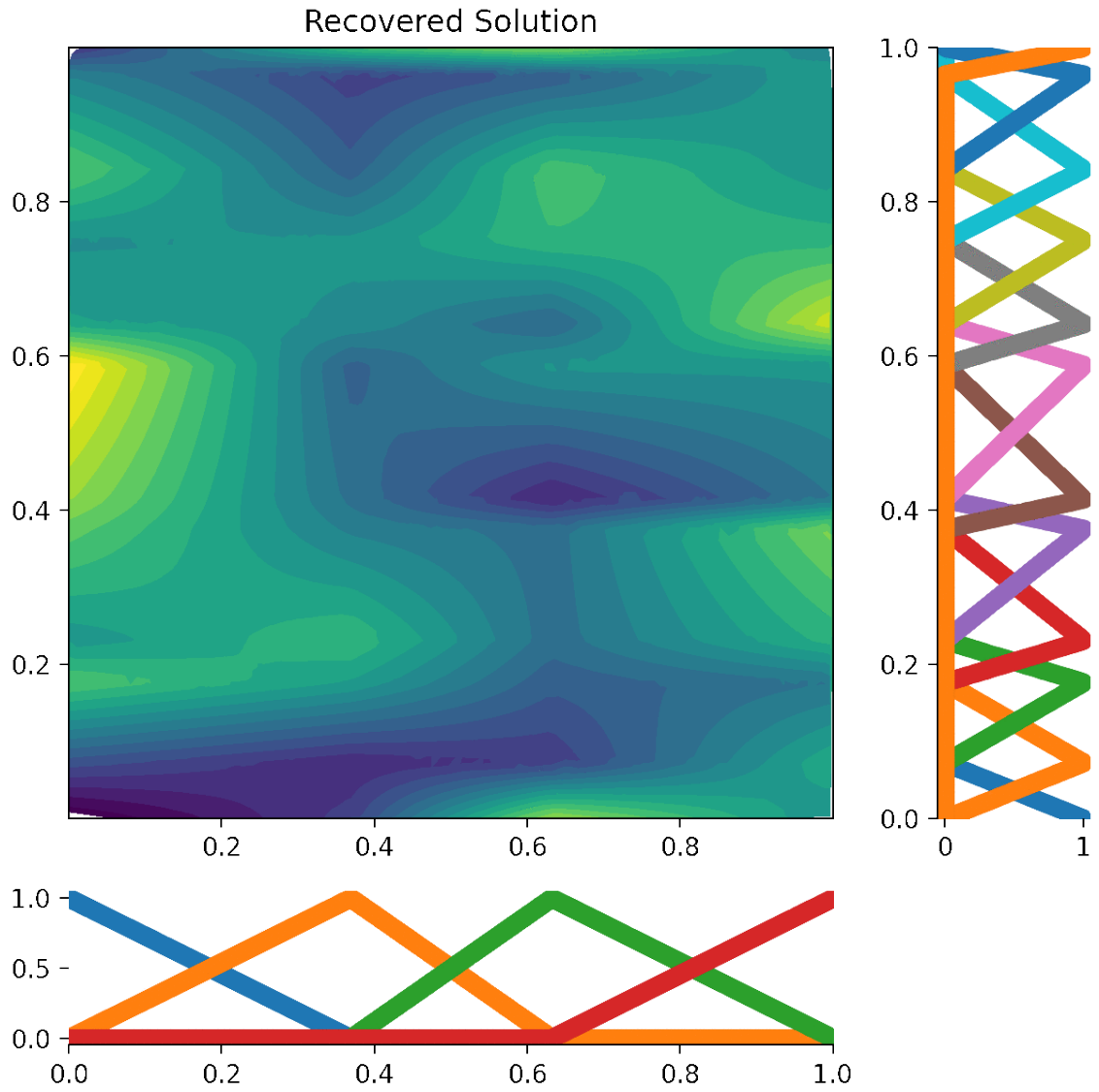
Example 1: Five Strip Problem

True Solution

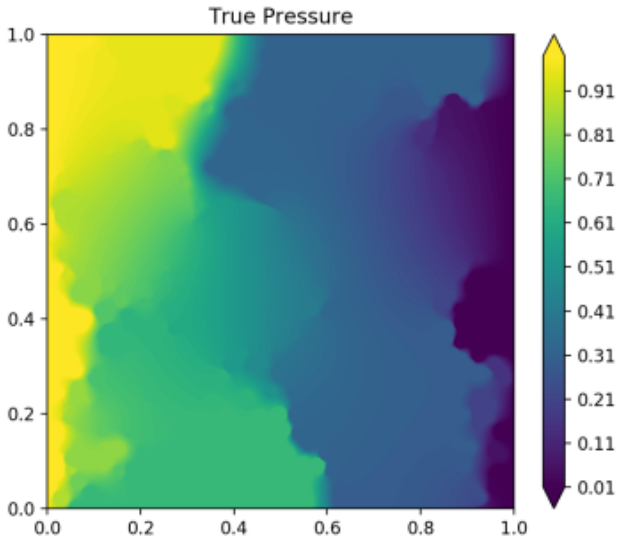
$p(x, y) = \beta_i \, x$ for each strip i

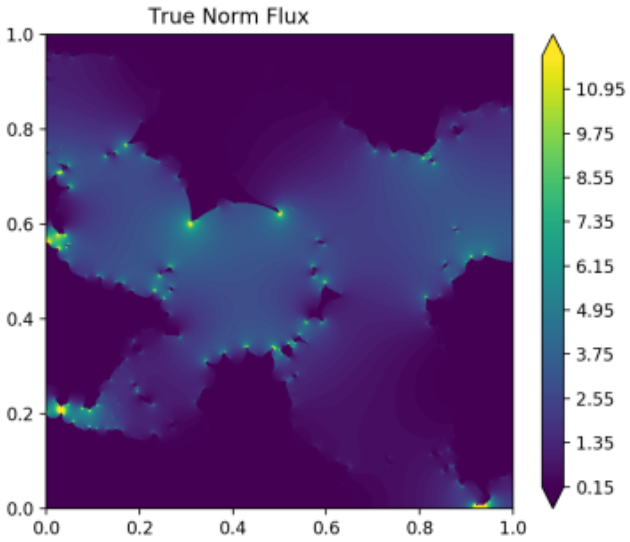
$\boldsymbol{F}(x, y) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

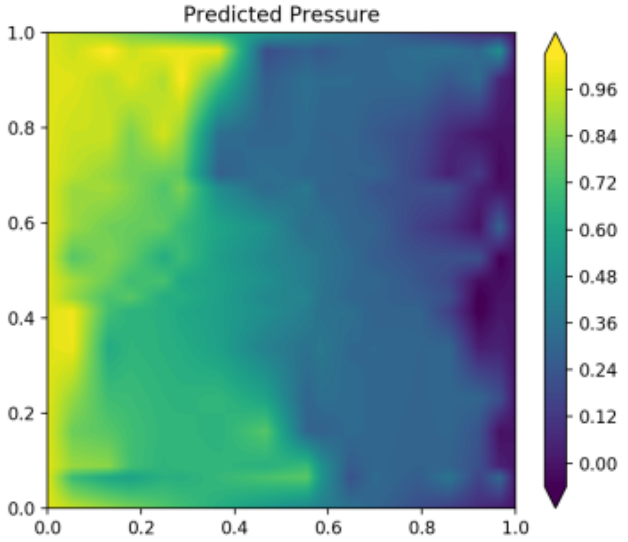
Generate p_{data}, F_{data} from true solution

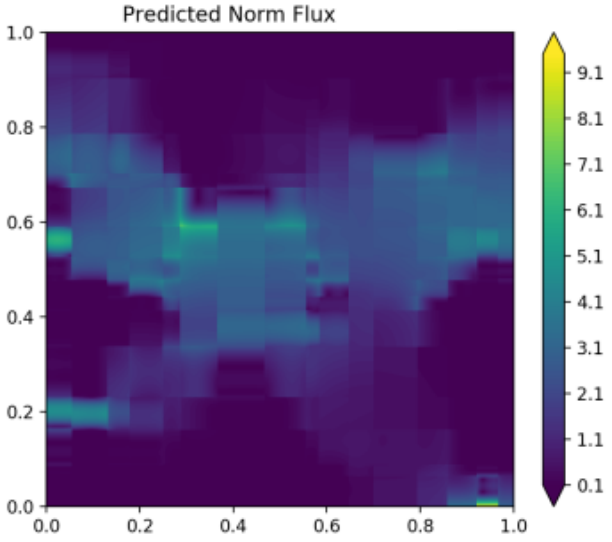


Example 2: Battery Problem









FEM simulation with ARIA

Mesh: 5.89 M nodes
11.8 M elements
Solve: 5.89 M DoF

ML chain complex model

POUs: 8 interior POUs
8 boundary POUs
Solve: 264 DoF

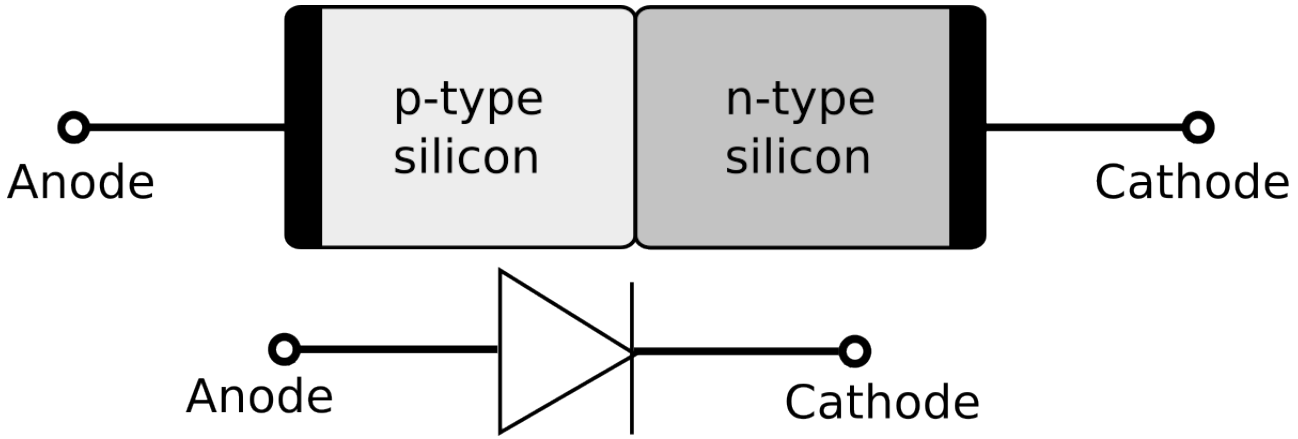
Example 3: Drift-Diffusion Equations

Problem Statement

Find $(\phi, n, p, \boldsymbol{J}_\phi, \boldsymbol{J}_n, \boldsymbol{J}_p) \in (C^0)^3 \times (C^1)^3$ such that

$$
\begin{aligned}
\boldsymbol{J} - \nabla \phi &= \boldsymbol{0} && \text{in } \Omega \\
\boldsymbol{J}_n + \nabla n - n \nabla \phi &= \boldsymbol{0} && \text{in } \Omega \\
\boldsymbol{J}_p + \nabla p + p \nabla \phi &= \boldsymbol{0} && \text{in } \Omega \\
-\nabla \cdot \boldsymbol{J} + n - p &= f && \text{in } \Omega \\
-\nabla \cdot \boldsymbol{J}_n + R(n,p) &= 0 && \text{in } \Omega \\
-\nabla \cdot \boldsymbol{J}_p - R(n,p) &= 0 && \text{in } \Omega
\end{aligned}
$$

where $R(n,p) = \frac{np - \alpha_0}{\beta_0(n+p)+\beta_1}$



ML Formulation

Machine learning FEEC problem
Solve the **nonlinear-constrained** optimization problem

$$\min_{\text{POU parameters } \boldsymbol{\theta},\ \text{metrics } \boldsymbol{D}\text{s},\ \boldsymbol{B}\text{s}} \left\| \phi_{\text{data}} - \sum_i \phi_i \psi_i \right\|_2^2 + \left\| n_{\text{data}} - \sum_i n_i \psi_i \right\|_2^2 + \left\| p_{\text{data}} - \sum_i p_i \psi_i \right\|_2^2 +$$

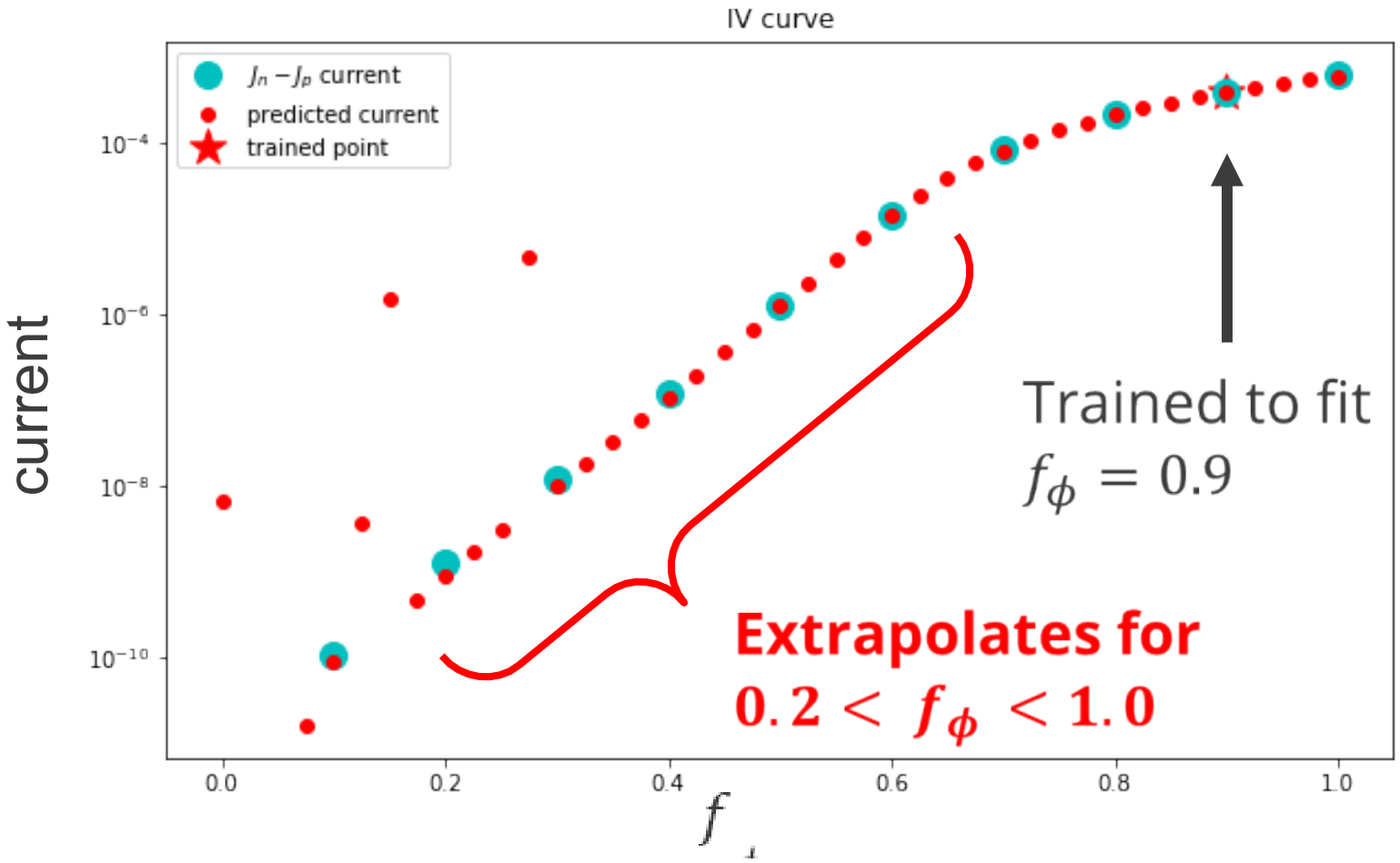
$$\left\| \boldsymbol{J}_{\phi_{\text{data}}} - \sum_{ij} \boldsymbol{J}_{\phi_{ij}} \psi_{ij} \right\|_2^2 + \left\| \boldsymbol{J}_{n_{\text{data}}} - \sum_{ij} \boldsymbol{J}_{n_{ij}} \psi_{ij} \right\|_2^2 + \left\| \boldsymbol{J}_{p_{\text{data}}} - \sum_{ij} \boldsymbol{J}_{p_{ij}} \psi_{ij} \right\|_2^2$$

$$\text{such that} \quad \boldsymbol{F}\left(\boldsymbol{\phi}, n, p, \boldsymbol{J}_{\boldsymbol{\phi}}, \boldsymbol{J}_n, \boldsymbol{J}_p\right) = \boldsymbol{0}$$

Example 3: Drift-Diffusion Equations

- Solution changes nonlinearly as we **vary f_ϕ voltage** across the transistor
- **Measure current** $(J_n - J_p) \cdot \vec{n}$ at boundary

Because model captures the nonlinearity exactly, we can extrapolate across a range of voltages!



Conclusion: Structure Preservation By Construction

Structure-preserving ML by learning a chain complex preserves physics invariants while learning parameters, coefficients, metric information.

Learning a chain complex guarantees:

- Generalized Stokes's Theorem
- Compatibility: $\delta_{k+1}\psi_i \in C^{k+1}$
- Exactness of the chain complex: $\delta_{k+1} \circ \delta_k = 0$

Our POU construction guarantees:

- Exact integration: no quadrature necessary, avoid variational crimes
- Adaptive parameterization: training focuses on areas that need refinement within data

References and Further Reading

References

- Actor, JA., et al. “Data-Driven Whitney Forms for Structure-Preserving Control Volume Analysis.” *Journal of Computational Physics,* in revision.
- Trask, N., Huang, A., Hu, X. "Enforcing exact physics in scientific machine learning: a data-driven exterior calculus on graphs." *Journal of Computational Physics*: 110969, 2022.
- Actor, JA., Huang, A., Trask, N. "Polynomial-Spline Neural Networks with Exact Integrals.“ *2022 IEEE SSCI,* 2022.
- Huang, A., et al. "Greedy Fiedler Spectral Partitioning for Data-driven Discrete Exterior Calculus." *Proceedings of the AAAI 2021 Spring Symposium on Combining Artificial Intelligence and Machine Learning with Physical Sciences. AAAI*, 2021.

Further Reading

- Adler, JH., et al. "A finite-element framework for a mimetic finite-difference discretization of Maxwell's equations." *SISC* 43.4: A2638-A2659, 2021.
- Arnold, D., Falk, RS., Winther, R. “Finite element exterior calculus, homological techniques, and applications”*, Acta Numerica:* 1-155, 2006.