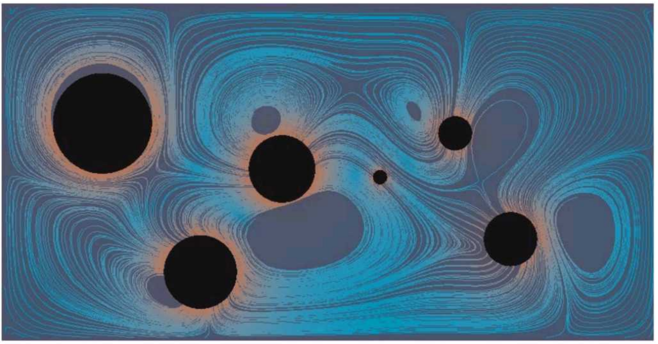
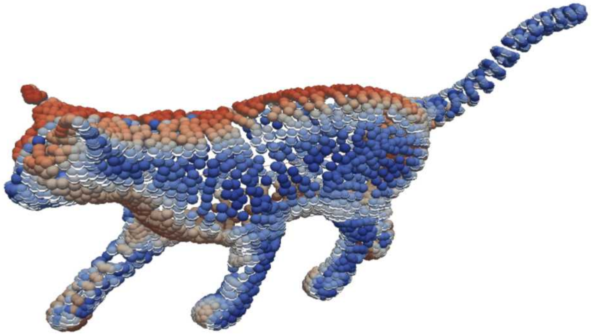
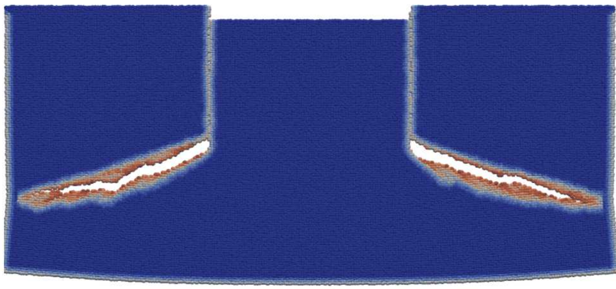
SAND2019-14555PE







# Compatible meshfree discretization

Nat Trask
Center for Computing Research
Sandia National Laboratories

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.

# What does meshfree mean?

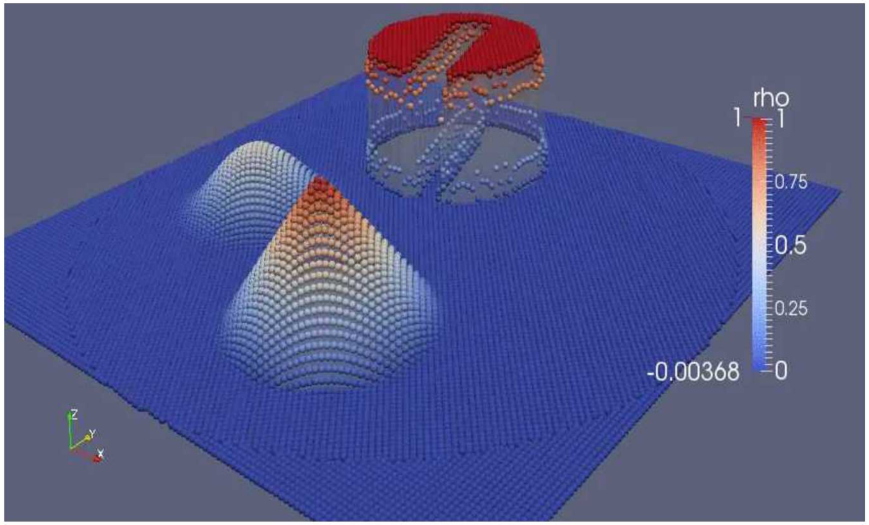
- Physics compatible FEM spaces defined via differential k-forms:
  - For a polygonal mesh in 3D

**Zero-form:** $\delta_{\mathbf{x}_i} \circ \mathbf{u}$

**One-form:** $\int_E \mathbf{u} \cdot d\mathbf{l}$

**Two-form:** $\int_F \mathbf{u} \cdot d\mathbf{A}$

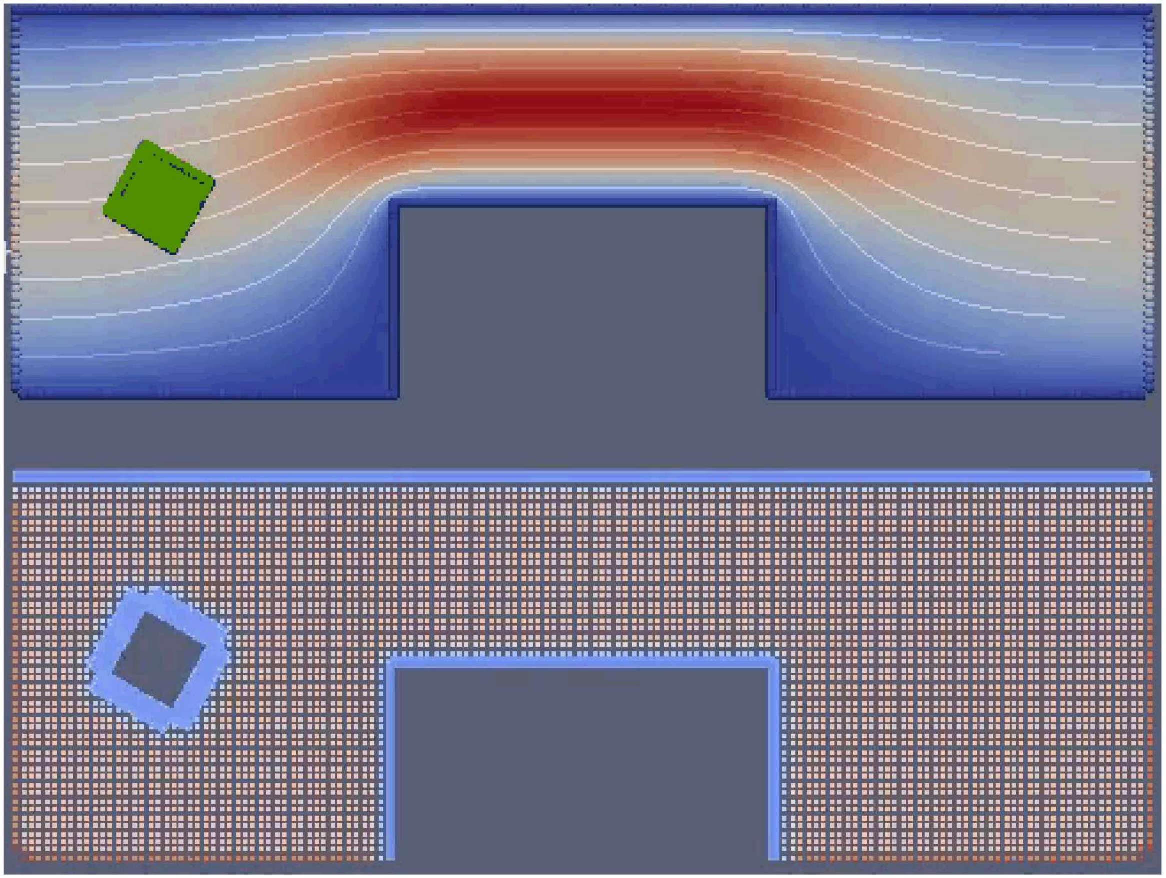
**Three-form:** $\int_C \mathbf{u} dV$




**A meshfree method uses only zero-forms as degrees of freedom**

- Easy to push points around if you don't care about preserving a mesh
- Exchange nice mathematical setting to get more descriptive models
  - No Stokes theorems (no conservation), no natural bilinear forms

# Why meshfree? Large deformation problems



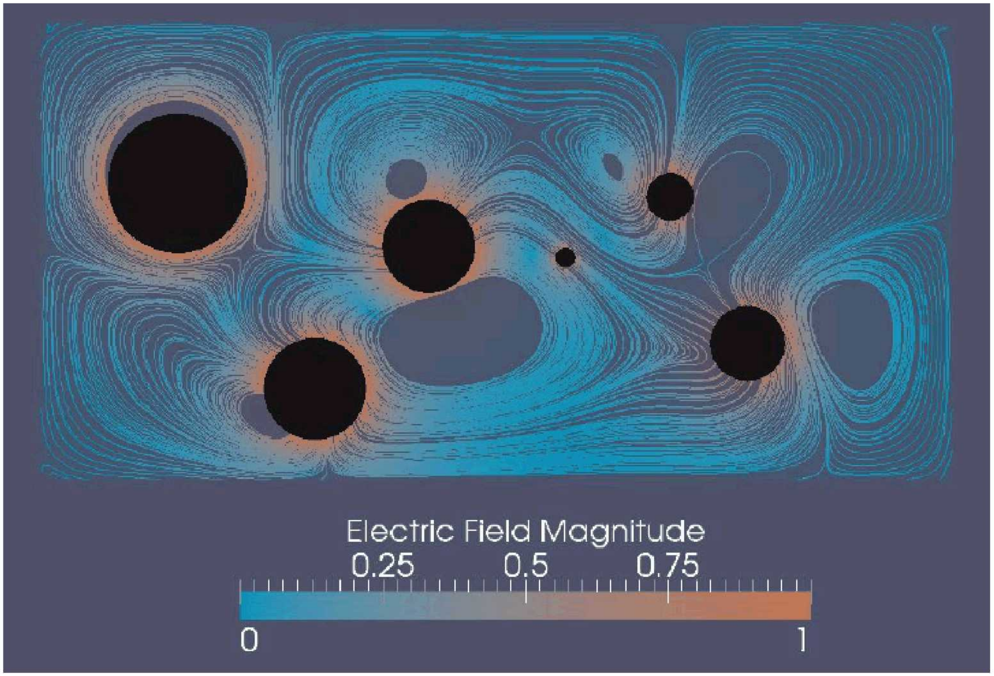
$$\begin{cases} -\nabla^2 \mathbf{u} + \nabla p = \mathbf{f} \\ \nabla \cdot \mathbf{u} = 0 \\ \mathbf{u}|_{\partial\omega} = \mathbf{U} + (x - \mathbf{X}) \times \Omega \\ \int_{\partial\omega} \sigma \cdot d\mathbf{A} = 0 \end{cases}$$

Trask, N., Maxey, M., Hu, X.
A compatible high-order meshless method for the Stokes equations with applications to suspension flows
*Journal of Computational Physics* (2018)

Hu, W., Trask, N., Hu, X.,Pan,W.
A spatially adaptive high-order meshless method for fluid–structure interactions.
*Computer Methods in Applied Mechanics and Engineering* (2019)

Trivial treatment of large deformation problems – no remeshing + remap

# Why meshfree? Large deformation problems




$$\begin{cases} -\nu\nabla^2\mathbf{u} + \nabla p = -\rho_e(\phi)\nabla\phi \\ \nabla\cdot\mathbf{u} = 0 \\ \mathbf{u} = \mathbf{w} \\ \mathbf{u} = \mathbf{V}_i + (x - \mathbf{X}_i)\times\mathbf{\Omega}_i \end{cases}$$

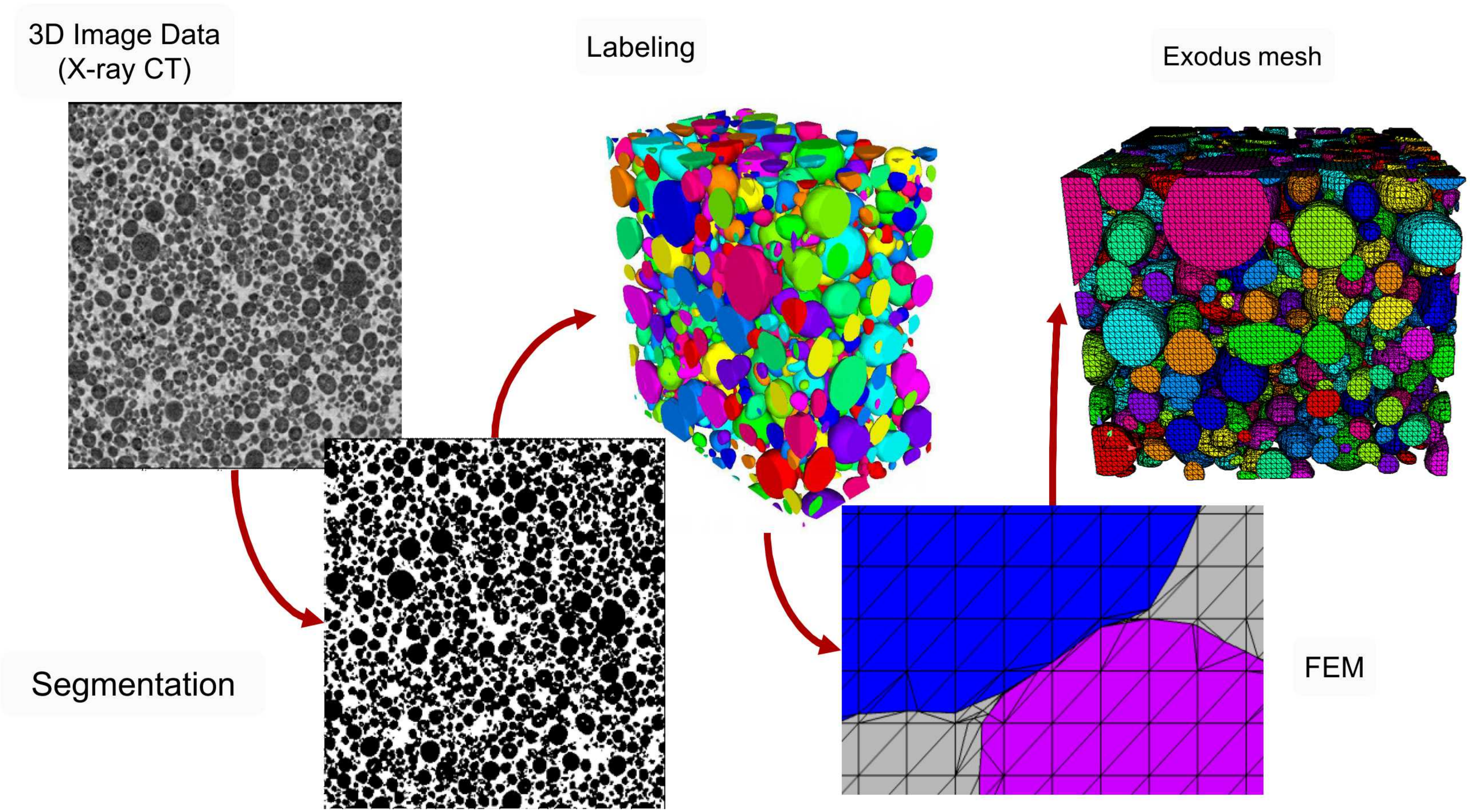
$$-l_c^2\nabla^4\phi + \nabla^2\phi = -\frac{\rho_e(\phi)}{\epsilon}$$

$$\begin{cases} 0 = \int_{\partial\Omega_i}\overline{\overline{\sigma}}\cdot d\mathbf{A} \\ 0 = \int_{\partial\Omega_i}\overline{\overline{\sigma}}\times(x - \mathbf{X}_i)\cdot d\mathbf{A} \end{cases}$$

$$\overline{\overline{\sigma}} = -\epsilon_0\left(\mathbf{E}\otimes\mathbf{E} + E^2\mathbf{I}\right) + -p\mathbf{I} + \frac{\nu}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^\mathsf{T}\right)$$

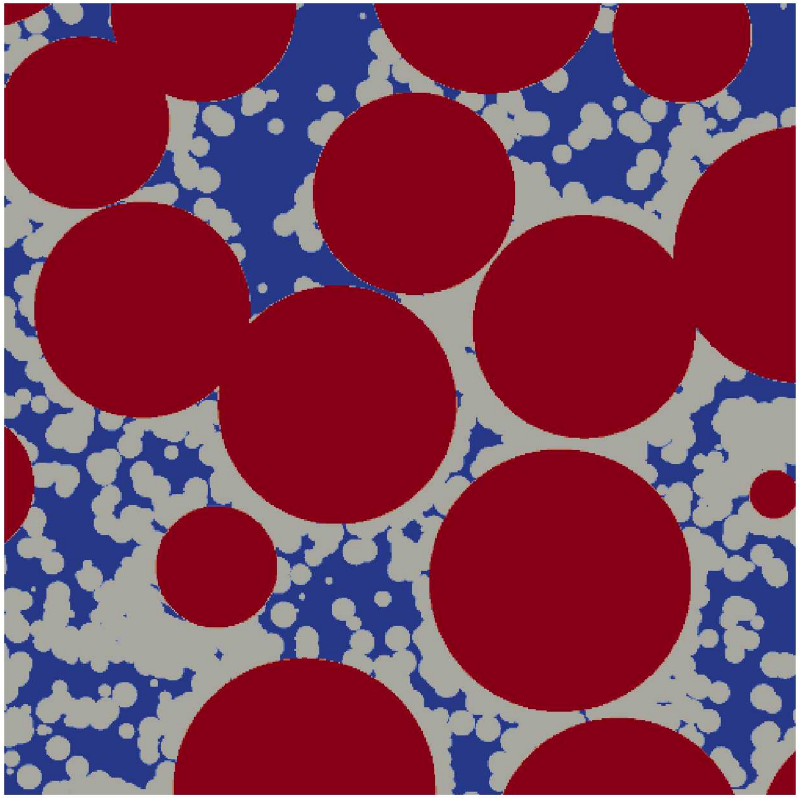
**Compatible meshfree discretization:** A framework for physics compatible discretization of multiphysics problems that mimics robustness of mimetic methods

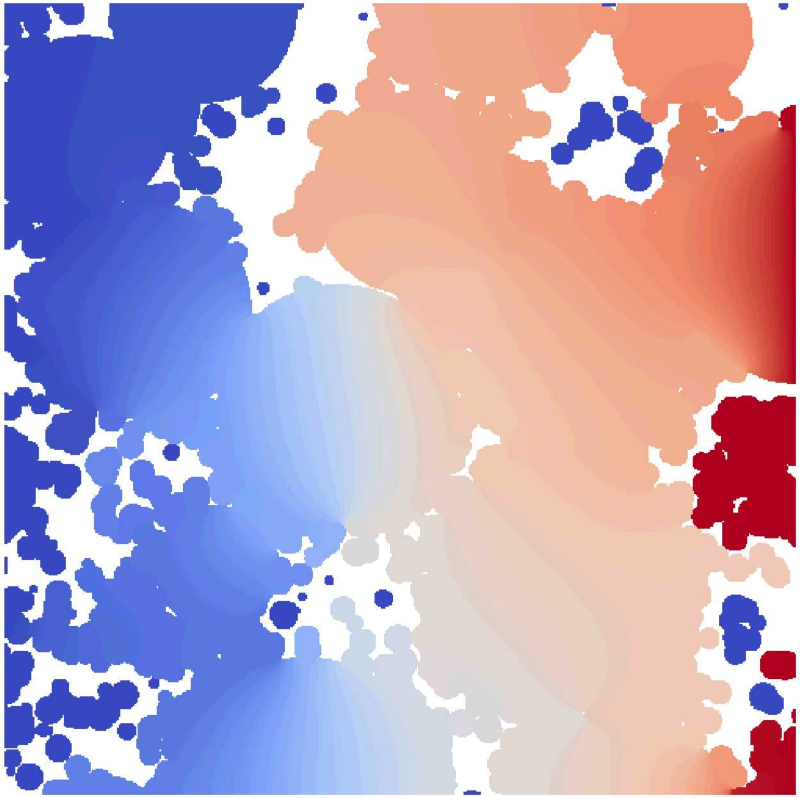
# Why meshfree? Automatic geometry discretization




- For engineering problems **meshing constitutes 60-70% of time to solution** (SAND-2005-4647), which cannot be improved by moving to larger computers
  - Automating geometry discretization is fundamental to developing large throughput workflows based on either experimental data or UQ/optimization

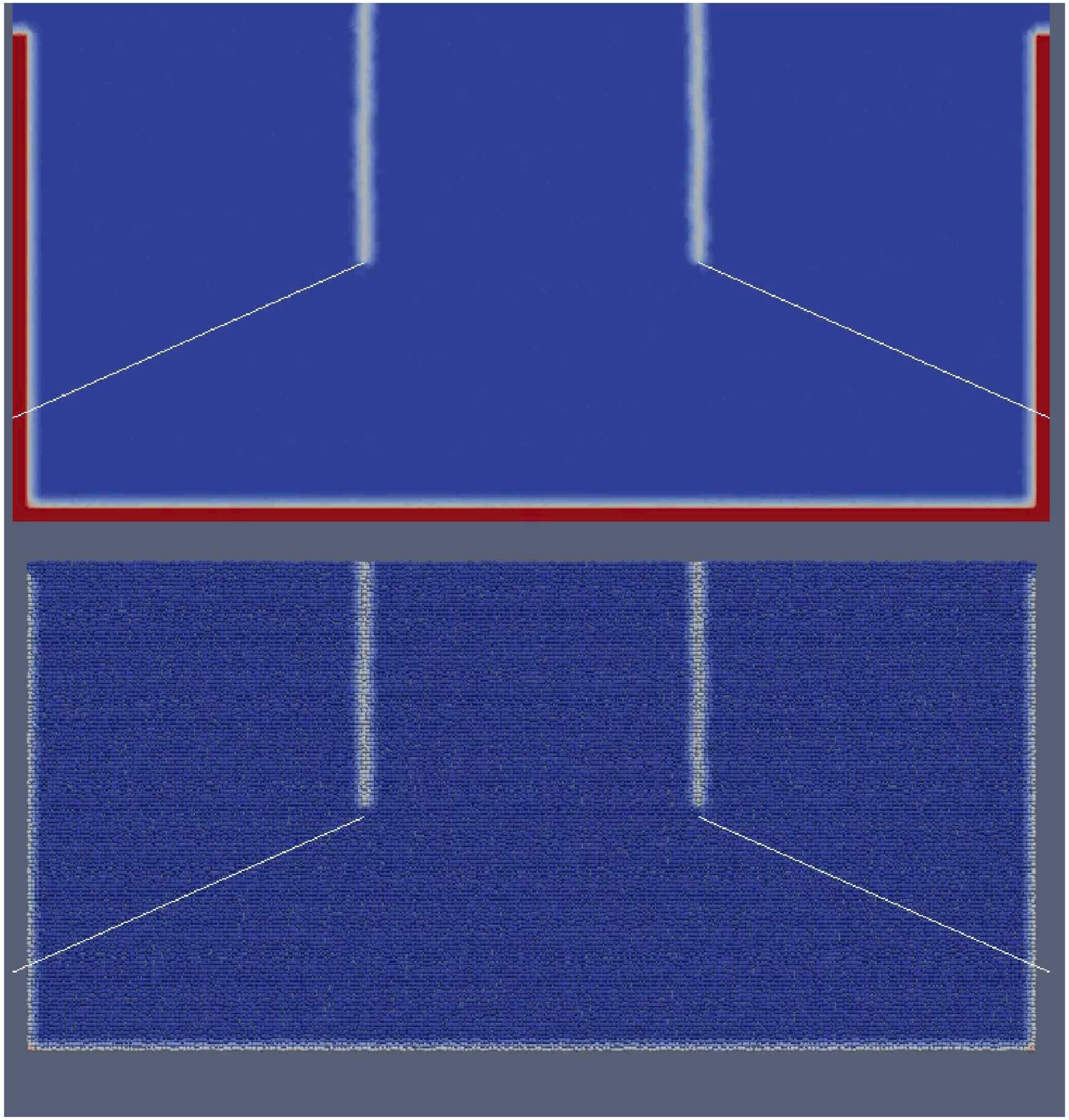
# Why meshfree? Automatic geometry discretization





- Meshfree methods operate **directly on the degrees of freedom available in experiment**
- Placing a particle at each voxel of the CT scan is sufficient to obtain a high-fidelity simulation without human-in-the-loop meshing process

# Why meshfree? Fracture mechanics

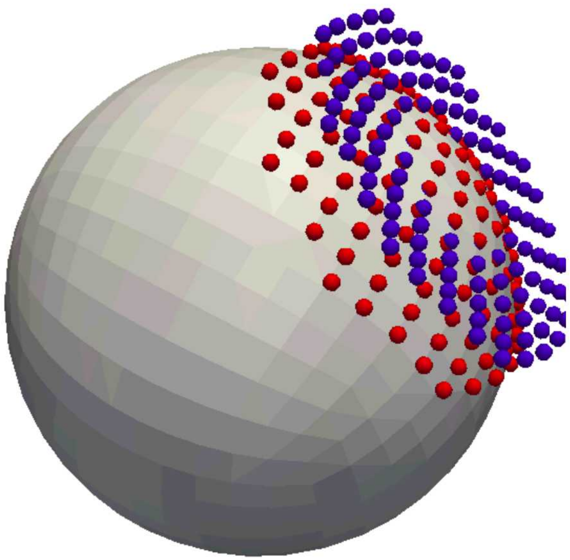


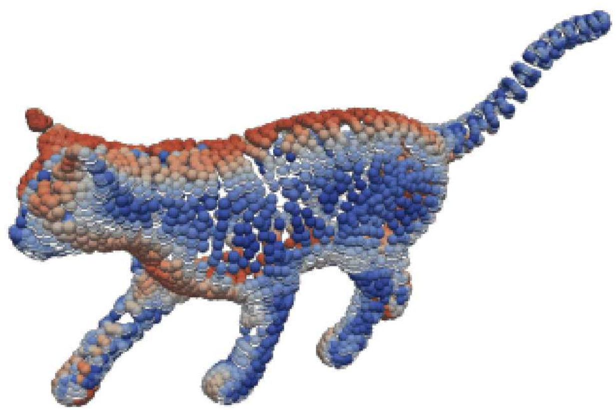
**Automatic treatment of topology changes:**
No need to reconnect elements, manage mesh quality, etc. as topology evolves as a function of solution

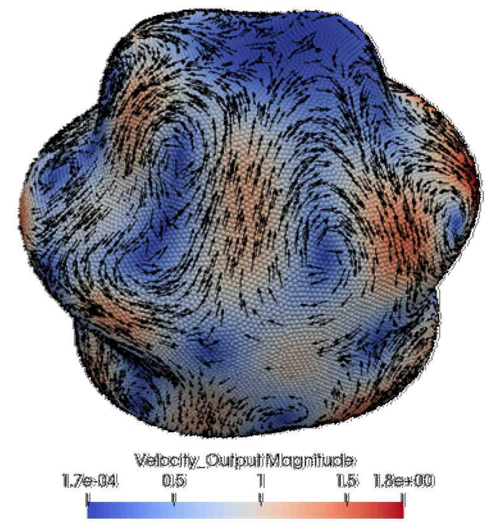
Trask, N., et al.
"An asymptotically compatible meshfree quadrature rule for nonlocal problems with applications to peridynamics."
*Computer Methods in Applied Mechanics and Engineering*
343 (2019): 151-165.

# Why meshfree? Differential geometry on evolving manifolds

To solve surface PDE, one may learn mapping between local charts and tangent space to access metric tensor, curvature, surface differential operators, etc.








$$\begin{cases} \mu_m \left(-\boldsymbol{\delta} \mathbf{d} \mathbf{v}^\flat + 2K\mathbf{v}^\flat\right) - \gamma \mathbf{v}^\flat - \mathbf{d}p & = & -\mathbf{b}^\flat \\ -\boldsymbol{\delta} \mathbf{v}^\flat & = & 0. \end{cases}$$

Trask, Nathaniel, and Paul Kuberry. "Compatible meshfree discretization of surface PDEs."*Computational Particle Mechanics*(2019): 1-7.
Gross, B. J., Trask, N., Kuberry, P.,&Atzberger, P. J. (2019).Meshfree Methods on Manifolds for Hydrodynamic Flows on Curved Surfaces:A Generalized Moving Least-Squares (GMLS) Approach.*arXiv preprint arXiv:1905.10469*.

# Outline

- Generalized moving least squares (GMLS)
  - An approximation theory framework for generating meshfree methods with rigorous accuracy guarantees
- Conservative meshless discretization
  - How can we construct conservative schemes if we don't have access to discrete Stokes theorems?
- Meshfree discretizations of nonlocal mechanics
  - Can we construct a meshfree discretization framework for integral operators for fracture mechanics?
- Meshfree machine learning
  - For scientific machine learning applications, can we use scattered data approximation theory to build learning frameworks appropriate for unstructured scientific data?

# Generalized moving least squares (GMLS)

Given $u \in V$, a framework for estimating operators $\tau \in V^*$ by finding an optimal reconstruction over a subspace $P \subset V$ which best matches unstructured samples $\Lambda := \{\lambda_i(u)\}_i$

$$\tau(u) \approx \tau^h(u)$$

$$p^* = \underset{p \in P}{\operatorname{argmin}} \left( \sum_j \lambda_j(p) - \lambda_j(u) \right)^2 W(\tau, \lambda_j)$$

$$\tau^h(u) := \tau(p^*)$$

**Example:**

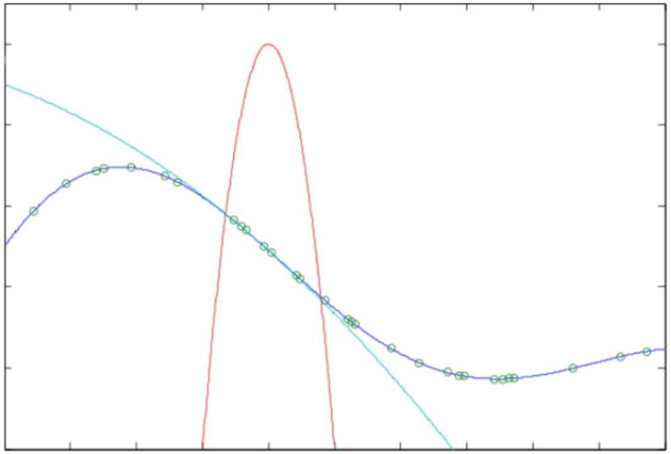
Approximate point evaluation of derivatives:

Target functional $\tau_i = D^\alpha \circ \delta_{x_i} \in V^*$

Reconstruction space $P = \pi_m$

Sampling functional $\lambda_j = \delta_{x_j} \in V^*$

Weighting function $W = W(||x_i - x_j||)$



# Generalized moving least squares (GMLS)

**Primal problem:**
Unconstrained QP

$$p^* = \underset{p \in P}{\operatorname{argmin}} \left( \sum_j \lambda_j(p) - \lambda_j(u) \right)^2 W(\tau, \lambda_j)$$

$$\tau^h(u) := \tau(p^*)$$

**Dual problem:**
Equality constrained QP

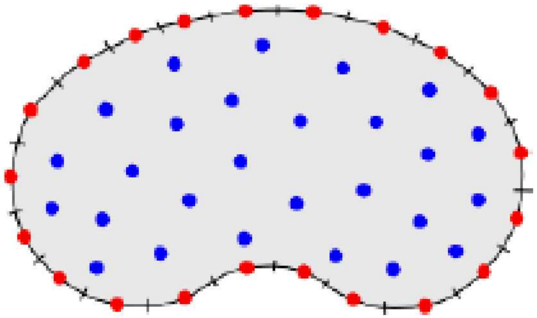
$$\tau^h(u) = \sum_j \alpha_j \lambda_j(u)$$

$$\underset{\alpha}{\operatorname{argmin}} \sum_j \alpha_j^2 \frac{1}{W(\tau, \lambda_j)}$$

$$\text{such that } \tau^h(p) = \sum_j \alpha_j \lambda_j(p) \quad \forall p \in P$$

# Preliminaries: Quasi-uniform point clouds

Sandia National Laboratories



**Definition 0.1. Fill+separation distances** Given point cloud $X = \{x_1, \ldots, x_N\} \subset \Omega$, define distances

$$h_X = \sup_{x \in \Omega} \min_{j \in X} ||x - x_j||^2$$

$$q_X = \frac{1}{2} \min_{j \neq i} ||x_i - x_j||^2$$

**Definition 0.2. Quasi-uniformity** A point cloud X is *quasi-uniform with respect to* $c_{qu}$ if

$$q_X \leq h_X \leq c_{qu} q_X$$

**Proposition 0.1.** *Suppose bounded* $\Omega$ *and quasi-uniform* $X$ *w.r.t.* $c_{qu} > 0$. *Then there exist* $c_1, c_2 > 0$ *such that*

$$c_1 N^{-\frac{1}{d}} \leq h_X \leq c_2 N^{-\frac{1}{d}}$$

# How do error estimates typically work in GMLS?

Consider first approximation of function from point samples using a polynomial reconstruction

# Classical MLS: quasi-interpolants [Wendland04]

**Definition 1.** ***Local polynomial reproduction:*** *A process defining* $\forall x_i \in X$ *an approximation* $s_u(x) = \sum_j \alpha_j u(x_j)$ *is a local polynomial reproduction if there exist* $C_1, C2 > 0$.

1. $\sum_j \alpha_j p_j = p(x) \quad \forall p \in P$
2. $\sum_j |\alpha_j| \leq C_1 \quad \forall x \in \Omega$
3. $\alpha_j(x) = 0$ *if* $||x - x_j||_2 > C_2 h_X$ *and* $x \in \Omega$

**Theorem 1.** *For bounded* $\Omega$*, define* $\Omega^* = \bigcup_{x \in \Omega} B(x, C_2 h_X)$. *If* $s_u$ *is a local polynomial reproduction of order* $m$ *and* $u \in C^{m+1}(\Omega^*)$ *then*

$$|u(x) - s_u(x)| \leq C h_X^{m+1} |u|_{C^{m+1}(\Omega^*)}$$

**Theorem 2.** *Consider the GMLS process with* $\tau = \delta_x$, $\lambda_j(u) = u(x_j)$, *and* $P = \Pi_m$. *If* $\Omega$ *is compact and satisfies a cone condition, and* $X$ *is quasi-uniform, then there exists a constant* $C > 0$ *such that* $supp(W) = C\, h_X$ *where the GMLS problem is solvable and forms a local polynomial reproduction.*

# Polynomial reproduction error estimate

*Proof.* Let $p \in \Pi_m$

$$
\begin{aligned}
|f(x) - s_f(x)| &\leq |f(x) - p(x)| + \left| p(x) - \sum_j \alpha_j f_j \right| \\
&\leq |f(x) - p(x)| + \sum_j |\alpha_j| \, |p_j - f_j| \\
&\leq ||f - p||_{L_\infty(B(x, C_2 h_X))} \left( 1 + \sum_j |\alpha_j| \right) \\
&\leq (1 + C_1) ||f - p||_{L_\infty(B(x, C_2 h_X))}
\end{aligned}
$$

□

# Classical MLS: derivative approximation [Mirzaei12]

**Definition 0.1. Local polynomial reproduction:** A process defining $\forall x_i \in X$ an approximation $D_h^\alpha u(x) = \sum_j \alpha_j u(x_j)$ is a local polynomial reproduction if there exist $C_1, C2 > 0$.

1. $\sum_j \alpha_j p_j = D^\alpha p(x) \quad \forall p \in V$
2. $\sum_j |\alpha_j| \leq C_1 h_X^{-|\alpha|} \quad \forall x \in \Omega$
3. $\alpha_j(x) = 0$ if $||x - x_j||_2 > C_2 h_X$ and $x \in \Omega$

**Theorem 0.1.** *For bounded $\Omega$, define $\Omega^* = \underset{x\in\Omega}{\cup} B(x, C_2 h_X)$. If $D_h^\alpha$ is a local polynomial reproduction of order $m$ and $u \in C^{m+1}(\Omega^*)$ then*

$$|D^\alpha u(x) - D_h^\alpha u(x)| \leq C h_X^{m+1-|\alpha|} |u|_{C^{m+1}(\Omega^*)}$$

**Theorem 0.2.** *Consider the GMLS process with $\tau(u) = D^\alpha u(x)$, $\lambda_j(u) = u(x_j)$, and $V = \Pi_m$. If $\Omega$ is compact and satisfies a cone condition, and $X$ is quasi-uniform, then there exists a constant $C > 0$ such that supp$(W) = C\, h_X$ where the GMLS problem is solvable and forms a local polynomial reproduction.*

# An abstract error analysis framework

**Basic technique:**

$$
\begin{aligned}
|\tau_{\mathbf{x}}(u) - \tau^h_{\mathbf{x}}(u)| &\leq |\tau_{\mathbf{x}}(u) - \tau_{\mathbf{x}}(p)| + |\tau_{\mathbf{x}}(p) - \tau^h_{\mathbf{x}}(u)|, \quad (\forall p \in P) \\
&\leq |\tau_{\mathbf{x}}(u) - \tau_{\mathbf{x}}(p)| + |\tau^h_{\mathbf{x}}(p - u)|, \quad \longleftarrow \text{reconstruction property} \\
&\leq |\tau_{\mathbf{x}}(u - p)| + \left| \sum_{i=1}^{N_p} \lambda_i(u - p)\, a^i_{\tau_{\mathbf{x}}} \right| \quad \longleftarrow \text{GMLS definition} \\
&\leq |\tau_{\mathbf{x}}(u - p)| + \max_{i \in I_{\mathbf{x}}} |\lambda_i(u - p)| \sum_{i \in I_{\mathbf{x}}} |a^i_{\tau_{\mathbf{x}}}|.
\end{aligned}
$$

$$\sum_{i \in I_{\mathbf{x}}} |a^i_{\tau_{\mathbf{x}}}| \leq C_W \|\tau_{\mathbf{x}}\|_{P^*} \|\Lambda^{-1}_{\mathbf{x}}\|$$

**Holds for any target functional and approximation space:**

$$|\tau_{\mathbf{x}}(u) - \tau^h_{\mathbf{x}}(u)| \leq |\tau_{\mathbf{x}}(u - p)| + C_W \|\tau_{\mathbf{x}}\|_{P^*} \|\Lambda^{-1}_{\mathbf{x}}\| \max_{i \in I_{\mathbf{x}}} |\lambda_i(u - p)|, \quad p \in P$$

# Broadly applicable toolbox for meshfree approximation

- All examples from beginning of talk fall into this framework
  - Ex: Data transfer applications

$$\lambda_i^e(\mathbf{u}) := \frac{1}{|e_i|}\int_{e_i} \mathbf{u}\cdot\mathbf{t}_i \qquad \lambda_i^f(\mathbf{u}) = \frac{1}{|f_i|}\int_{f_i} \mathbf{u}\cdot\mathbf{n}_i \qquad \lambda_i^v(u) := \frac{1}{|V_i|}\int_{V_i} u(\boldsymbol{y})\,d\boldsymbol{y}$$

  - Ex: Solving different PDES

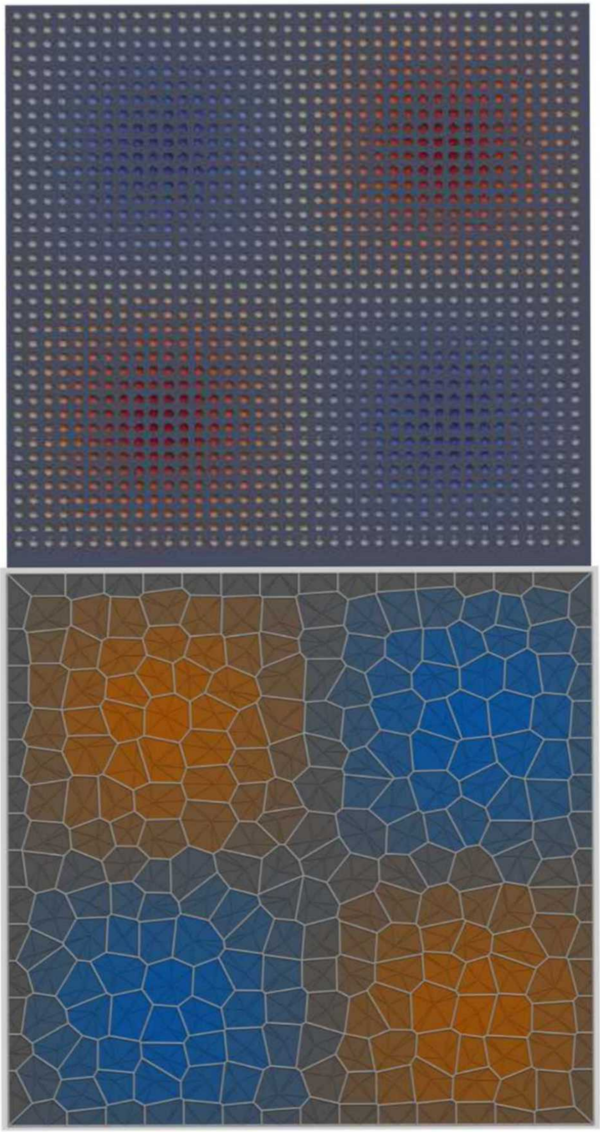
$$\tau(u) = div(u) \qquad \tau(u) = \int_{B(x)} K(x,y)u(y) - u(x)dy \qquad \tau(u) = \int_{\partial\Omega} \sigma(u)\cdot d\mathbf{A}$$

  - Ex: Handling divergence/curl constraints in saddle point problems

$$V_h = \left\{\mathbf{v} \in (\Pi_m)^d \middle| \quad \nabla\cdot\mathbf{v} = 0\right\}$$

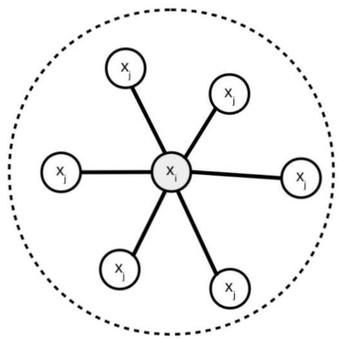
$$V_h = \left\{\mathbf{v} \in (\Pi_m)^d \middle| \quad \nabla\times\mathbf{v} = 0\right\}$$

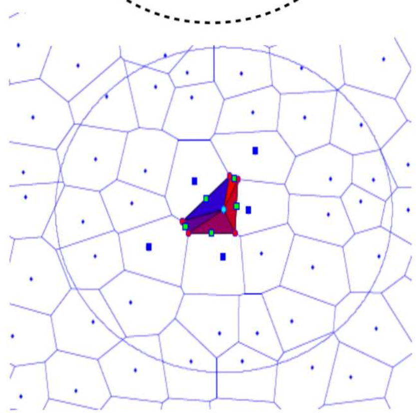
# Solving PDEs with or without a mesh

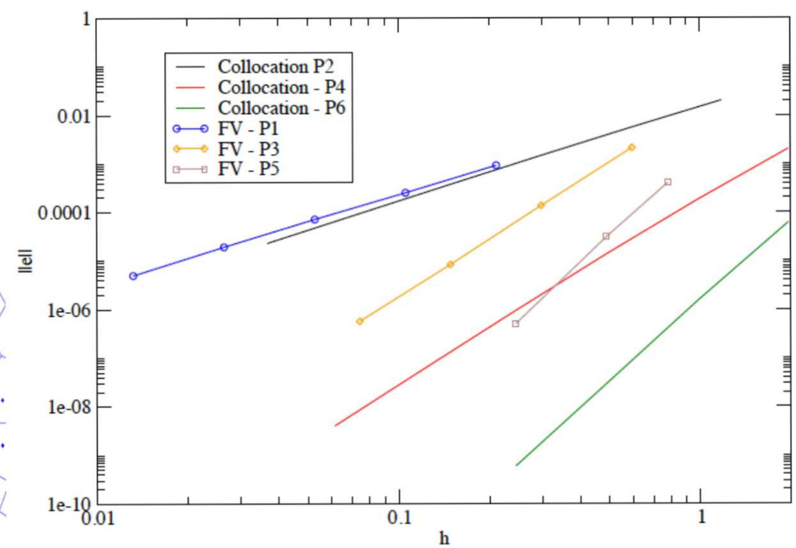


To generate mesh free schemes for $\nabla^2\phi = f$:

| | | Finite difference | Finite volume |
|---|---|---|---|
| Target functional | $\tau_i$ | $\nabla^2\phi(\boldsymbol{x}_i)$ | $\int_{face} \nabla\phi \cdot d\boldsymbol{A}$ |
| Reconstruction space | $\mathbf{V}$ | $P_m$ | $P_m$ |
| Sampling functional | $\lambda_j$ | $\phi(\boldsymbol{x}_j)$ | $\phi(\boldsymbol{x}_j)$ |
| Weighting function | $W$ | $W(\Vert\boldsymbol{x}_j - \boldsymbol{x}_i\Vert)$ | $W(\Vert\boldsymbol{x}_j - \boldsymbol{x}_i\Vert)$ |



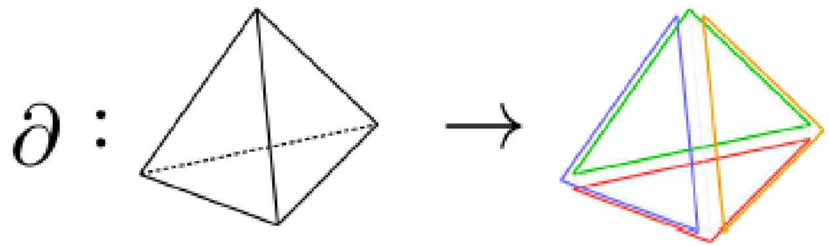





# Why is conservation hard in meshfree?

**Generalized Stokes theorem:**

$$\int_{\Omega} d\omega = \int_{\partial\Omega} \omega$$



**Gauss divergence theorem:**

$$\int_{C} \nabla \cdot \mathbf{u}\, dx = \int_{F \in \partial C} \mathbf{u} \cdot d\mathbf{A}$$

**Ingredients to cook up a conservative discretization:**

- A chain complex – **a good boundary operator**
- A consistent coboundary operator **– good function approximation**
- A measure on each mesh entity

# Quadrature with GMLS

Assume a basis, $\forall p \in \mathbf{V}$, $p = \mathbf{c}^\intercal \mathbf{P}$ and rewrite GMLS problem as

$$c^* = \underset{c \in \mathbb{R}^{dim(\mathbf{V})}}{\arg\min} \frac{1}{2} \sum_{j=1}^{N} \left(\lambda_j(u) - c^* \lambda_j(\mathbf{P})\right)^2 \omega(\tau; \lambda_j).$$

$$\tau(u) \approx c^* \tau(P^*)$$

**Ex:** Selecting $\tau = \int_c u \, \mathrm{dx}$, and defining the vector

$$\mathbf{v}_c = \int_c \mathbf{P} dx$$

we can see that a quadrature functionals may be represented as a pairing of the GMLS reconstruction coefficient vector with some vector in its dual space

$$I_c[u] = \mathbf{v}_c^\intercal \mathbf{c}^*$$

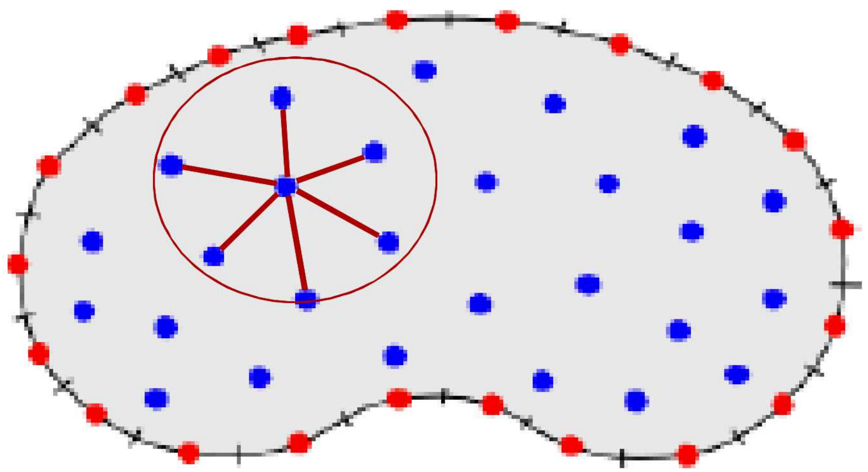
We seek to similarly define *meshfree quadrature functionals* with summation by parts properties.

# Mesh surrogate: $\varepsilon$-ball graph

**Construct a discrete divergence theorem of form:**

$$I_{c_i}[\nabla \cdot \mathbf{F}] = \sum_{j \in N_i^\epsilon} I_{f_{ij}}[\mathbf{F}] + \mathbb{1}_{i \in \partial\Omega} \int_{\partial\Omega_i} \mathbf{F} \cdot dA$$

$$I_{f_{ij}}[\mathbf{F}] = -I_{f_{ji}}[\mathbf{F}]$$



**Antisymmetry in fluxes lead to cancelation of telescoping sum, providing a global conservation statement:**

$$\sum_i I_{c_i}[\nabla \cdot \mathbf{F}] = \sum_{\substack{i \\ j \in \partial\Omega^c}} I_{f_{ij}}[\mathbf{F}] + \sum_{i \in \partial\Omega} \int_{\partial\Omega_i} \mathbf{F} \cdot d\mathbf{A}$$

$$= \int_{\partial\Omega} \mathbf{F} \cdot d\mathbf{A}$$

# Virtual divergence theorem construction

Let $\mathbf{F} \in P_1(\Omega)^d$. We assume the following ansatz for our *virtual divergence theorem.*

$$\mu_i \, (div\, \mathbf{F})_i = \sum_{j,\beta} \mu_{ij}^{\intercal} \mathbf{c}_{ij}(\mathbf{F}) + BC$$

where

- $\mu_i$ and $\mu_{ij} = -\mu_{ij}$ are measures **to be determined** corresponding to virtual volumes and face areas
- $\mathbf{c}_{ij}(\mathbf{F})$ is a vector of coefficients associated with the GMLS reconstruction of $\mathbf{F}$ over $P_1(\Omega)^d$ at virtual face $f_{ij}$
- Seek to enforce local polynomial reproduction property, so that for VDT is exact when applied to $\boldsymbol{F} \in P$

# How to get the areas?

Assume virtual areas $\mu_{ij}^{\alpha}$ may be expressed in terms of a *virtual area potential* multiplied by point evaluation of basis function at virtual face

$$\mu_{ij}^{\alpha} = \left(\psi_j^{\alpha} - \psi_i^{\alpha}\right) \phi^{\alpha}(\boldsymbol{x}_{ij})$$

Then we obtain the following **weighted graph Laplacian** problem for each basis function

$$\sum_j \left(\psi_j^{\alpha} - \psi_i^{\alpha}\right) \phi^{\alpha}(\boldsymbol{x}_{ij}) = \mu_i div\, \phi_i^{\alpha}$$

Following Fredholm alternative, this has solution if

- $\sum_i \mu_i = \mu(\Omega)$
- $\mu_i > 0$

**O(N) solution using black-box AMG**

# How to get the volumes?

Assumed we have a process for generating volumes satisfying

- $\sum \mu_i = \mu(\Omega)$
- $\mu_i > 0$

Solve the following equality constrained QP

$$\mu^* = \underset{\mu}{argmin} \sum_i \mu_i^2 \omega_i$$

$$\text{such that } \sum_i \mu_i = \mu(\Omega)$$

For any weight $\omega_i > 0$, this provides the definition

$$\mu_i = \frac{\omega_i}{\sum_k \omega_k} \mu(\Omega)$$

**Theorem.** *Assume a quasi-uniform pointset $X$ and compactly supported $\{\omega_i\}$. Then there exist $C_1, C_2 > 0$ satisfying*

$$C_1 h^d \leq \mu_i \leq C_2 h^d$$

# How to get area scaling?

- A work in progress…

**Sketch:**
Graph-Laplacian admits variational description

$$a(u, v) = b(v)$$

Given a Poincare-like inequality (Fiedler eigenvalue problem)

$$||u||_2^2 \leq C\, N^{\alpha} a(u, u)^2$$

Then we can control the energy norm with standard Lax-Milgram estimates

$$N^{-\alpha} ||u||_2^2 \leq a(u, u)^2 \leq ||b||_2 ||u||_2$$

which lets us control the virtual area by the forcing in the graph Laplacian problems

$$|u_j - u_i| \leq \sqrt{a(u, u)} \leq N^{\frac{\alpha}{2}} ||b||_2$$

# Results: singularly perturbed advection-diffusion

Consider conservation laws for conserved variable q

$$\partial_t q + \nabla \cdot \mathbf{F} = 0$$

Where we will assume steady state and the following fluxes:

- **Darcy:**

$$\boldsymbol{F} = -\mu \nabla \phi$$

- **Singularly perturbed advection diffusion:**

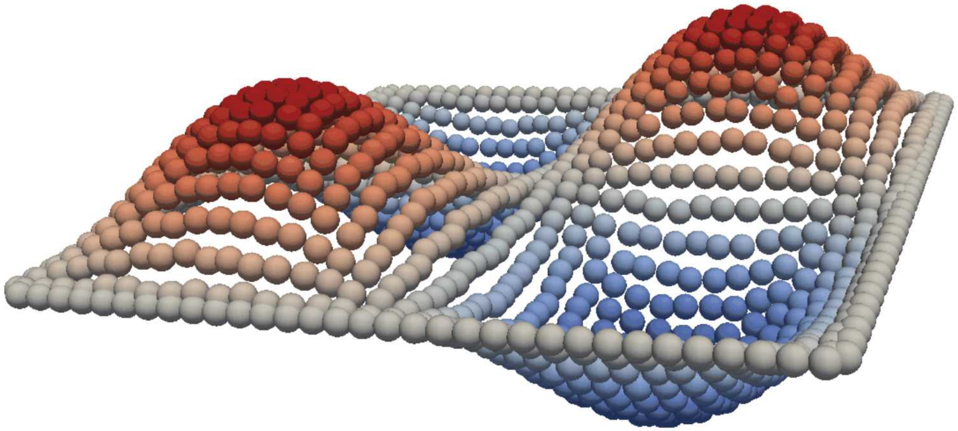
$$\boldsymbol{F} = -\mu \nabla \phi + \boldsymbol{a} \phi$$

- **Linear elasticity:**

$$\boldsymbol{F} = \lambda \left( \nabla \cdot \boldsymbol{u} \right) \boldsymbol{I} + \mu \left( \nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathsf{T}} \right)$$

All problems will be shown for discontinuous material properties to highlight flux continuity of approach.

# First order truncation error for discrete divergence



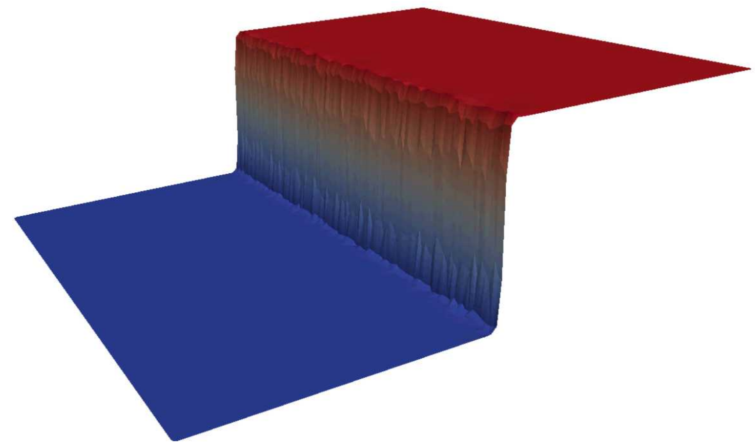
| H | unweighted | weighted |
|---|---|---|
| 1/16 | 0.081 | 0.058 |
| 1/32 | 0.049 | 0.032 |
| 1/64 | 0.024 | 0.015 |
| 1/128 | 0.011 | 0.0072 |

$$\mathbf{F} = -\langle \cos x \sin y, \sin x \cos y \rangle$$

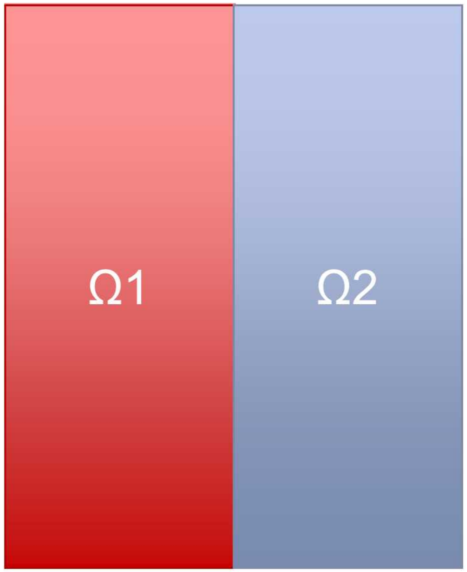
# Darcy: jumps in material properties

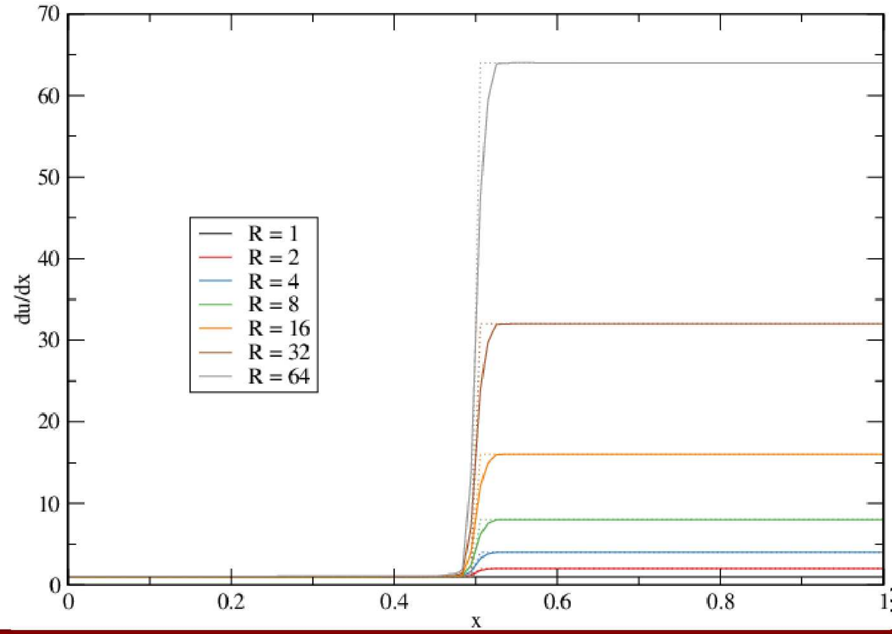
Flux continuity across interface:

$$[\kappa \nabla \phi \cdot \mathbf{n}] = 0$$



$$\nabla \phi \rightarrow$$



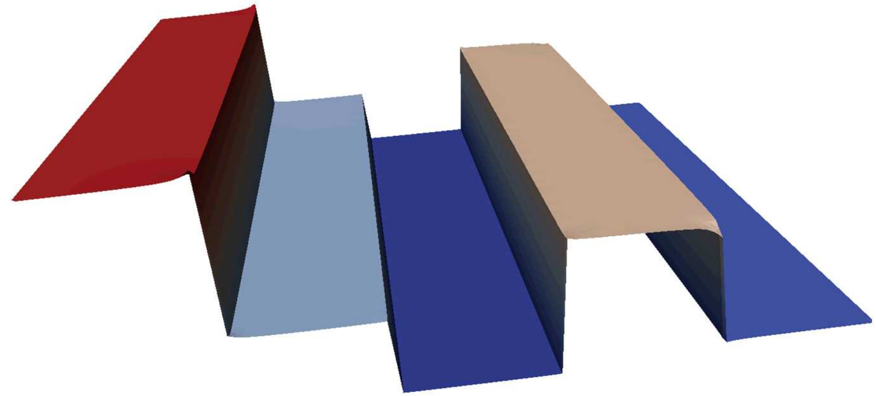



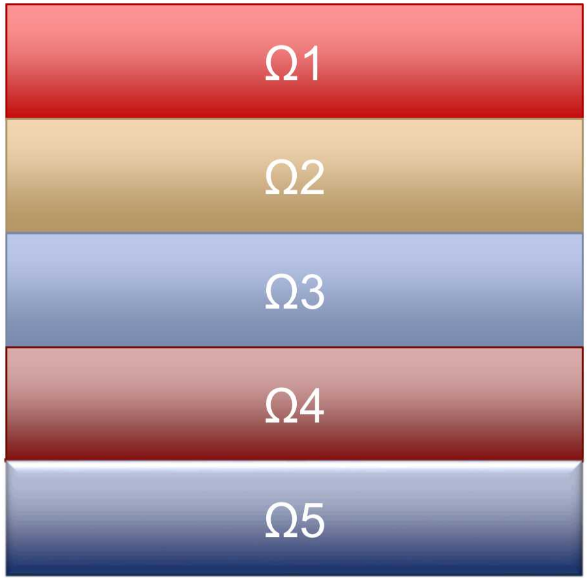
# Darcy: jumps in material properties

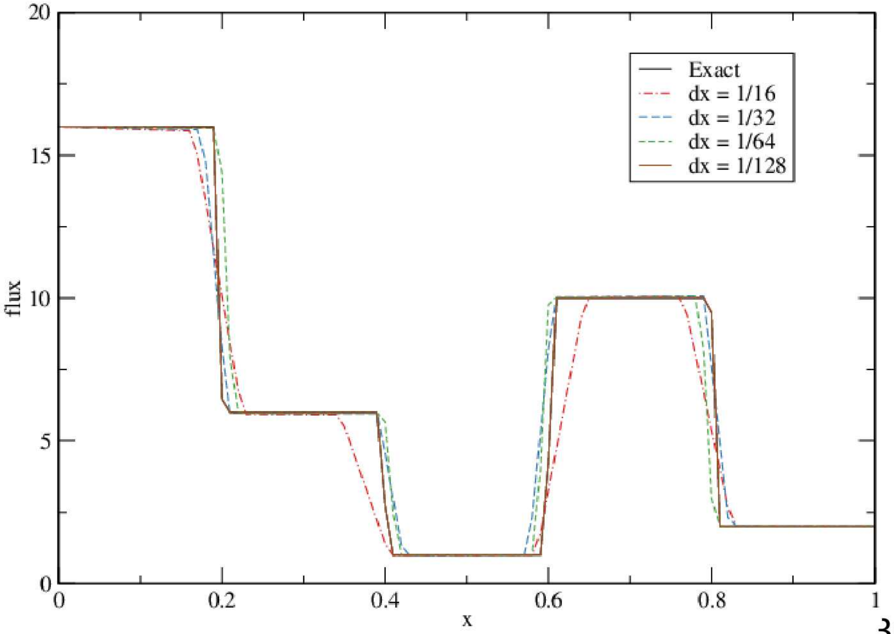
Flux continuity across interface:

$$[\kappa \nabla \phi \cdot \mathbf{n}] = 0$$

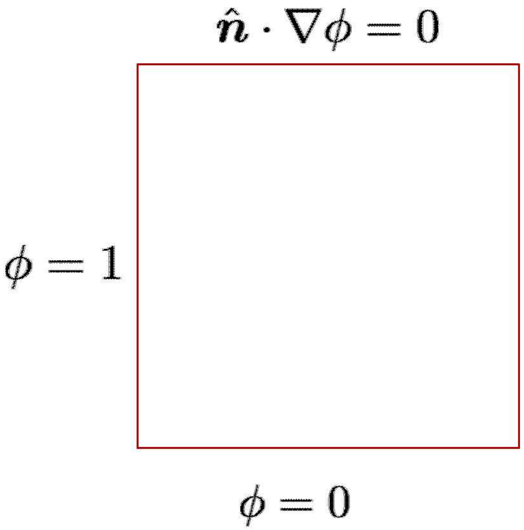
$$\nabla \phi \rightarrow$$

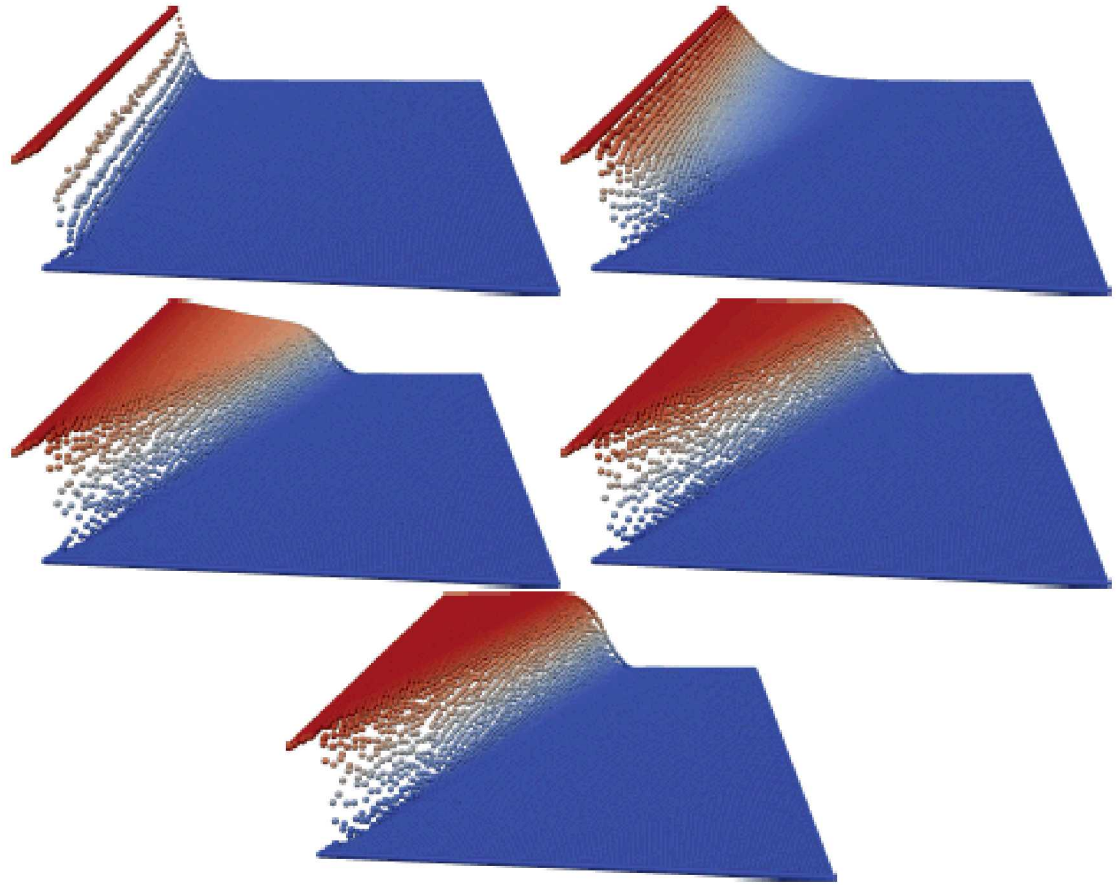




# Singularly perturbed advection diffusion



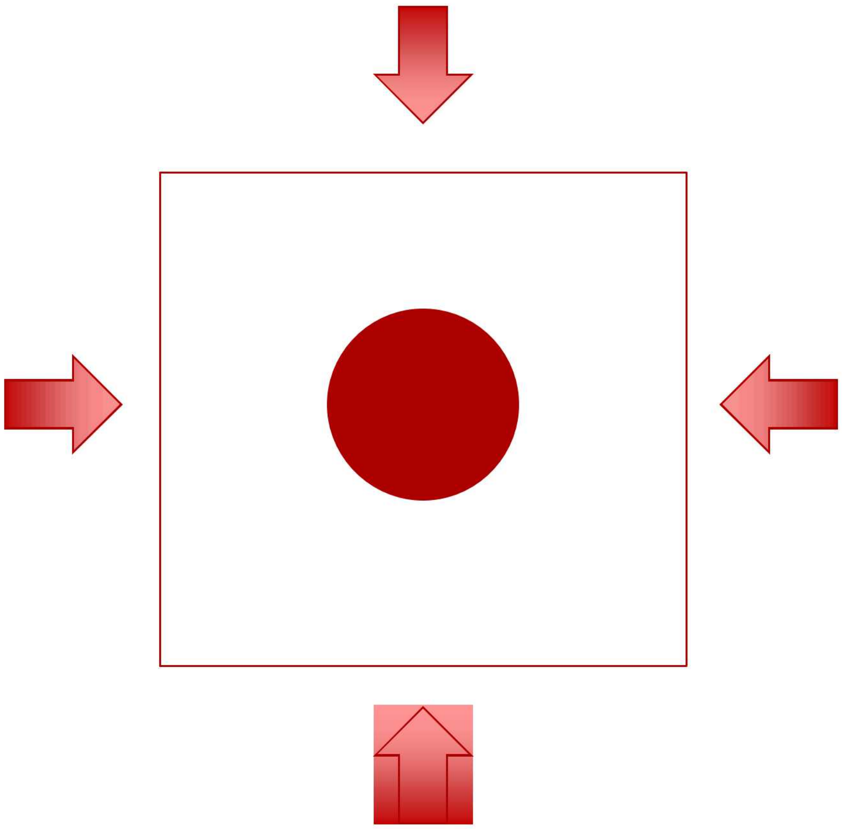



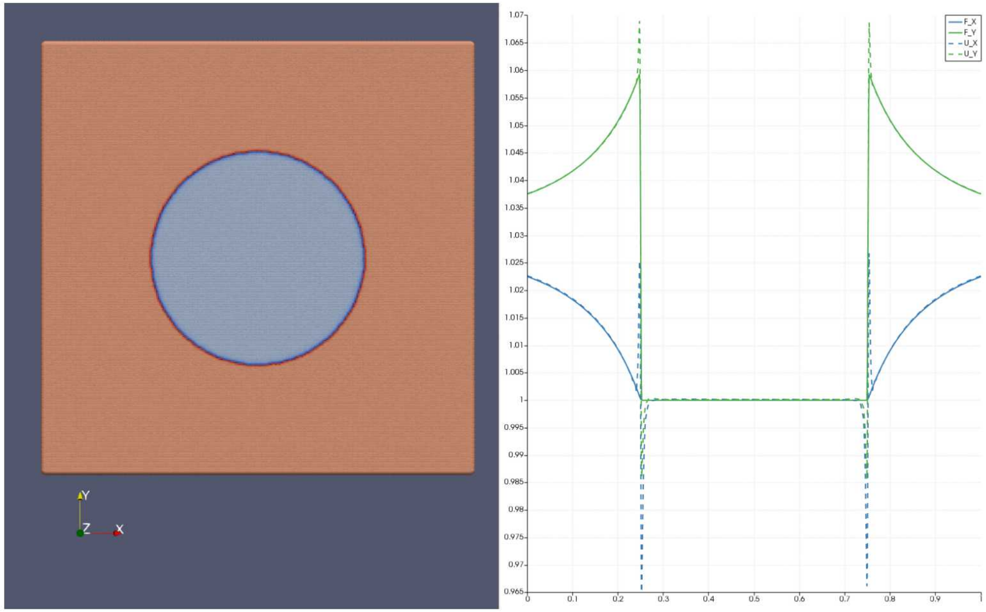
$$\frac{\partial}{\partial t}\phi + \nabla \cdot \mathbf{F} = 0$$

$$\mathbf{F} = \mathbf{a}\phi - \epsilon\nabla\phi$$

Single timestep
$Co \in \{1, 10, 100, 1000, \infty\}$
demonstrating L-stability

# Linearly elastic composite materials






Hydrostatic loading of a stiff inclusion
- Normal stress continuity across interface

# Unification with Combinatorial Hodge theory

**Combinatorial chain complex**

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

**Combinatorial co-chain complex**

$$C^0 \xrightarrow{\delta_0} C^1 \xrightarrow{\delta_1} C^2.$$

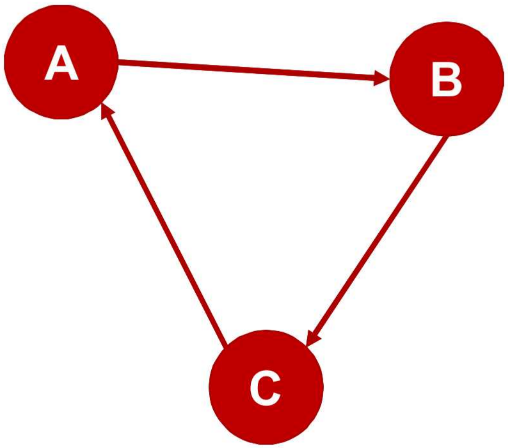
**Example: combinatorial gradient**

$$\delta_0 : C^1 \to C^0$$

$$\delta_0 \phi_{ij} = \phi_j - \phi_i$$

**Note that:**

- $\delta_0$ does *not* converge to $\nabla$
- $\delta_k \circ \delta_{k-1} = 0$




An example 3-clique $(A, B, C)$ belonging to $C_2$

Jiang, Xiaoye, et al. "Statistical ranking and combinatorial Hodge theory." Mathematical Programming 127.1 (2011): 203-244.

# Current work: a meshfree de Rham sequence

Previous scheme can be rewritten in combinatorial Hodge notation:

$$\mu_c div_h(u)_c = \delta_0^* \left[(\delta_0 \psi_f)^\intercal D_f c_f(u)\right]$$

We may similarly define a mimetic curl operator by moving to the right on the combinatorial de Rham complex

$$I_f[\nabla \times u] = \delta_1^* \left[(\delta_1 \psi_e)^\intercal D_e c_e(u)\right]$$

Choosing $I_f[\nabla \times u] = (\delta_0 \psi_f)^\intercal D_f c_f(\nabla \times u)$ we obtain

$$\mu_c div_h \circ curl_h(u)_c = \delta_0^* \delta_1^* \left[(\delta_1 \psi_e)^\intercal D_e c_e(u)\right] = 0$$

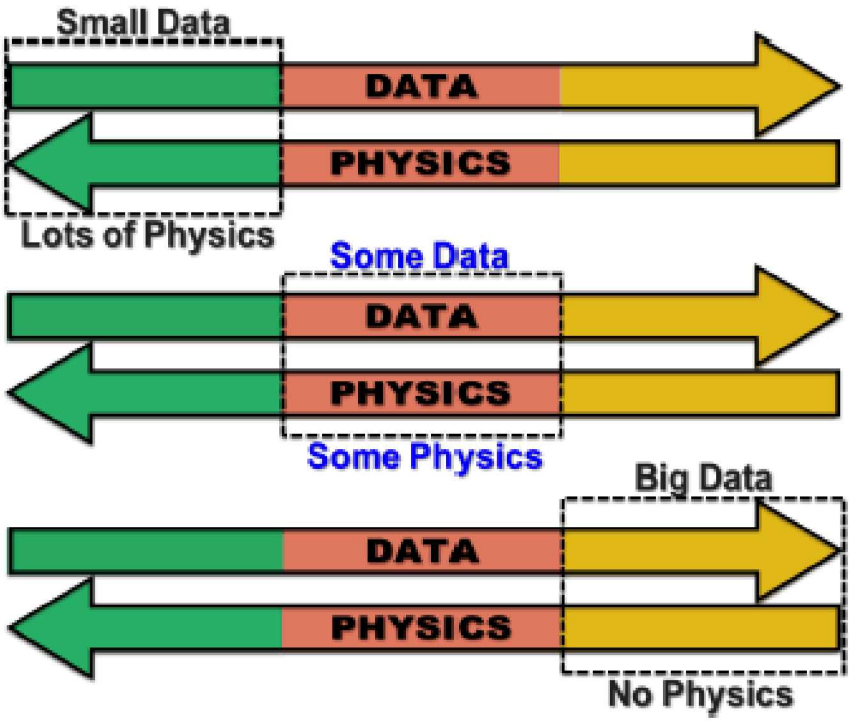
**A direct extension to define virtual curl satisfying div*curl = 0**

# What is scientific machine learning?

**Traditional scientific computing:**
Known model, known theory leading to good discretization with FEM, data primarily for V+V only

**SciML:**
Known model form, unknown constitutive relationships or closures, small amount of high fidelity data

**Traditional machine learning:**
No physics, unknown input/output relationship, learn on huge amounts of data + universal approximation




- ***Sparse data:***
  - Data is either expensive, experimentally intractable, or legally unobtainable
- ***Unstructured data:***
  - Data comes from Lagrangian probes, unstructured FEM meshes – unlike traditional ML on images

***Physics Informed ML:*** **Exploit domain expertise to augment sparse data**

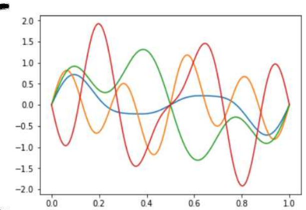
# Operator regression

Given a collection of functions $u_i \in V$, a functional $\tau_x \in V^*$, and a domain $\Omega$ can we infer $\tau_x$ from observations of the form

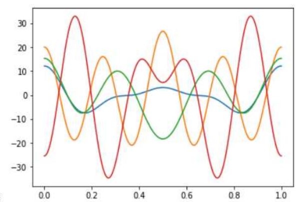
$$\{u_i(x), \tau_x[u_i]\}_{i=1}^{N}?$$

**Example:**

$\{$




,




$\} \Longrightarrow \tau_x = \frac{\partial u}{\partial \hat{x}}\Big|_{\hat{x}=x}$

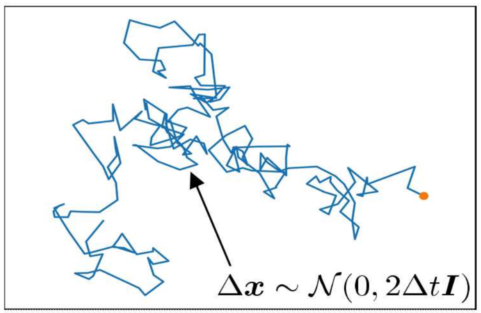
$$\mathcal{L} = \underset{\xi}{argmin} \sum_i ||\tau_x[u_i] - \mathcal{L}_\xi[u_i]||^2_{V^*}$$

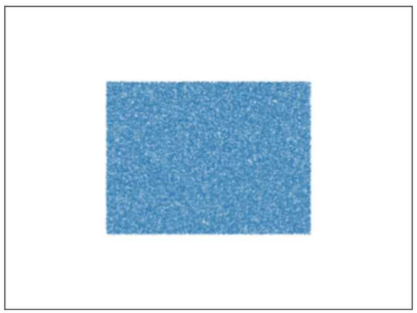
We present learning frameworks corresponding to choice of parameterization:

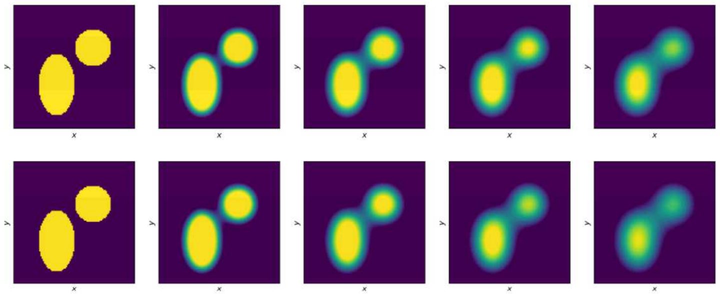
- **GMLS-Nets:** Use meshfree approximation theory to regress operators characterized by scattered samples of data
- **Fourier regression:** Characterize operators via parameterization of Fourier symbol
- **Nonlocal operator regression:** Characterize nonlocal operators via parameterization of nonlocal kernel

# What applications may be characterized in this way?

- **data driven model discovery**
- **numerical homogenization**
- **surrogate model development**





SDE $dX = dW_t$ PDE $\partial_t u = \nabla^2 u$

# Example:

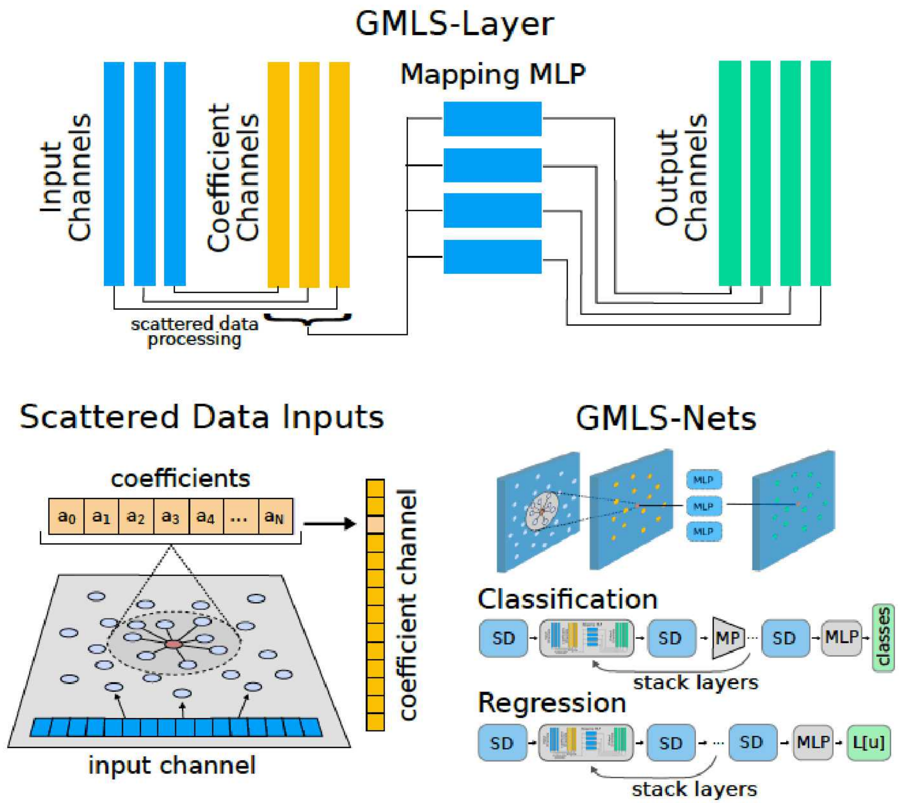
**Assumed physics:** $\partial_t u = \mathcal{L}(u)$

**Observed high-fidelity dynamics:** $\left\{ u(t_n), \frac{u(t_{n+1}) - u(t_n)}{\Delta t} \right\}$

# GMLS-Nets: SciML architecture for unstructured data

w/ Ravi Patel (SNL), Paul Atzberger (UCSB)

- Assume a basis $\Phi$, so that $p \in P \rightarrow p = a^{\intercal}\Phi$
- GMLS thus provides an *optimal local encoding* of data in terms of the coefficient $a$, providing a low-dimensional encoding that may e.g. exploit physics
- Traditionally, GMLS estimates $\tau(u) = a^{\intercal}\tau(\Phi)$, assuming one has knowledge of how the target functional. Instead we seek an operator $q_{\xi} : a \rightarrow \mathbb{R}$, and use gradient descent to tune $\xi$ to match data
- Functionally identical to convolutional networks - we get a stencil that reproduces the operator, but no restriction on e.g. Cartesian data, collar region, etc.

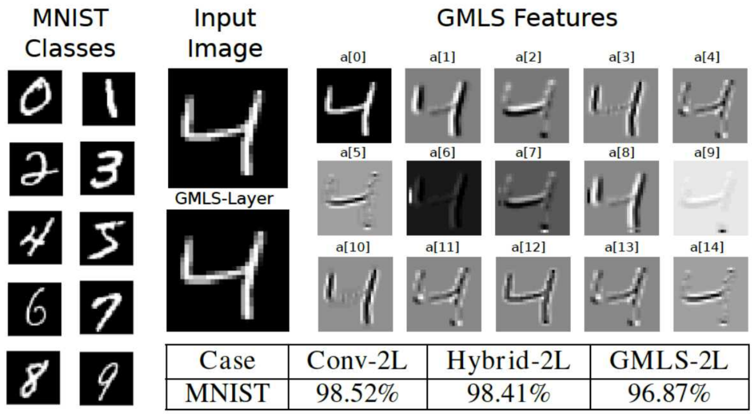



Recently accepted at neuroIPS (https://arxiv.org/pdf/1909.05371.pdf)

**Open-source software:** code and training sets publically available for:

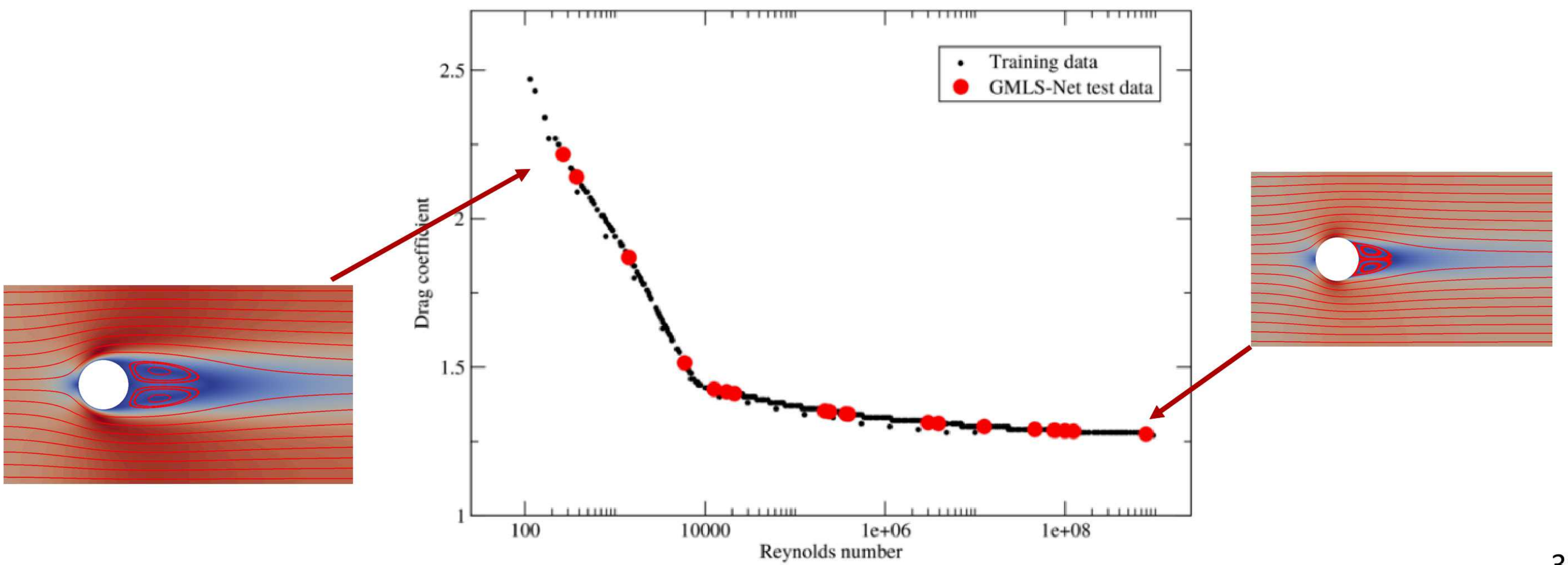
- Tensorflow (https://github.com/rgp62/gmls-nets)
- PyTorch (https://github.com/atzberg/gmls-nets)

# GMLS-Nets: results




| Case | Conv-2L | Hybrid-2L | GMLS-2L |
|---|---|---|---|
| MNIST | 98.52% | 98.41% | 96.87% |

- Provides similar performance to convNets on MNIST due to similar feature extraction capability
- Generalizes convNets to unstructured scientific data:
  - Prediction of drag from cell center velocity field taken from FV data
  - No pressure/viscosity information: drag characterizated entirely by flow



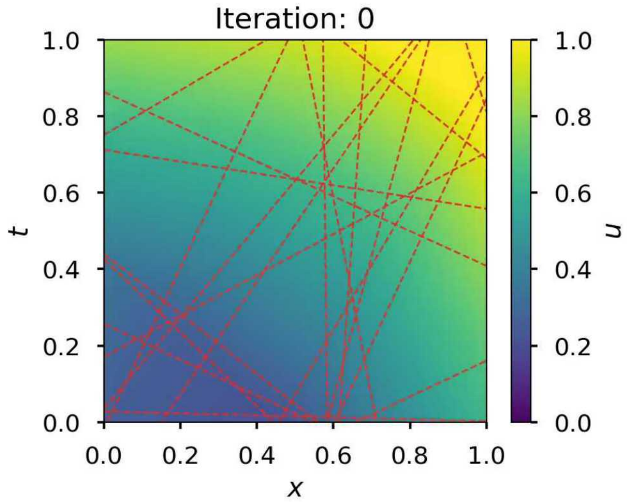
# Physics informed neural networks (PINNs)

Sandia National Laboratories

**Idea:** regularize loss with terms encoding physics knowledge

$$\mathbf{L} = \mathbf{L}_{data} + \epsilon \mathbf{L}_{physics}$$

$$\mathbf{L} = ||u_{data} - \mathcal{NN}||_{\ell_2}^2 + \epsilon ||\mathcal{L}[u_{data}] - \mathcal{L}[\mathcal{NN}]||_{\ell_2}^2$$

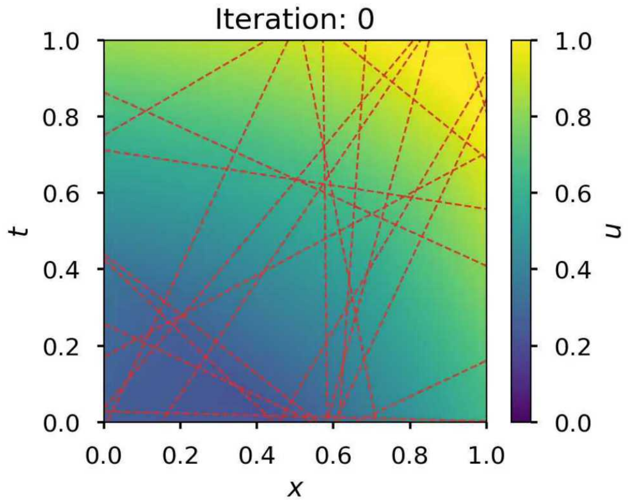



**Example:**

$$\partial_t u + \partial_x u = 0$$

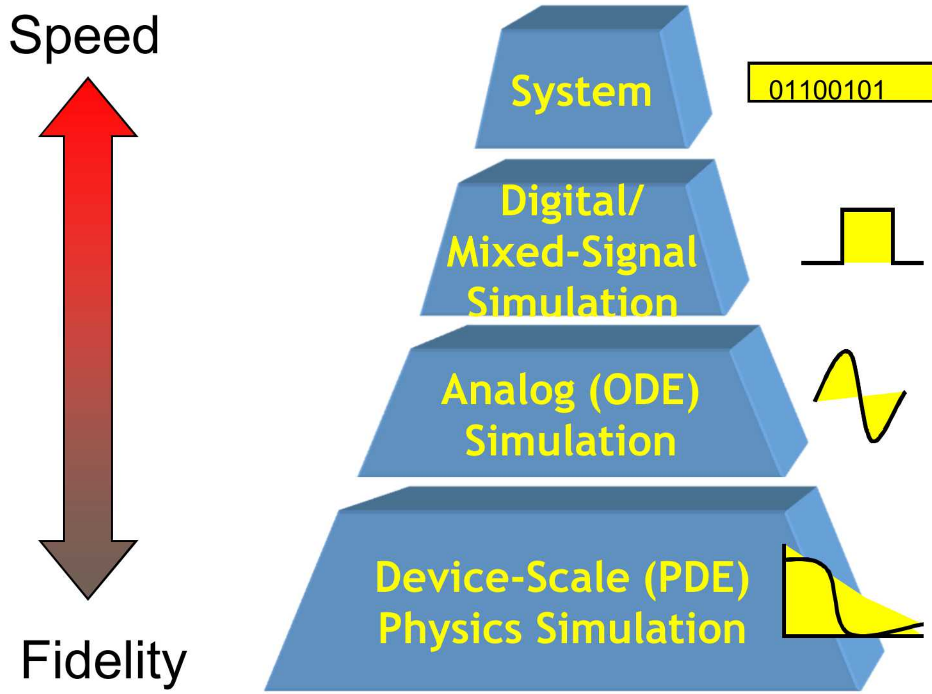
$$u(x,0) = \text{Tent function}$$

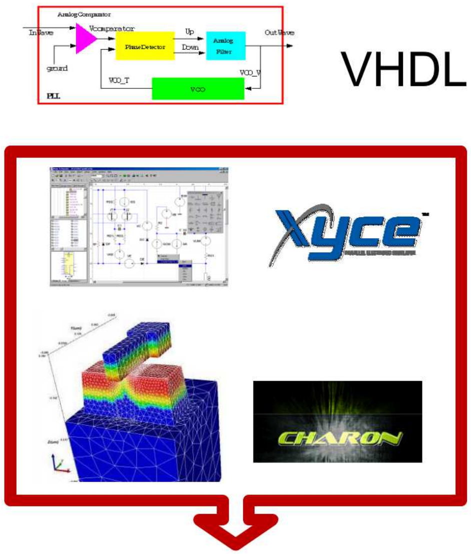
$$u(0,t) = 0$$




**Joint work with Karniadakis, Philms team to improve performance of PINNs**

# PIRAMID: Target problem




Targeted level of speed
+ fidelity

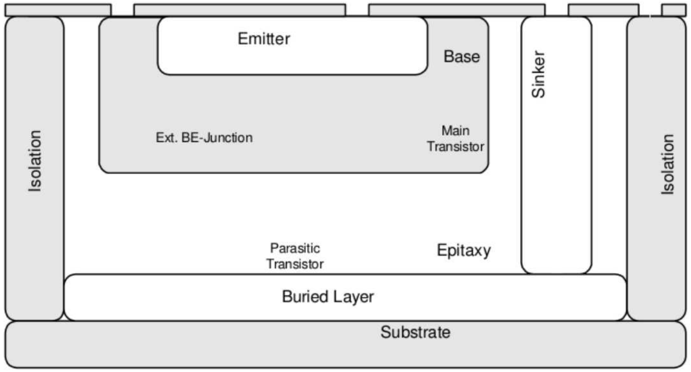
- For system scale simulation, want a process for extracting scalable low-cost network/circuit models from high-fidelity but slow device-scale models
- Often **data-sparse** (high-fidelity sim are expensive/experiments are costly)
- Current approach developing models "by hand" may take **one decade** and multiple domain experts to develop predictive network model!

**LDRD goal: develop a novel Machine Learning approach to automate and reduce physics → analog model incorporation effort from decades/years to months/weeks.**

# Device simulation: TCAD

Technology Aided Computer Design (**TCAD**) simulates semiconductor device **geometry** and underlying 2D/3D **physics** at high resolution and fidelity.

## TCAD BJT device model




## VanRoosbroeck Equations

Poisson equation +
Drift-Diffusion carrier transport

$$\nabla \cdot \epsilon \nabla \phi = -(p - n + N_D^+ - N_A^-)$$

$$\frac{\partial n}{\partial t} = \frac{1}{q} \nabla \cdot (-\mu_n n E - D_n \nabla n) - R_n(n,p)$$

$$\frac{\partial p}{\partial t} = -\frac{1}{q} \nabla \cdot \left(\mu_n p E - D_p \nabla p\right) - R_p(n,p)$$

✓ Considers **device 2D/3D geometry** and **localized physics**, and can incorporate changes due to environmental physics.

✗ Rigorous calibration and validation are time consuming.

✗ Simulation of more than a couple devices is impractical.

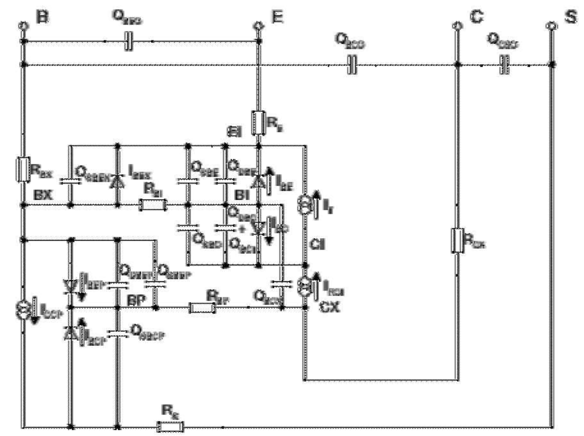
# Device simulation: circuit compact models

**Compact models (CM)** are computationally simpler analog models capturing the functional behavior of the device, e.g., can produce I-V characteristics.

Modified Nodal Analysis
(Kerchoff's Current Law +
Branch Constitutive Equations)

$$\begin{bmatrix} \frac{1}{R_1} & -\frac{1}{R_1} & 0 & 1 & 0 \\ -\frac{1}{R_1} & \frac{1}{R_1}+\frac{1}{R_2}+\frac{1}{R_3} & -\frac{1}{R_2} & 0 & 0 \\ 0 & -\frac{1}{R_2} & \frac{1}{R_2} & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \\ i_{v1} \\ i_{v2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ V1 \\ V2 \end{bmatrix}$$

Circuit compact model




✓ Captures **functional device behavior via electrical pathways (runs much faster than TCAD)** and can be **augmented to include new physics.**

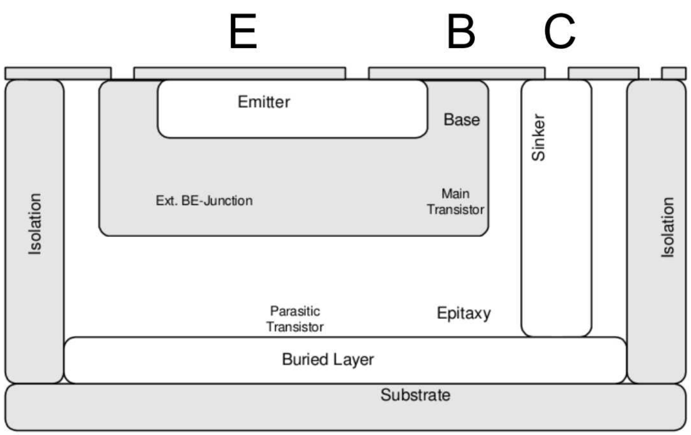
✗ Manual mapping of local physics due to **operating environments** to circuit components is tedious, **severely limiting rapid incorporation of environmental effects into new designs**. New physics = More iterations!

✗ E.g., **QASPR HBT neutron damage model took 7 years to develop**.

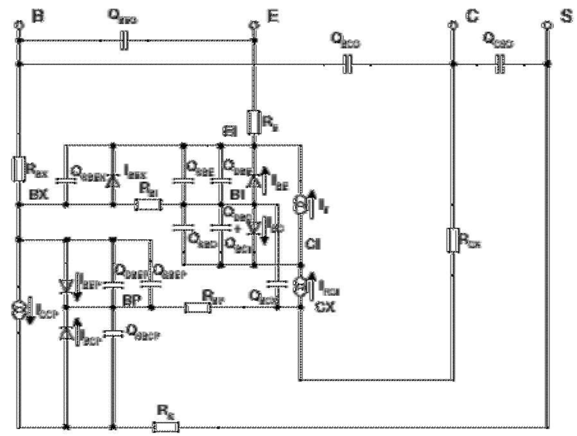
# Objective

"Can we **rapidly and reliably** transfer new physics knowledge to new and existing compact models?"

TCAD BJT device model




Circuit compact model



**PIRAMID aims to introduce a novel ML approach to**
**automate the entire process in one pass!**

- How can we automatically extract a network/circuit topology from PDE model?
- How do we parameterize extracted network?

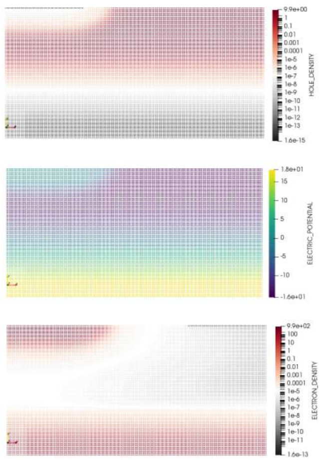
# PIRAMID process flowchart

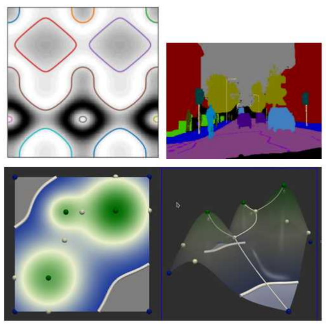
**Physics Priming (PP)**
Perfunctory TCAD

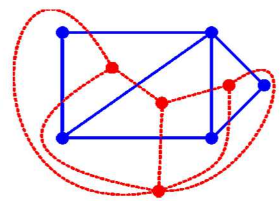
**Region Recognition (RR)**
ML + TDA

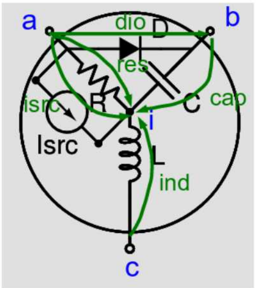
**Topology Tailoring (TT)**

**Interaction Identification (II)**
(seeded w/ established CMs)










Train and Adapt
(using available experimental data)

Simulate high fidelity physics.

Identify significant regions (ML+ Topological Data Analysis)

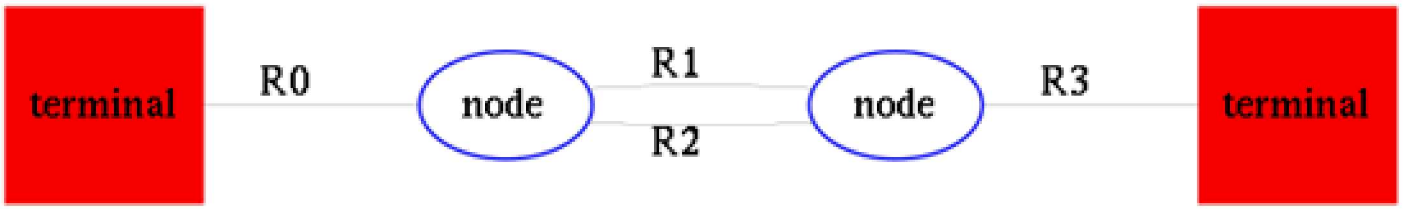
Identify interactions between significant regions.

Prescribe electronics components to physical interactions.

Generate **positive feedback** in the machine learning process (supervised training).

# Physics informed compact modeling

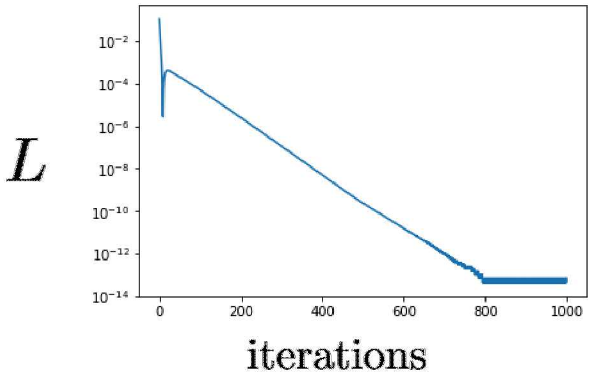
Given a graph extracted from PDE model, use operator regression to "dress" the graph with circuit components




$$\mathbf{L} = \mathbf{L}_{Kirchhoff} + \epsilon \mathbf{L}_{data}$$

$$\mathbf{L} = \sum_{i \in \text{nodes}} \sum_{j \in N_i} |I_{ij}|^2 + \epsilon \sum_{i \in \text{nodes}} |\phi_i^{data} - \phi_i|^2$$

**Proof of concept:**
Given voltages,
Find resistors that balance currents while providing a target current



# Conclusions

- Generalized moving least squares provides an approximation theoretic framework for estimating functionals from general scattered data
- We have used this technique in a wide range of applications
  - Here we presented recent work using GMLS to develop conservative schemes
  - Looking for collaborators to analyze these schemes and apply to subsurface fracture problems
  - For scientific machine learning applications GMLS provides an effective architecture for operator regression
- Incorporation of graphical models with operator regression a promising area to incorporate topological structure into data-driven model development