Reduced-Order Modeling in Nonlinear Computational Mechanics and Applications to Fractures and Thermal Fatigues

J. S. Chen

University of California, San Diego

Qizhi He

San Diego State University

Shigeki Kaneko, Shinobu Yoshimura

The University of Tokyo

Center for Extreme Events Research

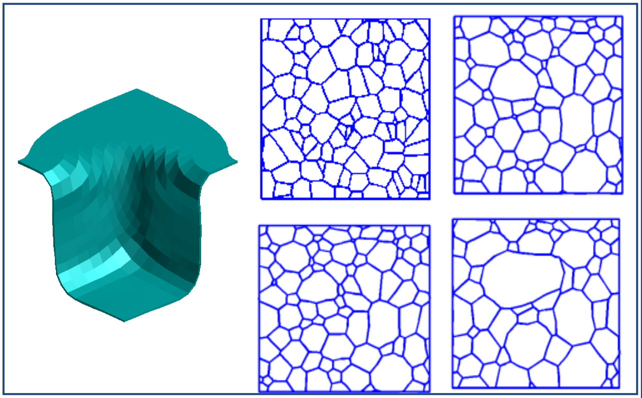
UC San Diego
Jacobs School of Engineering

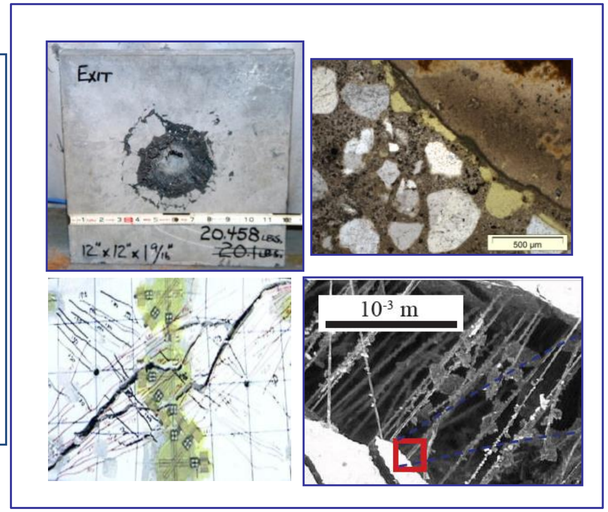
Three Pillars of Science

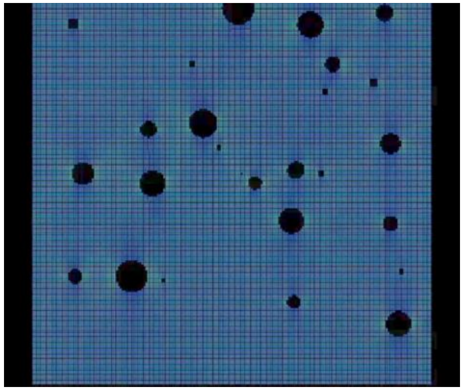
Observation (data)
Hypothesis (models) ⟷ Verification ⟷ Prediction

Observation (data) Hypothesis (models)	Verification	Prediction
Theory	Experimentation Simulation	Theory Experimentation Simulation

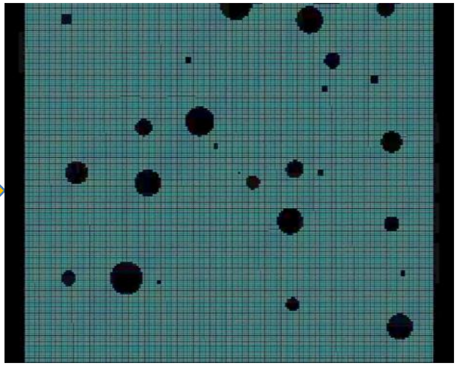
Simulation Models







Physics, Chemistry, Mechanics, Biology, Materials → Mathematical Models → Computational Methods →



Are We Solving the Equations Right? (Verification)

- Accuracy
- Stability
- Efficiency: Model Order Reduction (Lecture 2)

Are We Solving the Right Equations? (Validation)

- Physical models
- Materials models
- Data-Driven (Machine Learning) (Lecture 3)

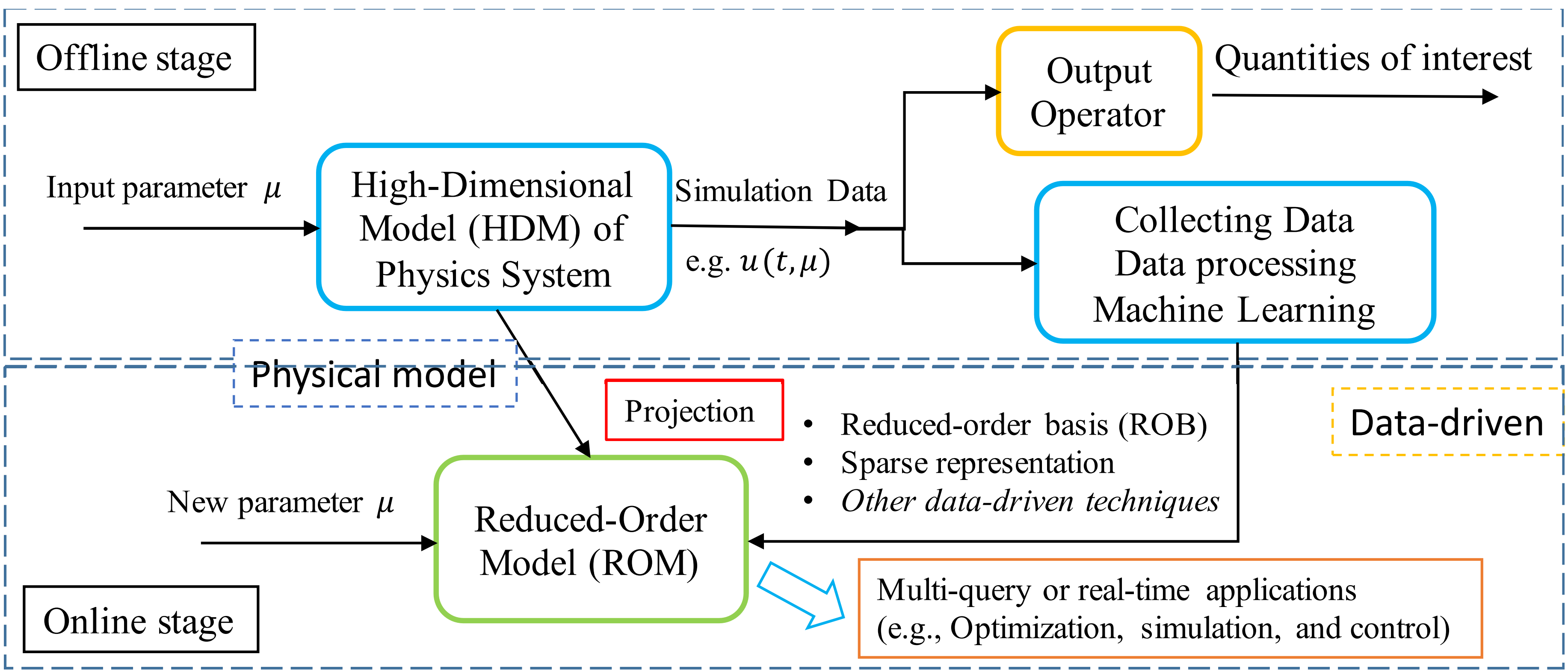
Lab 1: Convolutional Neural Network (CNN) for crack detection

Model Order Reduction (MOR) for Parameterized Systems

Generic parameterized mechanical problem

IBVP: $\mathcal{L}(u,\mathbf{x},t;\boldsymbol{\mu})=0,\ \ \mathbf{x}\in\Omega\subset\mathbb{R}^d,\ \boldsymbol{\mu}\in\mathbb{R}^p$ *(input parameters)*

BC: $\mathcal{B}(u,\mathbf{x},t;\boldsymbol{\mu})=0,\ \ \mathbf{x}\in\partial\Omega$



Proper Orthogonal Decomposition (POD)

High-dimensional model

Karhunen 1946; Loeve 1955; Sirovich 1987; Jolliffe 2002

$$\boldsymbol{f}_{\text{int}}(\boldsymbol{u}, \boldsymbol{\mu}) - \boldsymbol{f}_{\text{ext}}(\boldsymbol{\mu}) = \boldsymbol{0} \quad \text{with} \quad \boldsymbol{f}_{\text{int}} \in \mathbb{R}^{\mathcal{N}}$$

- Offline data collection: $\boldsymbol{X}_s = [\boldsymbol{u}(\boldsymbol{\mu}_1), \ldots, \boldsymbol{u}(\boldsymbol{\mu}_{N_s})] \in \mathbb{R}^{\mathcal{N} \times N_s}$ $\quad \mathcal{X}_s = \text{span}\{\boldsymbol{u}(\boldsymbol{\mu}_1), \ldots, \boldsymbol{u}(\boldsymbol{\mu}_{N_s})\}$

  where $r = \text{rank}(\boldsymbol{X}_s)$

- POD Basis $\{\boldsymbol{v}_i\}_{i=1}^k, k << N$

  Minimization of reconstruction error: $$\min_{\{\boldsymbol{v}_i\}_{i=1}^k} \sum_{j=1}^{N_s} \left\| \boldsymbol{u}(\boldsymbol{\mu}_j) - \underbrace{\sum_{i=1}^{k} (\boldsymbol{u}(\boldsymbol{\mu}_j)^T \boldsymbol{v}_i) \boldsymbol{v}_i}_{\boldsymbol{u}^r(\boldsymbol{\mu}_j)} \right\|_2^2,$$

  $$\text{subject to} \quad \boldsymbol{v}_i^T \boldsymbol{v}_j = \delta_{ij}, \quad i, j = 1, \ldots, k, \; k << N$$

- Related to Principal component analysis (PCA) & SVD:

  $\boldsymbol{V}\boldsymbol{V}^T\boldsymbol{X}_s \equiv \tilde{\boldsymbol{X}}_s$

  $= \textit{reconstruction of } \boldsymbol{X}_s$

  Maximum of covariance: $$\arg \min_{\{\boldsymbol{v}_i\}_{i=1}^k, \boldsymbol{V}^T\boldsymbol{V} = \mathbf{I}} \sum_{j=1}^{N_s} \left\| \boldsymbol{u}(\boldsymbol{\mu}_j) - \sum_{i=1}^{k} (\boldsymbol{u}(\boldsymbol{\mu}_j)^T \boldsymbol{v}_i) \boldsymbol{v}_i \right\|_2^2 = \arg \min_{\boldsymbol{V}^T\boldsymbol{V} = \mathbf{I}} \left\| \boldsymbol{X}_s - \boldsymbol{V}\boldsymbol{V}^T\boldsymbol{X}_s \right\|_F^2$$

  $$= \arg \max_{\boldsymbol{V}^T\boldsymbol{V} = \mathbf{I}} \text{trace}(\boldsymbol{V}^T \boldsymbol{X}_s \boldsymbol{X}_s^T \boldsymbol{V}),$$

✓ *a* priori Error Estimate: $\boldsymbol{X}_s = \boldsymbol{V}_r \boldsymbol{\Sigma}_r \boldsymbol{W}_r^T,$

$$\boldsymbol{\Sigma}_r = \text{diag}(\sigma_1, \ldots, \sigma_r) \in \mathbb{R}^{r \times r}, \quad \text{with } \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0,$$

$$\boxed{\sum_{j=1}^{N_s} \left\| \boldsymbol{u}(\boldsymbol{\mu}_j) - \sum_{i=1}^{k} (\boldsymbol{u}(\boldsymbol{\mu}_j)^T \boldsymbol{v}_i) \boldsymbol{v}_i \right\|_2^2 = \left\| \boldsymbol{X}_s - \boldsymbol{V}\boldsymbol{V}^T\boldsymbol{X}_s \right\|_F^2 = \sum_{i=k+1}^{r} \sigma_i^2}$$

Fracture in Engineering Problems

Simulating damage initiation and subsequent global structural failure is one of the most active topics

Concrete Cracks

Hepatic surgery
(H. Courtecuisse, J. Allard, P. Kerfriden et al. 2014)

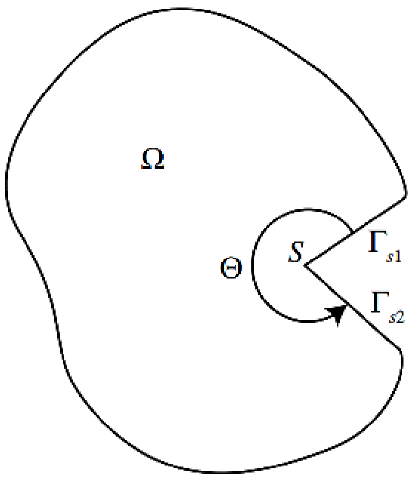
However, a primary challenge of applying *MOR for fracture* is how to represent the local nature of *discontinuities* and *singularities* in the low-dimensional subspaces

Existing methods

- Hybrid methods full-order/Reduced-order based on domain decomposition *(Galland et al. 2011; Kerfriden et al. 2012, 2013)*
- Local-global method *(Niroomandi et al. 2012)*

goal: physics-preserving ROM

Proposed MOR for Fracture Mechanics



Solution decomposition

$$u^h(\mathbf{x}) = \bar{u}^h(\mathbf{x}) + \hat{u}^h(\mathbf{x}) = \sum_{I}^{N} \psi_I(\mathbf{x}) d_I + \sum_{M=1}^{Nenr} F_M(\mathbf{x}) a_M$$

$\bar{u}^h$: smooth solution

$\hat{u}^h$: non-smooth solution

Integrated Singular Boundary Function Method (ISBFM)

[Olson et al., 1991; Georgiou et al., 1996]

$$\nabla \cdot (\mathbf{C} : \nabla_s \mathbf{F}_M) = \mathbf{0} \text{ in } \Omega$$
$$\mathbf{n} \cdot (\mathbf{C} : \nabla_s \mathbf{F}_M) = \mathbf{0} \text{ on } \Gamma_C$$

→ ISBFM Galerkin formulation

- ➢ Low-order quadrature
- ➢ *Sparse system*

Physics-preserving decomposed reduction projection

$$\mathbf{d}^r = \mathbf{P}\boldsymbol{\alpha}, \quad \mathbf{d}^r \in \mathbb{R}^N, \boldsymbol{\alpha} \in \mathbb{R}^k, k \ll N$$

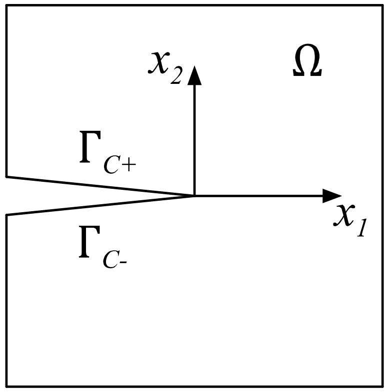
Chen, J. S., Marodon, C. and Hu, H. Y., "Model Order Reduction for Meshfree Solution of Poisson Singularity Problems," IJNME, Vol. 102, pp. 1211–1237, 2015.

Fine-Scale Fracture Model

- Linear elastic fracture mechanics (LEFM)

$$\mathrm{div}\boldsymbol{\sigma} = 0, \quad \text{in } \Omega$$
$$\mathbf{u} = \mathbf{g}, \quad \text{on } \Gamma_g$$
$$\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{h}, \quad \text{on } \Gamma_h$$

$$\boldsymbol{\sigma} = \mathbf{C} : \nabla_s \mathbf{u}^h = \mathbf{C} : \nabla_s (\bar{\mathbf{u}}^h + \hat{\mathbf{u}}^h)$$

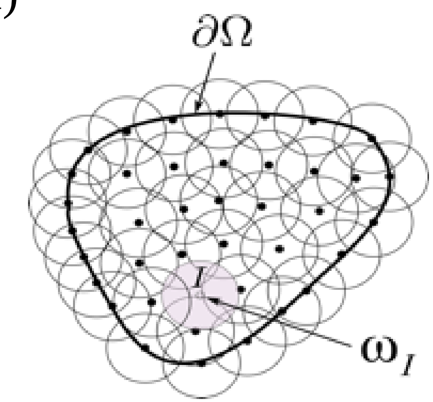


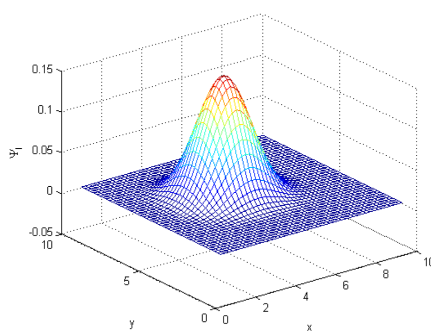
$$\Gamma_{C+} \cup \Gamma_{C-} = \Gamma_C \subset \Gamma_h$$

- Reproducing Kernel Particle Method (RKPM)

$$\bar{\mathbf{u}}^h(\mathbf{x}) = \sum_{I}^{N} \psi_I(\mathbf{x}) \bar{\mathbf{d}}_I = \boldsymbol{\Psi} \bar{\mathbf{d}}$$

$$\psi_I = \underbrace{\mathbf{H}^T(\mathbf{0}) \mathbf{M}^{-1}(\mathbf{x}) \mathbf{H}(\mathbf{x} - \mathbf{x}_I)}_{\text{correction}} \underbrace{\phi_\alpha(\mathbf{x} - \mathbf{x}_I)}_{\text{kernel}}$$





✓ *Naturally capture jump across crack*

Liu, Jun, Zhang, Int. J. Numer. Meth. Engng. 1995
Chen, Pan, Wu, Liu, Comput. Methods Appl. Mech. Engrg. 1996
Belytschko, Krongauz, Organ et al, Comput. Methods Appl. Mech. Engrg. 1996

Near-tip Enrichment Basis Function

The enrichment basis functions derived from William's solution (*William 1952*) are composed of symmetric and anti-symmetric components to the crack surface

$$\boxed{\begin{aligned} &\nabla\cdot(\mathbf{C}:\nabla_s\mathbf{F}_M)=\mathbf{0} \quad \text{in } \Omega \\ &\mathbf{n}\cdot(\mathbf{C}:\nabla_s\mathbf{F}_M)=\mathbf{0} \quad \text{on } \Gamma_C \end{aligned}}$$

$$\hat{u}_i^h=\sum_{M=1}^{\hat{N}}F_{Mi}^{\mathrm{s}}\hat{d}_M^{\mathrm{s}}+\sum_{M=1}^{\hat{N}}F_{Mi}^{\mathrm{as}}\hat{d}_M^{\mathrm{as}} \quad \text{or} \quad \hat{\mathbf{u}}^h=\sum_{M=1}^{2\hat{N}}\mathbf{F}_M\hat{\mathbf{d}}_M$$

$$\mathrm{F}_M^{\mathrm{s}}=\begin{bmatrix} F_{M1}^{\mathrm{s}}(r,q) \\ F_{M2}^{\mathrm{s}}(r,q) \end{bmatrix}=\begin{bmatrix} r^{M/2}\left[\left(k+\dfrac{M}{2}+(-1)^M\right)\cos\left(\dfrac{M}{2}\right)-\dfrac{M}{2}\cos\left(\left(\dfrac{M}{2}-2\right)q\right)\right] \\ r^{M/2}\left[\left(k-\dfrac{M}{2}-(-1)^M\right)\sin\left(\dfrac{M}{2}\right)+\dfrac{M}{2}\sin\left(\left(\dfrac{M}{2}-2\right)q\right)\right] \end{bmatrix}$$

$$\mathrm{F}_M^{\mathrm{as}}=\begin{bmatrix} F_{M1}^{\mathrm{as}}(r,q) \\ F_{M2}^{\mathrm{as}}(r,q) \end{bmatrix}=\begin{bmatrix} -r^{M/2}\left[\left(k+\dfrac{M}{2}-(-1)^M\right)\sin\left(\dfrac{Mq}{2}\right)-\dfrac{M}{2}\sin\left(\left(\dfrac{M}{2}-2\right)q\right)\right] \\ r^{M/2}\left[\left(k-\dfrac{M}{2}+(-1)^M\right)\cos\left(\dfrac{Mq}{2}\right)+\dfrac{M}{2}\cos\left(\left(\dfrac{M}{2}-2\right)q\right)\right] \end{bmatrix}$$

He, Q., Chen, J. S, Marodon, C., "A Decomposed Subspace Reduction for Fracture Mechanics based on the Meshfree Integrated Singular Basis Function Method", Computational Mechanics, Vol. 63, pp. 593–614, 2019.

Standard Galerkin Weak From

Galerkin weak form

Find $\mathbf{u}^h = \bar{\mathbf{u}}^h + \hat{\mathbf{u}}^h \in \mathbb{V}^h \subset H^1(\Omega)$, such that

$$a(\delta\mathbf{u}^h, \mathbf{u}^h) = l(\delta\mathbf{u}^h) \quad \forall \delta\mathbf{u}^h \in \mathbb{V}^h$$

where $a(\delta\mathbf{u}^h, \mathbf{u}^h) = \boxed{\int_\Omega \delta u^h_{(i,j)} C_{ijkl} u^h_{(k,l)} \mathrm{d}\Omega} - \int_{\Gamma_{g_i}} \delta\sigma^h_{ij} n_j u^h_i \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \delta u^h_i \sigma^h_{ij} n_j \mathrm{d}\Gamma + \beta \int_{\Gamma_{g_i}} \delta u^h_i u^h_i \mathrm{d}\Gamma$

$$l(\delta\mathbf{u}^h) = \int_{\Gamma_{h_i}} \delta u^h_i h_i \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \delta\sigma^h_{ij} n_j g_i \mathrm{d}\Gamma + \beta \int_{\Gamma_{g_i}} \delta u^h_i g_i \mathrm{d}\Gamma$$

Needs very higher order quadrature due to $\hat{\mathbf{u}}^h \propto O\left(r^{1/2}\right)$

Standard Galerkin formulations

$$\delta\bar{u}_i \begin{cases} \int_\Omega \delta\bar{u}^h_{(i,j)} C_{ijkl} \left(\bar{u}^h_{(k,l)} + \hat{u}^h_{(k,l)}\right) \mathrm{d}\Omega - \int_{\Gamma_{g_i}} \delta\bar{\sigma}^h_{ij} n_j u^h_i \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \delta\bar{u}^h_i C_{ijkl} u^h_{(k,l)} n_j \mathrm{d}\Gamma + \beta \int_{\Gamma_{g_i}} \delta\bar{u}^h_i u^h_i \mathrm{d}\Gamma \\ = \int_{\Gamma_{h_i}} \delta\bar{u}^h_i h_i \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \delta\bar{\sigma}^h_{ij} n_j g_i \mathrm{d}\Gamma + \beta \int_{\Gamma_{g_i}} \delta\bar{u}^h_i g_i \mathrm{d}\Gamma \end{cases}$$

$$\delta\hat{u}_i \begin{cases} \int_\Omega \delta\hat{u}^h_{(i,j)} C_{ijkl} \left(\bar{u}^h_{(k,l)} + \hat{u}^h_{(k,l)}\right) \mathrm{d}\Omega - \int_{\Gamma_{g_i}} \delta\hat{\sigma}^h_{ij} n_j u^h_i \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \delta\hat{u}^h_i C_{ijkl} u^h_{(k,l)} n_j \mathrm{d}\Gamma + \beta \int_{\Gamma_{g_i}} \delta\hat{u}^h_i u^h_i \mathrm{d}\Gamma \\ = \int_{\Gamma_{h_i}} \delta\hat{u}^h_i h_i \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \delta\hat{\sigma}^h_{ij} n_j g_i \mathrm{d}\Gamma + \beta \int_{\Gamma_{g_i}} \delta\hat{u}^h_i g_i \mathrm{d}\Gamma \end{cases}$$

ISBFM Galerkin Weak From

Integrated Singular Boundary Function Method (ISBFM)

$$\boxed{\begin{aligned} &\hat{\sigma}^h_{ij,j} = C_{ijkl}\hat{u}^h_{(k,l),j} = 0 \quad \text{in } \Omega \\ &C_{ijkl}\hat{u}^h_{(k,l)}n_j = 0 \qquad \text{on } \Gamma_C \end{aligned}} \Rightarrow$$

$$\int_\Omega \delta u_{(i,j)} C_{ijkl}\hat{u}^h_{(k,l)} \mathrm{d}\Omega = \int_{\Gamma_{g_i}+\Gamma_{h_i}} \delta u_i C_{ijkl}\hat{u}^h_{(k,l)} n_j \mathrm{d}\Gamma$$

$$= \int_{\Gamma_{g_i}+\bar{\Gamma}_{h_i}} \delta u_i C_{ijkl}\hat{u}^h_{(k,l)} n_j \mathrm{d}\Gamma, \quad \forall\ \delta u \in \mathbb{V}^h$$

where $\bar{\Gamma}_{h_i} = \Gamma_{h_i} \setminus \Gamma_C$

ISBFM-Galerkin formulations

$$\int_\Omega \delta\bar{u}^h_{(i,j)} C_{ijkl}\bar{u}^h_{(k,l)} \mathrm{d}\Omega - \int_{\Gamma_{g_i}} \left(\delta\bar{u}^h_i \left(C_{ijkl}\bar{u}^h_{(k,l)}n_j - \beta\bar{u}^h_i\right) + \delta\bar{\sigma}^h_{ij} n_j \bar{u}^h_i\right) \mathrm{d}\Gamma$$

$$+\int_{\bar{\Gamma}_{h_i}} \delta\bar{u}^h_i C_{ijkl}\hat{u}^h_{(k,l)} n_j \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \left(\delta\bar{\sigma}^h_{ij} n_j \hat{u}^h_i - \delta\bar{u}^h_i \beta\hat{u}^h_i\right) \mathrm{d}\Gamma$$

$$= \int_{\bar{\Gamma}_{h_i}} \delta\bar{u}^h_i h_i \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \left(\delta\bar{\sigma}^h_{ij} n_j g_i - \delta\bar{u}^h_i \beta g_i\right) \mathrm{d}\Gamma$$

$$\int_{\bar{\Gamma}_{h_i}} \delta\hat{u}^h_{(i,j)} C_{ijkl}\bar{u}^h_k n_l \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \left(\delta\hat{\sigma}^h_{ij} n_j \bar{u}^h_i - \delta\hat{u}^h_i \beta\bar{u}^h_i\right) \mathrm{d}\Gamma$$

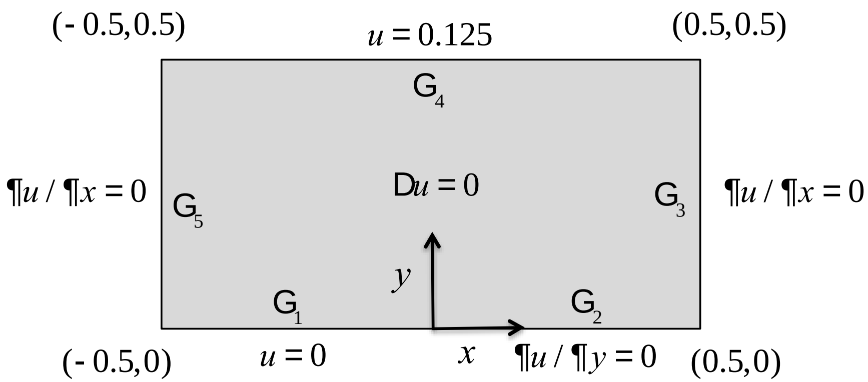
$$+\int_{\bar{\Gamma}_{h_i}} \delta\hat{u}^h_i C_{ijkl}\hat{u}^h_{(k,l)} n_j \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \left(\delta\hat{\sigma}^h_{ij} n_j \hat{u}^h_i - \delta\hat{u}^h_i \beta\hat{u}^h_i\right) \mathrm{d}\Gamma$$

$$= \int_{\bar{\Gamma}_{h_i}} \delta\hat{u}^h_i h_i \mathrm{d}\Gamma - \int_{\Gamma_{g_i}} \left(\delta\hat{\sigma}^h_{ij} n_j g_i - \delta\hat{u}^h_i \beta g_i\right) \mathrm{d}\Gamma$$

- Non-smooth enrichment functions simply appear on boundaries away from the crack tip
- Avoid taking high-order domain quadrature while capturing the singularity
- Sparse coupling sub-matrix

Convergence Test for Problem with Singularity

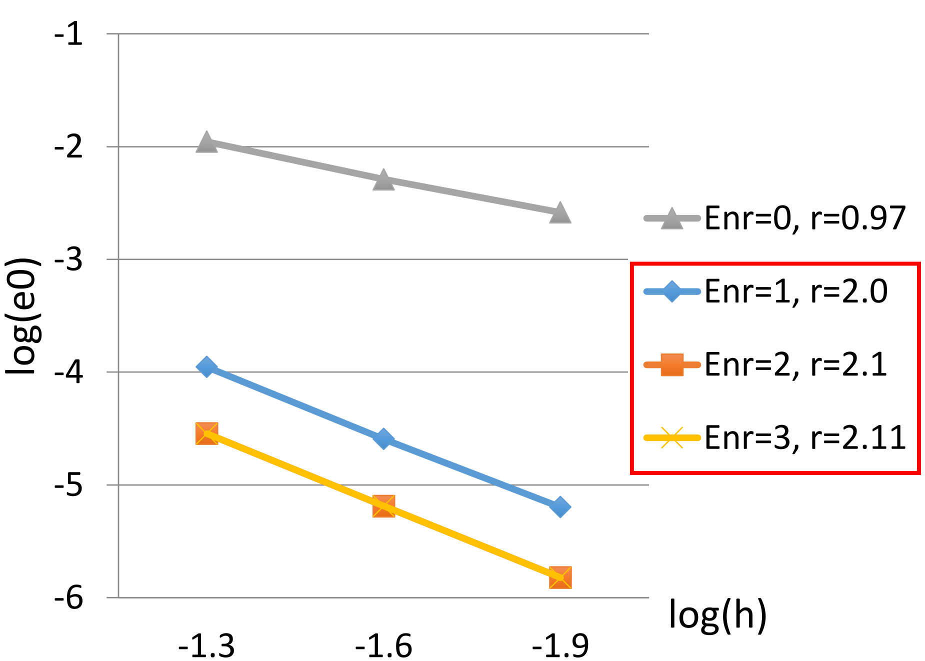
Crack-beam (Motz's) Problem ($\alpha = 0.5$)

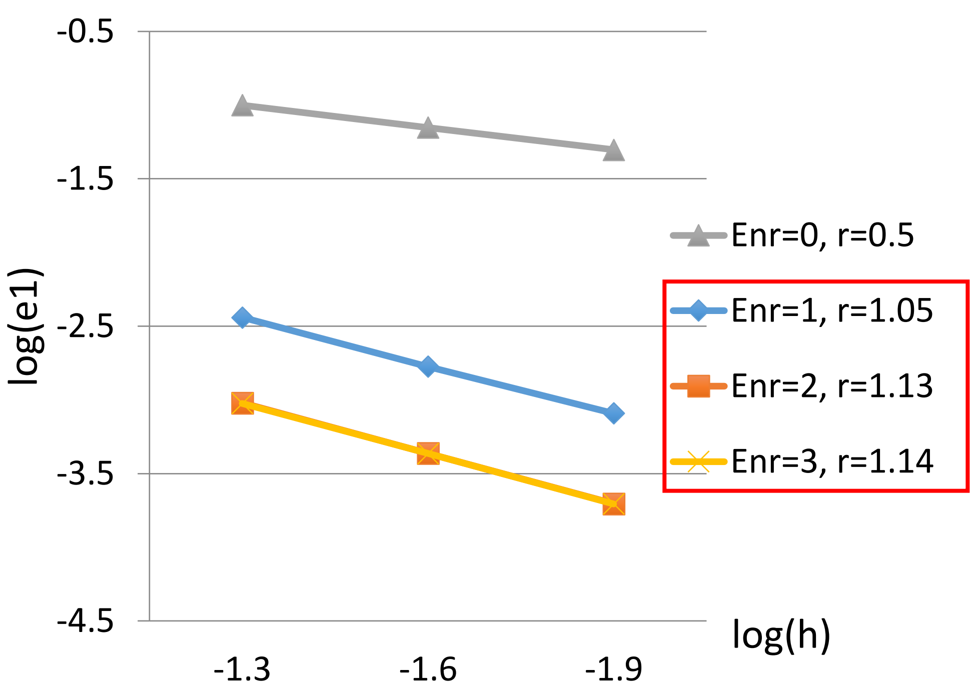


- <u>Singularities are well captured</u>
- <u>Optimal convergence is restored</u>

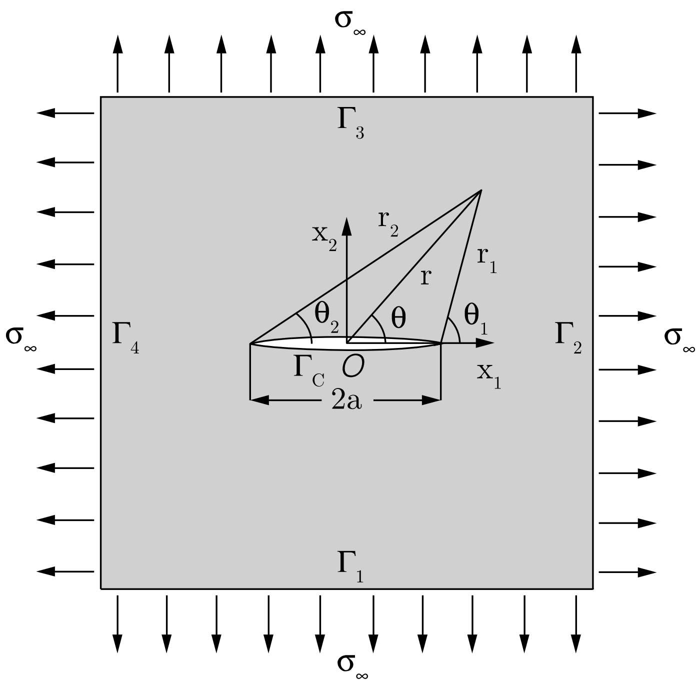
$$\left\|u-u^h\right\|_0 \le \tilde{c}h^\eta \left\|u\right\|_{1+\alpha}, \eta = 2\alpha = 1$$

$$\left\|u-u^h\right\|_1 \le ch^\mu \left\|u\right\|_r, \ \mu = \min(k,\alpha) = 0.5$$





Loaded line crack model



$$\Gamma_2 \subset \Gamma_g \text{ and } \{\Gamma_1 \cup \Gamma_3 \cup \Gamma_4\} \subset \Gamma_h$$

Model setting

$$E = 64{,}000 \ \mathrm{N/mm^2}$$

$$\nu = 0.2$$

$$\sigma_\infty = 1$$

Meshfree approximation setting

$$k = 1$$

$$a = 1.71\,h$$

$$\beta = 100E/h$$

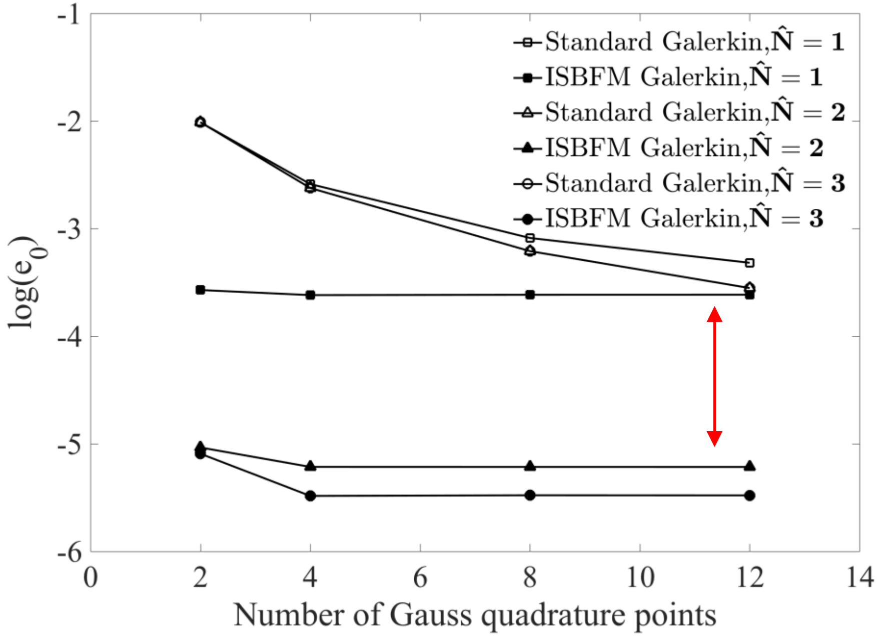
$$\bar{N} = 36 \times 36$$

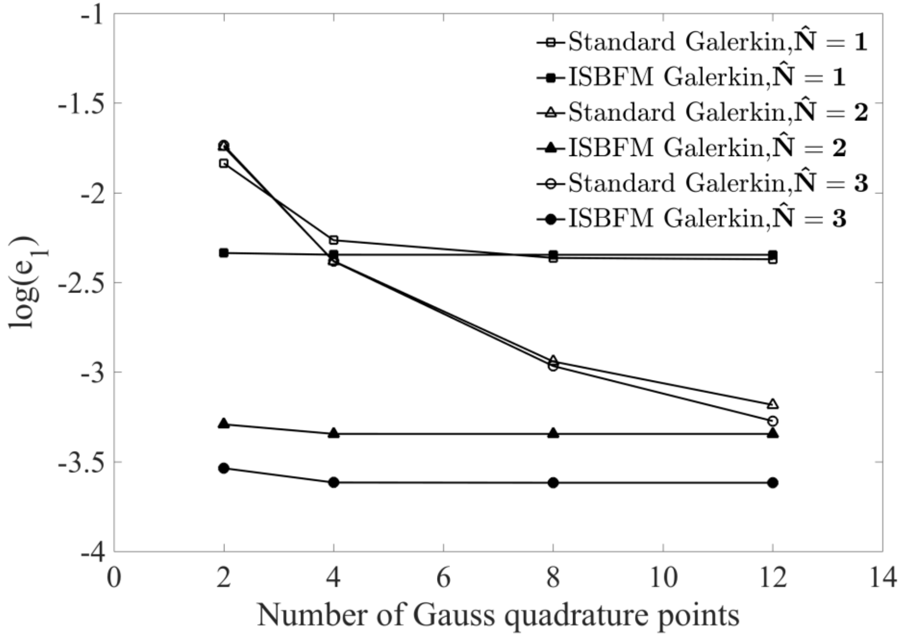
Error measurement:

$$e_0 = \frac{\left\|\mathbf{u} - \mathbf{u}^h\right\|_{L_2}}{\left\|\mathbf{u}\right\|_{L_2}},\ e_1 = \frac{\left\|\nabla^s\mathbf{u} - \nabla^s\mathbf{u}^h\right\|_{L_2}}{\left\|\nabla^s\mathbf{u}\right\|_{L_2}},\ e_{KI} = \frac{\left|K_I - K_I^h\right|}{\left|K_I\right|}$$

Comparison of Standard and ISBFM Galerkin Methods

$\hat{N}$: number of non-smooth enrichment basis





ISBFM-Galerkin method improves more than 2 orders of accuracy while using much less quadrature points in fine-scale modeling.

Computational burden reduction

Decomposed Subspace Reduction Method

Discretized by the enriched meshfree approxiamtion, the ISBFM Galerkin formulation results in a <u>*full-scale discrete system*</u> of dimension $N = 2(\bar{N} + \hat{N})$ (2D problem)

Full-scale model

$$\mathbf{K}\mathbf{d} = \begin{bmatrix} \bar{\mathbf{K}} & \widehat{\mathbf{K}} \\ \widehat{\mathbf{K}}^{\mathrm{T}} & \hat{\mathbf{K}} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{d}} \\ \hat{\mathbf{d}} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{f}} \\ \hat{\mathbf{f}} \end{bmatrix} = \mathbf{f}$$

Reduced-order approximation $\mathbf{d}^r = \mathbf{P}\boldsymbol{\alpha}, \quad \boldsymbol{\alpha} \in \mathbb{R}^k, k \ll N$ Modal analysis-ROM

$$\mathbf{d}^r(x) = \sum_{I=1}^{\bar{N}} \Psi_I \bar{\mathbf{d}}_I^r + \sum_{J=1}^{\hat{N}} \mathbf{F}_J \hat{\mathbf{d}}_J^r$$

$$\mathbf{P}^T\mathbf{K}\mathbf{P}\boldsymbol{\alpha} = \mathbf{P}^T\mathbf{f}$$

Decomposed projection

$$\mathbf{P} = \begin{bmatrix} \bar{\mathbf{P}} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{P}} \end{bmatrix}$$

$$\begin{bmatrix} \bar{\mathbf{P}} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{P}} \end{bmatrix}^T \begin{bmatrix} \bar{\mathbf{K}} & \widehat{\mathbf{K}} \\ \widehat{\mathbf{K}}^{\mathrm{T}} & \hat{\mathbf{K}} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{P}} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{P}} \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{\alpha}} \\ \hat{\boldsymbol{\alpha}} \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{P}} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{P}} \end{bmatrix}^T \begin{bmatrix} \bar{\mathbf{f}} \\ \hat{\mathbf{f}} \end{bmatrix}$$

DSR-ROM ($\hat{\mathbf{P}} = \mathbf{I}$)

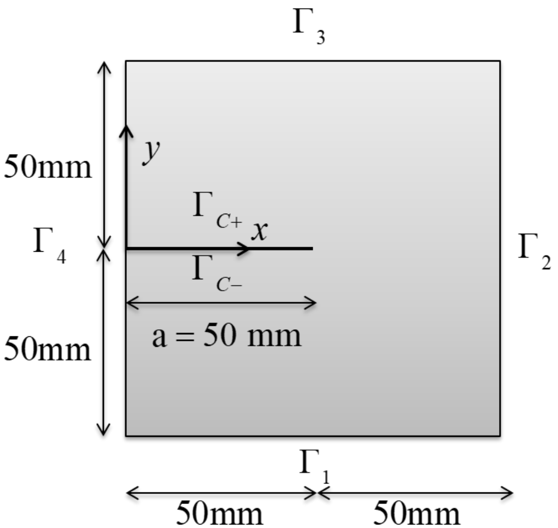
$$\begin{bmatrix} \bar{\mathbf{P}}^{\mathrm{T}}\bar{\mathbf{K}}\bar{\mathbf{P}} & \bar{\mathbf{P}}^{\mathrm{T}}\widehat{\mathbf{K}} \\ \widehat{\mathbf{K}}^T\bar{\mathbf{P}} & \hat{\mathbf{K}} \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{\alpha}} \\ \hat{\mathbf{d}}^r \end{bmatrix} = \begin{bmatrix} \tilde{\bar{\mathbf{K}}} & \bar{\mathbf{P}}^{\mathrm{T}}\widehat{\mathbf{K}} \\ \widehat{\mathbf{K}}^T\bar{\mathbf{P}} & \hat{\mathbf{K}} \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{\alpha}} \\ \hat{\mathbf{d}}^r \end{bmatrix} = \begin{bmatrix} \tilde{\bar{\mathbf{f}}} \\ \hat{\mathbf{f}} \end{bmatrix}$$

- *Sparse sub-matrix due to ISBFM-Garlerkin*
- *a low-rank representation that only reduces the smooth system and well preserves non-smooth subspace*

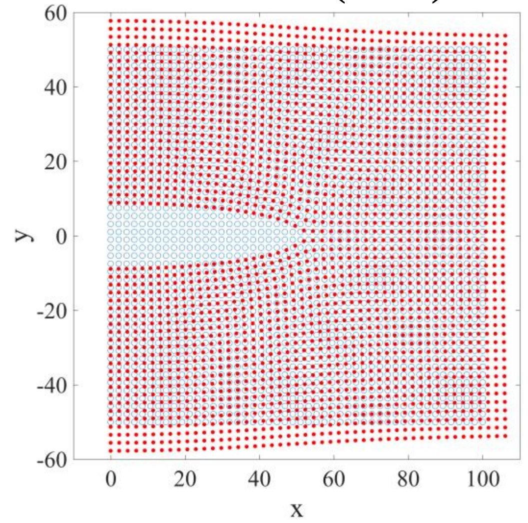
He, Q., Chen, J. S, Marodon, C., “A Decomposed Subspace Reduction for Fracture Mechanics based on the Meshfree Integrated Singular Basis Function Method”, Computational Mechanics, Vol. 63, pp. 593–614, 2019.

Reduced-Order Displacement Approximations

- Line crack model: MA (Modal analysis) v.s. DSR (Decomposed subspace reduction)



Reference (FSM)



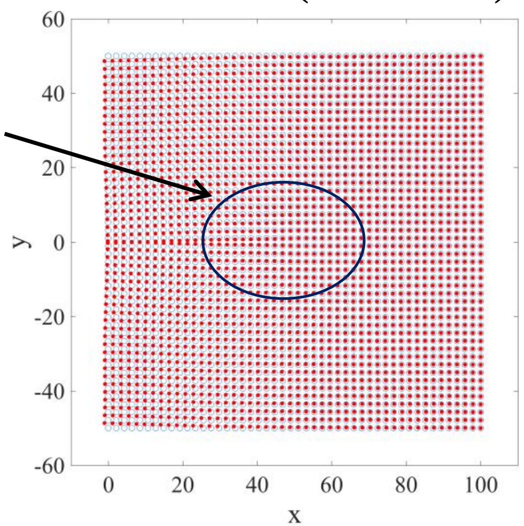
$$N = 1154\ (1152 + 2)$$

$$k = 40, \frac{k}{N} \approx 3.5\%$$

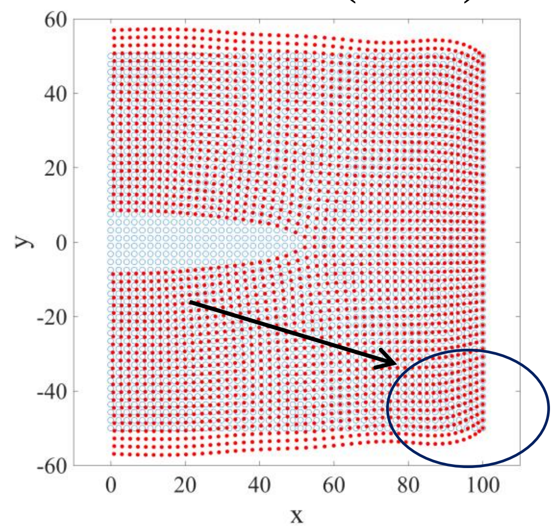
- Proper scaling

$$\boxed{\mathbf{F}_J^* = \chi \mathbf{F}_J(r^{J/2}) = \frac{1}{\gamma l_c^{J/2}} \mathbf{F}_J(r^{J/2}),\ \gamma = 50}$$

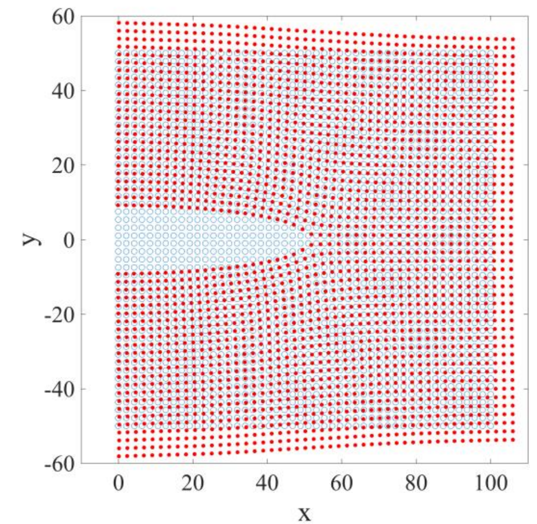
ISBFM-MA (non-scaled)



ISBFM-MA (scaled)

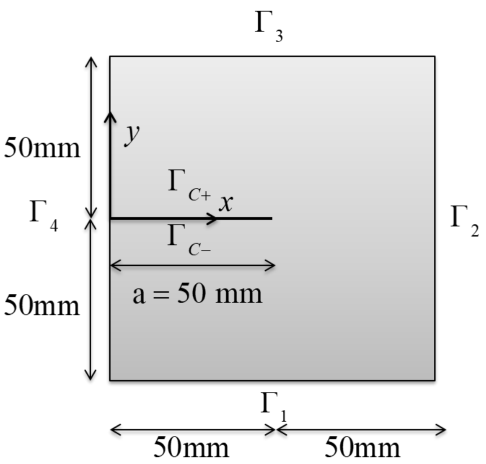


ISBFM-DSR



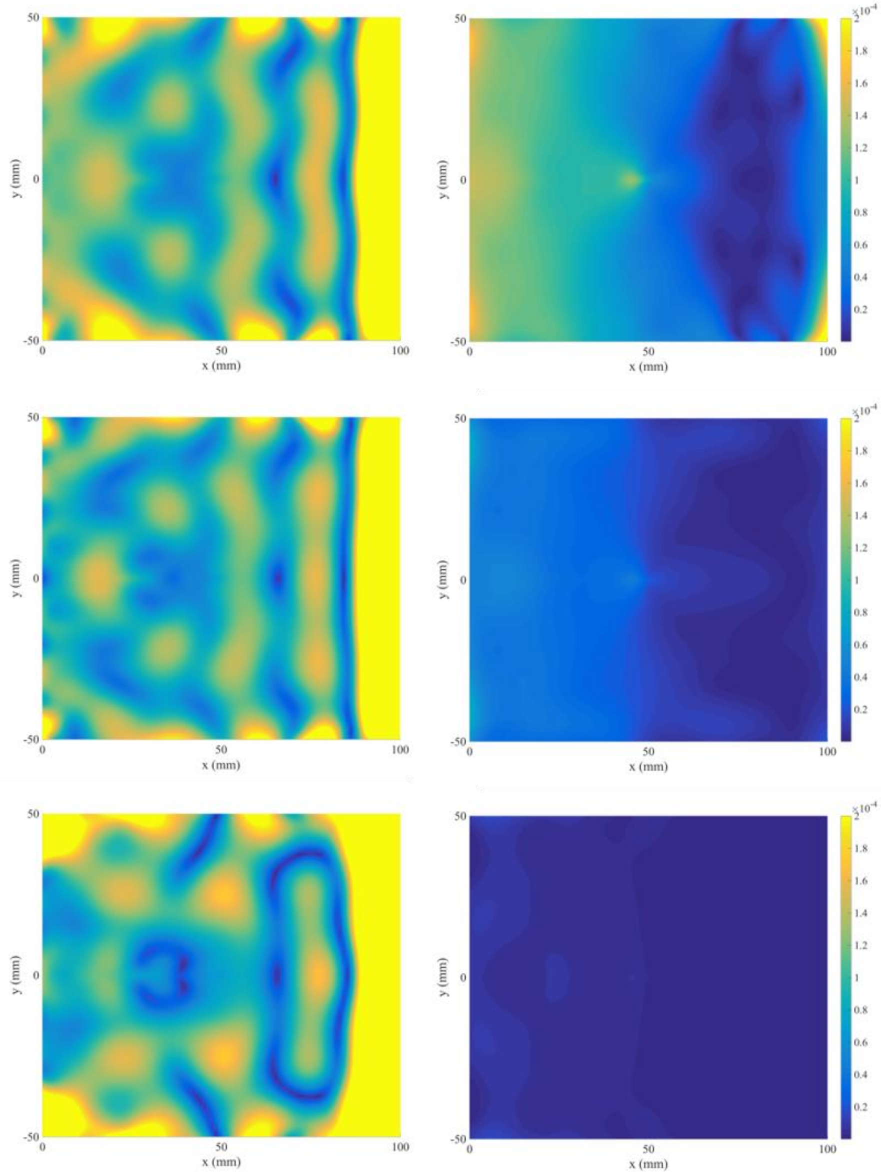
Relative Error for Reduced-Order Approximations

- Reduced-order modeling under <u>different order of enrichment functions</u> $\hat{N} = 1, 2, 5$



$36 \times 36 \; (\bar{N} = 1296)$

❖ MA Reduction ❖ DSR Reduction



$k = 80 \; (k / N \approx 3.1\%)$

$\hat{N} = 1$

$k = 80 \; (k / N \approx 3.1\%)$

$\hat{N} = 2$

$k = 80 \; (k / N \approx 3.1\%)$

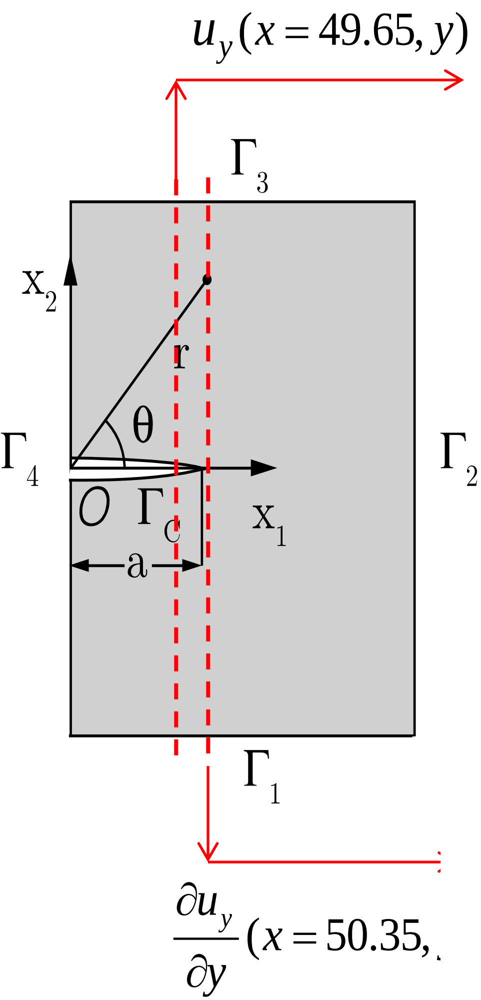
$\hat{N} = 5$

Euclidian norm of the <u>reduced-order solution</u> against the <u>fine-scale solution</u>

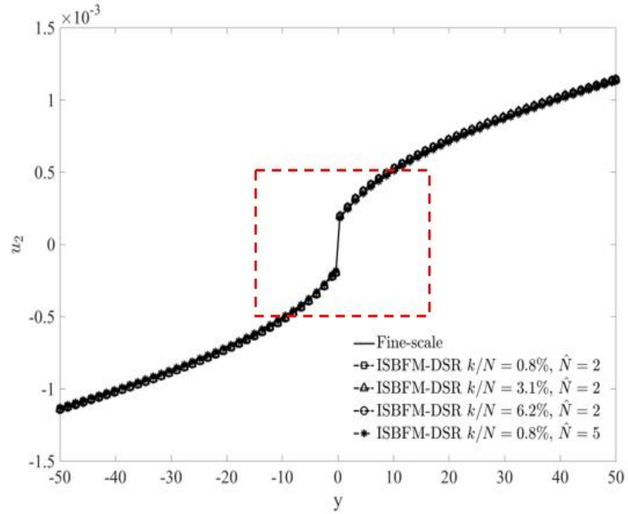
$$e_E(\mathbf{x}) = |\mathbf{u}^h(\mathbf{x}) - \mathbf{u}^r(\mathbf{x})|$$

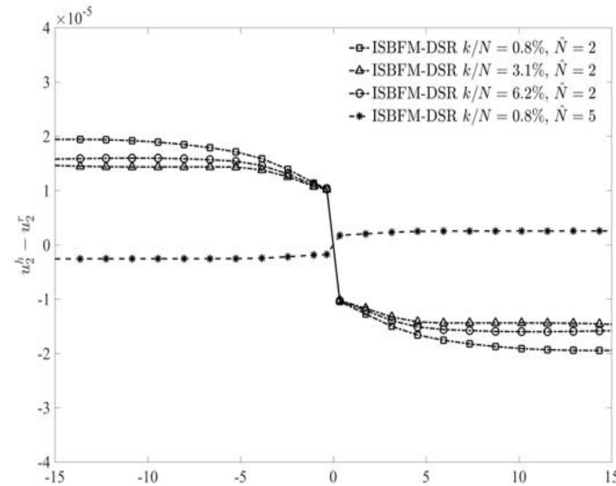
Reduced-Order Modeling of Loaded Crack Problem

- Approximation of near-tip features under different reduction ratio k/N $(k = 20, 80, 160)$

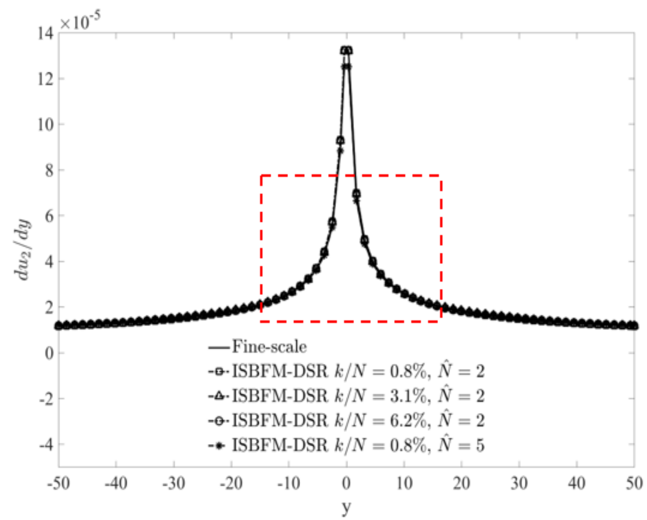


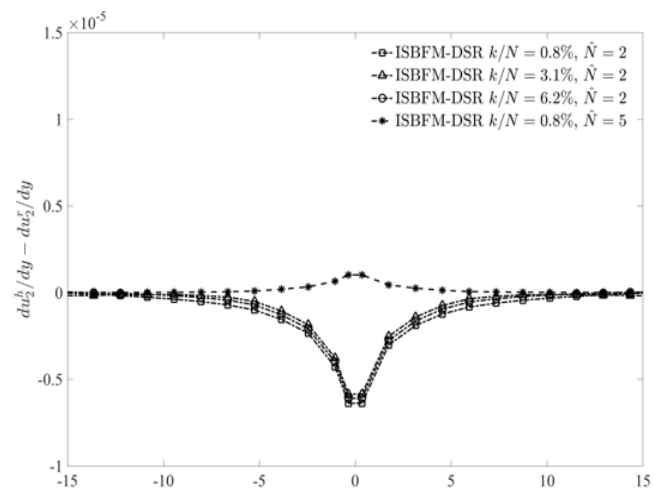
❖ DSR Reduction for discontinuity





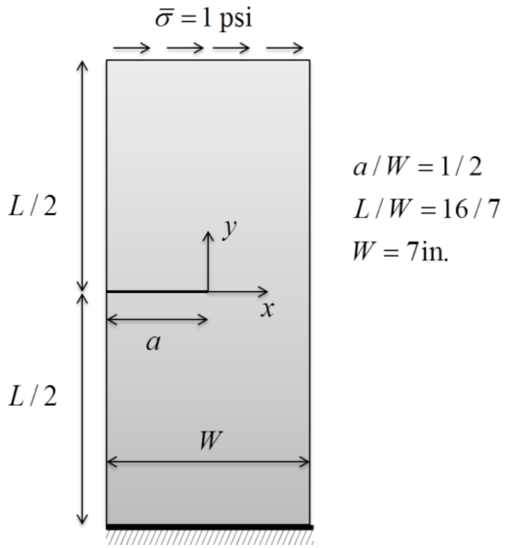
❖ DSR Reduction for near-tip singularity

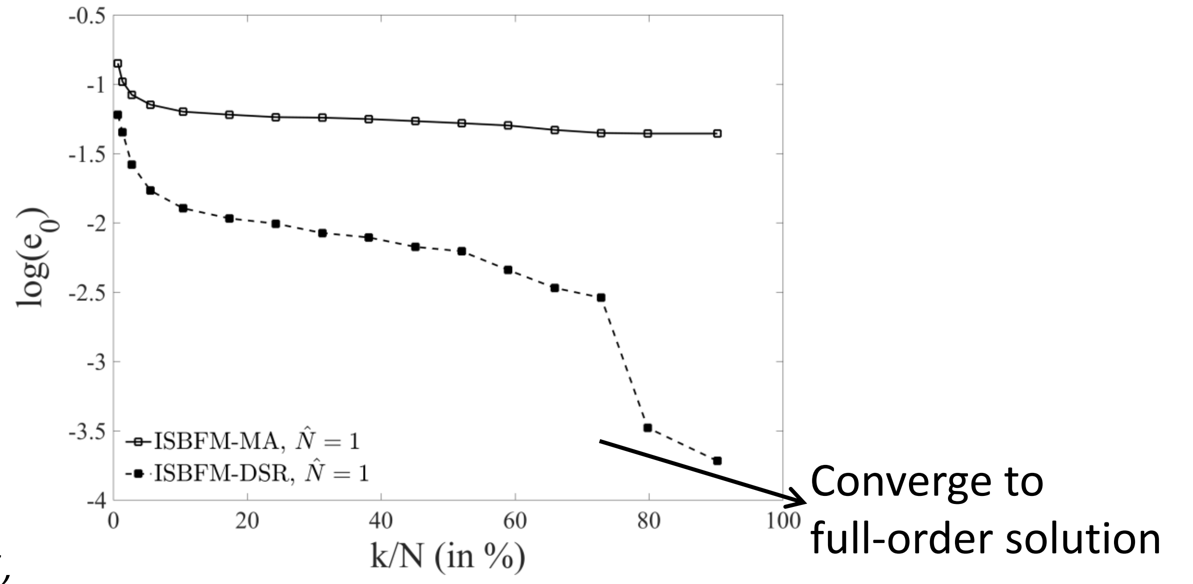




Reduced-Order Modeling of Mixed-Mode Problem

- Reduced-order modeling under <u>different percentages of reduction</u> k/N $(k = 20, 80, 160)$





(Fleming, Chu, Moran, Belytschko 1997,

- Stress intensity factors by DSR reduced order modeling

Reference stress intensity factors: $K_I = 34.0$ psi$\sqrt{\text{in}}$, $K_{II} = 4.55$ psi$\sqrt{\text{in}}$				
	e_{KI}		e_{KII}	
k/N	ISBFM-MA	ISBFM-DSR	ISBFM-MA	ISBFM-DSR
2.8%	6.0×10^{-2}	1.7×10^{-2}	4.1×10^{-1}	1.2×10^{-1}
38.2%	5.3×10^{-2}	4.3×10^{-3}	9.8×10^{-2}	4.0×10^{-2}
90.2%	1.5×10^{-2}	8.9×10^{-6}	9.5×10^{-2}	4.1×10^{-4}

Nonlinear Model Reduction (Hyper-Reduction)

POD-Galerkin

$$\boldsymbol{V}^T \boldsymbol{f}_{\text{int}}(\boldsymbol{V}\boldsymbol{u}^r(\boldsymbol{\mu})) - \boldsymbol{V}^T \boldsymbol{f}_{\text{ext}}(\boldsymbol{\mu}) = \boldsymbol{0},$$

$$\boldsymbol{f}_{\text{int}}(\boldsymbol{u}(\boldsymbol{\mu})) = \boldsymbol{A}\boldsymbol{u}(\boldsymbol{\mu}) + \boxed{\boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu})),}$$

Nonlinear function w.r.t. state

"Lifting-bottleneck"

$$\hat{\boldsymbol{f}}(\boldsymbol{u}^r) := \boldsymbol{V}^T \boldsymbol{f}(\boldsymbol{V}\boldsymbol{u}^r) = \sum_{g=1}^{N_g} \boldsymbol{V}_{\mathcal{I}_g}^T \boldsymbol{f}_g(\boldsymbol{V}\boldsymbol{u}^r) w_g, \quad \sim O(\alpha(n) + 2nk) \text{ flops}$$

- The nonlinear term is a function of the unknown (cannot precomputed)
- The online computational cost scales with the underlying discretization

System Approximation

POD approximation of nonlinear terms (internal force vector, residual vector, etc.)

- Nonlinear snapshots : $\boxed{\mathbf{X}_f = [\boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu}_1)), \ldots, \boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu}_{N_s}))] \in \mathbb{R}^{\mathcal{N} \times N_s}}$

Collateral POD basis: $\boxed{\boldsymbol{Z} = [\boldsymbol{z}_1, \ldots, \boldsymbol{z}_{\hat{k}}] \in \mathbb{R}^{\mathcal{N} \times \hat{k}} \quad (\hat{k} \ll \mathcal{N})}$

Attemp: POD approximation $\boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu})) \approx \boldsymbol{Z}\boldsymbol{c}(\boldsymbol{\mu}) = \sum_{i=1}^{\hat{k}} \boldsymbol{z}_i c_i(\boldsymbol{\mu}).$

Kaneko, S., Wei, H., He, Q., Chen, J. S. and Yoshimura, S., "A Hyper-reduction Computational Method for Accelerated Modeling of Thermal Cycling-Induced Plastic Deformations," Journal of the Mechanics and Physics of Solids, Vol. 151, 104385, 2021. https://doi.org/10.1016/j.jmps.2021.104385

"Gappy"-type interpolation method [Everson and Sirovich 1995]

Discrete empirical interpolation method (DEIM)

$$\boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu})) \approx \boldsymbol{Z}\boldsymbol{c}(\boldsymbol{\mu}) = \sum_{i=1}^{\hat{k}} \boldsymbol{z}_i c_i(\boldsymbol{\mu}).$$

Sample a few entries using selection matrix

$$\boxed{\boldsymbol{P}^T} \boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu})) = \boxed{\boldsymbol{P}^T} \boldsymbol{Z}\boldsymbol{c}(\boldsymbol{\mu}), \quad \Longrightarrow \quad \boldsymbol{c}(\boldsymbol{\mu}) = \boxed{(\boldsymbol{P}^T\boldsymbol{Z})^{-1}\boldsymbol{P}^T} \boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu}))$$

$$\boldsymbol{P} = [\boldsymbol{e}_{\wp_1}, \ldots, \boldsymbol{e}_{\wp_{\hat{n}}}] \in \mathbb{R}^{\mathcal{N}\times\hat{n}} \qquad e_{\wp i} = [0,\ldots,0,1,0,\ldots,0]^T \in \mathbb{R}^n$$

DEIM approximation

$$\boxed{\boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu})) \approx \tilde{\boldsymbol{f}}(\boldsymbol{u}(\boldsymbol{\mu})) = \boldsymbol{Z}\boldsymbol{c}(\boldsymbol{\mu}) = \boldsymbol{Z}(\boldsymbol{P}^T\boldsymbol{Z})^{-1}\boldsymbol{P}^T\boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu})) = \boldsymbol{Z}\hat{\boldsymbol{Z}}^{-1}\boldsymbol{P}^T\boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu})),}$$

"Gappy"-POD

$$\boldsymbol{c} = \arg\min_{\boldsymbol{c}\in\mathbb{R}^{\hat{k}}} \left\| \boldsymbol{P}^T\boldsymbol{f} - \boldsymbol{P}^T\boldsymbol{Z}\boldsymbol{c} \right\|_2.$$

$$\boxed{\boldsymbol{f} \approx \boldsymbol{Z}\boldsymbol{c} = \boldsymbol{Z}(\boldsymbol{P}^T\boldsymbol{Z})^{\dagger}\boldsymbol{P}^T\boldsymbol{f}.}$$

Greedy Algorithm: determine the selection matrix $\boldsymbol{P}$ based on the empirical bases $\boldsymbol{Z}$

Barrault et al. 2004, Chaturantabut and Sorensen, 2010, Carlberg et al. 2011

Greedy Algorithm

INPUT: $\{z_l\}_{l=1}^{m}$ linearly independent POD basis

OUTNPUT: $\vec{\wp} = [\wp_1, \ldots, \wp_m]^T \in \mathbb{R}^m$

1. $[|\rho|, \wp_1] = \max\{|z_1|\}$
2. $\boldsymbol{Z} = [z_1]$, $\boldsymbol{P} = [\boldsymbol{e}_{\wp_1}]$, $\vec{\wp} = [\wp_1]$
3. **for** $l = 2$ to m **do**

   solve $(\boldsymbol{P}^T \boldsymbol{Z})\boldsymbol{c} = \boldsymbol{P}^T z_l$ for $\boldsymbol{c}$

   $\boldsymbol{r} = z_l - \boldsymbol{Z}\boldsymbol{c}$

   $[|\rho|, \wp_l] = \max\{|\boldsymbol{r}|\}$

   $\boldsymbol{Z} \leftarrow [\boldsymbol{Z}\ z_l]$, $\boldsymbol{P} \leftarrow [\boldsymbol{P}\ \boldsymbol{e}_{\wp_l}]$, $\vec{\wp} \leftarrow \left[\vec{\wp}^T \quad \wp_l\right]^T$
4. **end for**

Idea: The DEIM algorithm selects an index (DOF of the discretization) at each iteration to limit growth of an error bound.

- Error bound $\|\boldsymbol{f} - \tilde{\boldsymbol{f}}\|_2 \le c\,\xi_{\boldsymbol{Z}^\perp}(\boldsymbol{f})$,

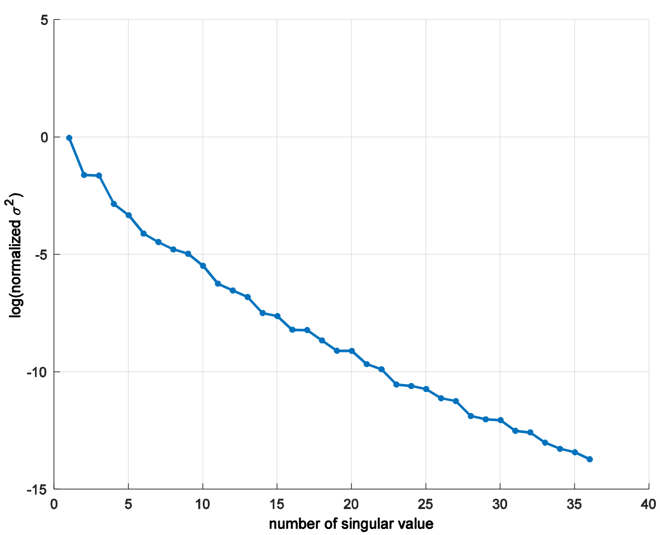
  where $\xi_{\boldsymbol{Z}^\perp}(\boldsymbol{f}) = \|(\mathbf{I} - \boldsymbol{Z}\boldsymbol{Z}^T)\boldsymbol{f}\|_2$ and $c = \|(\boldsymbol{P}^T\boldsymbol{Z})^{-1}\|_2$

Chaturantabut and Sorensen, 2010

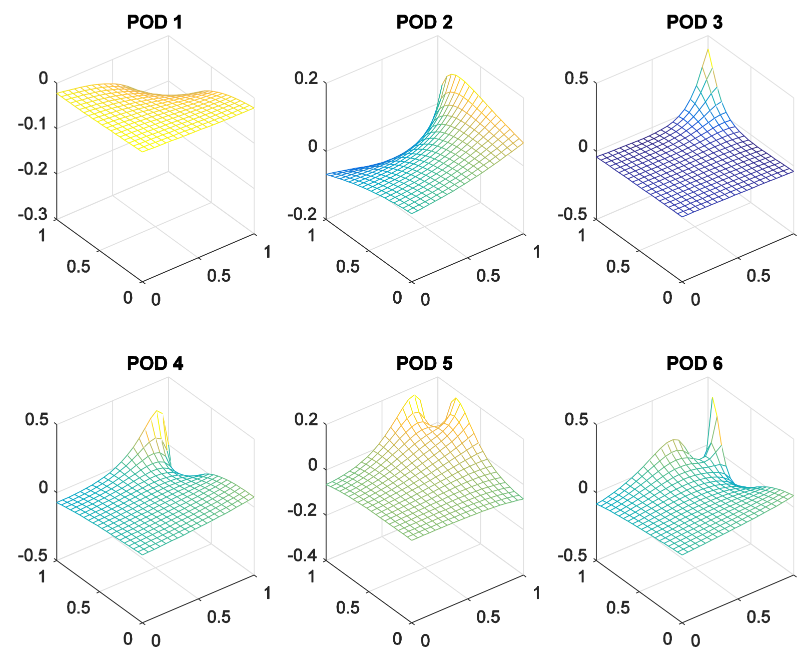
2D Nonlinear Parameterized Function

The nonlinear function in W$=[0,1]^2$ is discretized by $n = 20$´ 20 equidistant grid and sampled on a $N_s = 25$´ 25 equidistant grid in parameter domain $\mathcal{D} = [0,1]^2$

$$g^1(\boldsymbol{x};\mu) = \frac{1}{\sqrt{((1-x_1)-(0.99\mu_1-1))^2+((1-x_2)-(0.99\mu_2-1))^2+0.1^2}}.$$

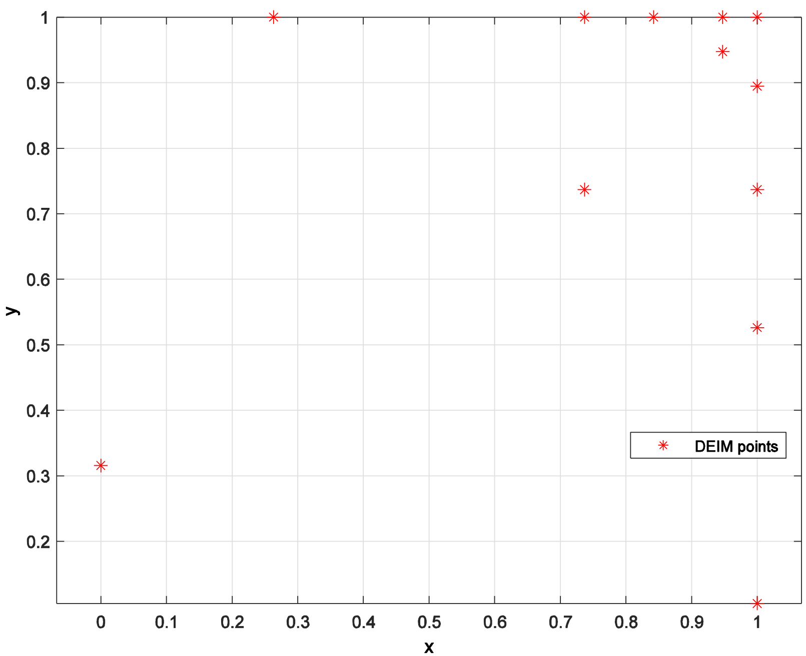


Normalized singular value $\sigma_i^2 / (\sum_i \sigma_i^2)$

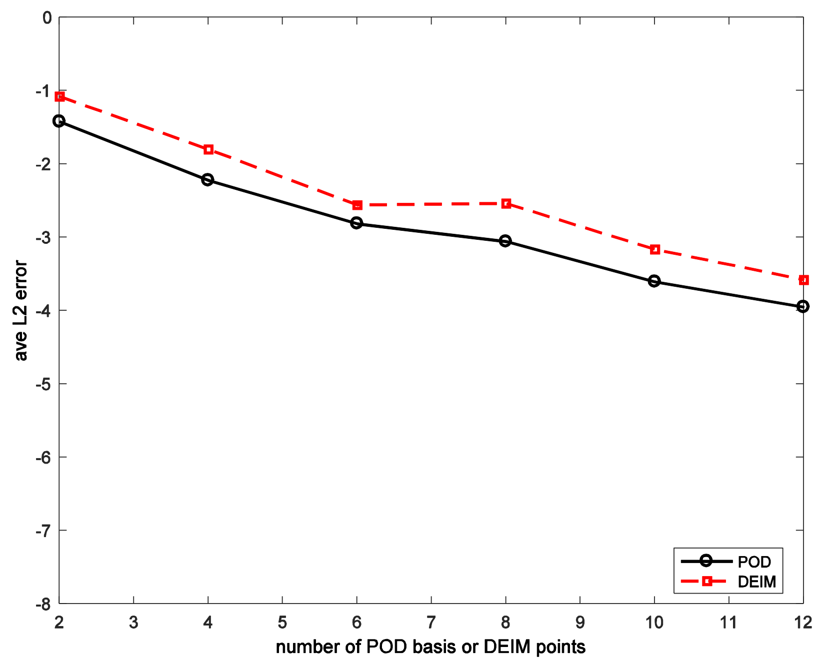


POD modes

2D Nonlinear Parameterized Function



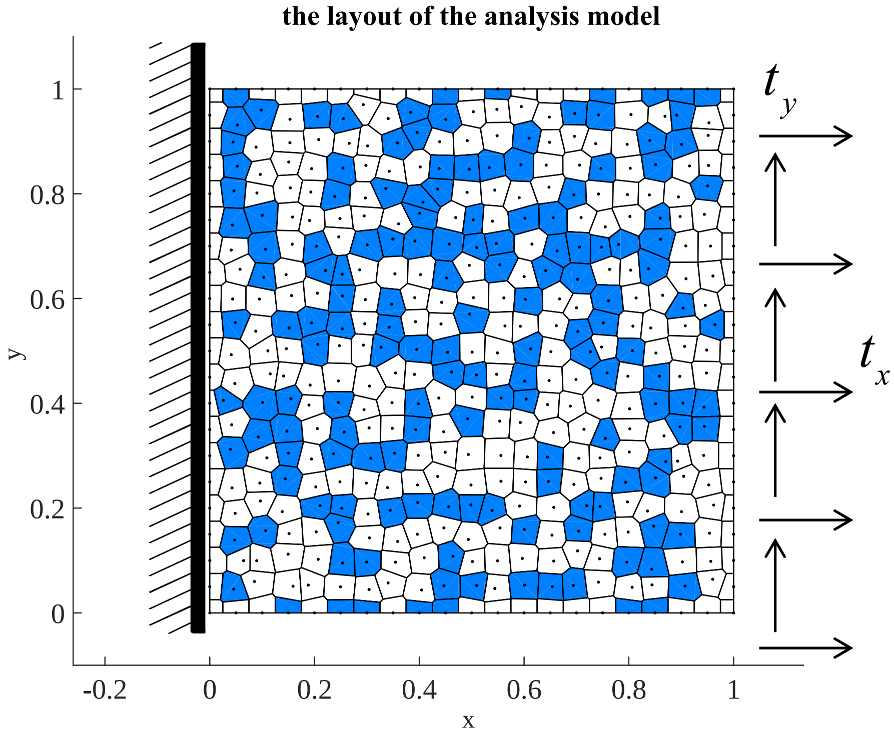
12 DEIM points selected by greedy algorithm "sensor"



Average L2 error of POD and DEIM approximation for the training data

$$\hat{k} / \mathcal{N} \approx 3\% \ (\hat{k} = 12)$$

Two-Phase Hyperelastic Material



	Mat. 1 (White)	Mat. 2(Blue)
E	3.0e7	1.5e6
v	0.49	0.4

$n = 441$

$n_s = 20$

$n_{test} = 10$ (not in the snapshot samples)

$$(x, y) \in \Omega = [0,1]^2 \subset \mathbb{R}^2$$

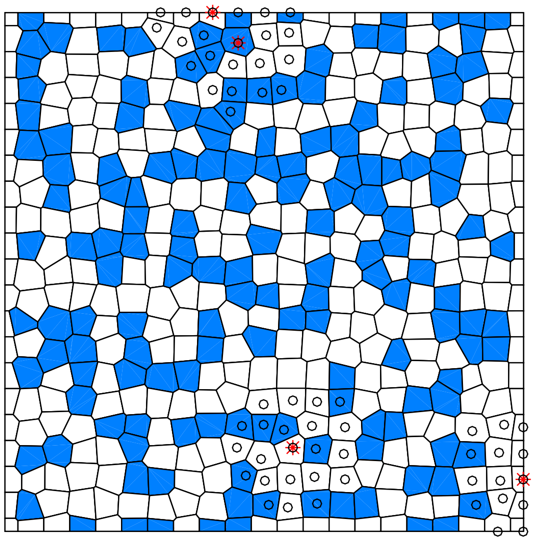
$$\boldsymbol{\mu} = (t_x, t_y) \in \mathcal{D}$$

$$\mathcal{D} = [-1\times 10^4, 1\times 10^4] \times [-2\times 10^5, 2\times 10^5] \subset \mathbb{R}^2$$

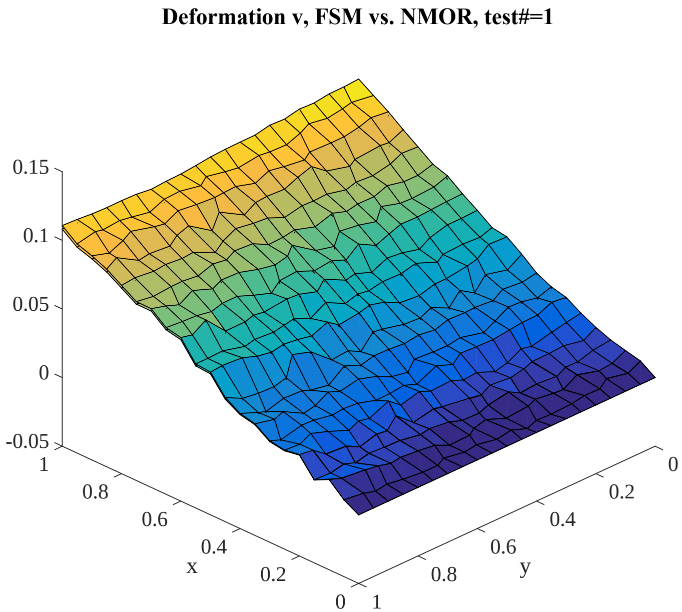
Average relative error:

$$\varepsilon_r = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \frac{\| \boldsymbol{d}(\mu_i) - \tilde{\boldsymbol{d}}(\mu_i) \|}{\| \boldsymbol{d}(\mu_i) \|},$$

Two-Phase Hyperelastic Material



DEIM points at SCNI cells



	$k = 2, \hat{k} = 6, N_{DEIM}/N = 13\%$	
	Normalized CPU time	Average relative error
POD	38.3%	1.52×10^{-1}
POD-DEIM	15.2%	1.97×10^{-1}

Robust Nonlinear MOR via Manifold Learning

Full-order model $$\boldsymbol{A}\boldsymbol{u} + \boldsymbol{f}(\mu, \boldsymbol{u}) = \boldsymbol{0},$$

Collect state vectors in offline stage $\boldsymbol{X}_s = [\boldsymbol{u}(\boldsymbol{\mu}_1), \ldots, \boldsymbol{u}(\boldsymbol{\mu}_{N_s})] \in \mathbb{R}^{\mathcal{N} \times N_s}$

POD Approx. (Recon.) $$\boldsymbol{u}(\boldsymbol{\mu}) \approx \tilde{\boldsymbol{u}}(\boldsymbol{\mu}) = \boldsymbol{V}\boldsymbol{u}^r(\boldsymbol{\mu}) = \sum_{i=1}^{k} \boldsymbol{v}_i u_i^r(\boldsymbol{\mu}), \quad \widetilde{\boldsymbol{X}}_S = \boldsymbol{V}\boldsymbol{V}^T \boldsymbol{X}_S$$

POD ROM $$\boldsymbol{V}^T \boldsymbol{A}\boldsymbol{V}\boldsymbol{u}^r + \boldsymbol{V}^T \boldsymbol{f}(\mu, \boldsymbol{V}\boldsymbol{u}^r) = \boldsymbol{0}.$$

Nonlinear Model Order Reduction

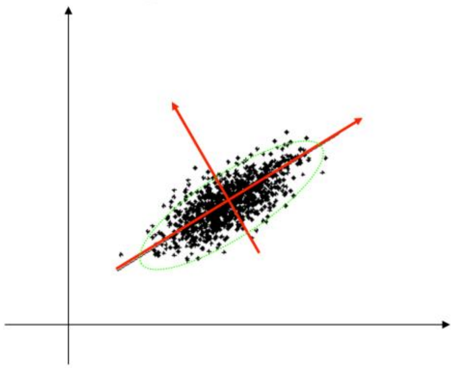
Collect both state and nonlinear vectors offline $\mathbf{X}_f = [\boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu}_1)), \ldots, \boldsymbol{f}(\boldsymbol{u}(\boldsymbol{\mu}_{N_s}))] \in \mathbb{R}^{\mathcal{N} \times N_s}$

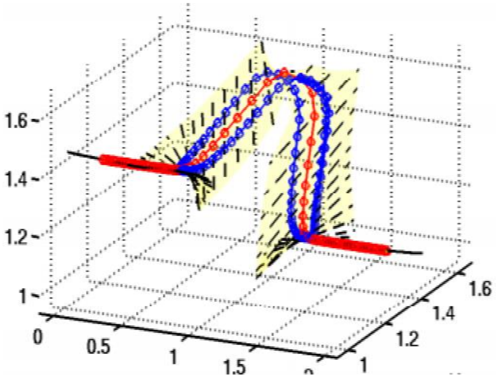
DEIM Approx. $$\boldsymbol{f}(\boldsymbol{u}(\mu)) \approx \tilde{\boldsymbol{f}}(\boldsymbol{u}(\mu)) = \boldsymbol{Z}(\boldsymbol{P}^T \boldsymbol{Z})^{\dagger} \boldsymbol{P}^T \boldsymbol{f}(\boldsymbol{u}(\mu)) = \boldsymbol{Z}\hat{\boldsymbol{Z}}^{\dagger} \boldsymbol{P}^T \boldsymbol{f}(\boldsymbol{u}(\mu)),$$

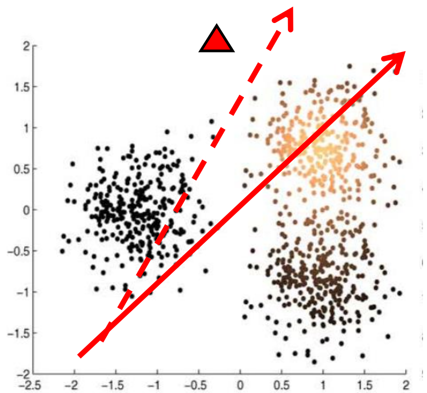
POD-DEIM ROM $$\boldsymbol{V}^T \boldsymbol{A}\boldsymbol{V}\boldsymbol{u}^r + \boldsymbol{V}^T \boldsymbol{Z}\hat{\boldsymbol{Z}}^{\dagger} \boldsymbol{P}^T \boldsymbol{f}(\mu, \boldsymbol{V}\boldsymbol{u}^r) = \boldsymbol{0}$$

Limitations of POD based DEIM

- POD method only works well for data that is ***Gaussian*** or lying on a ***“flat” manifold***. It is very ***sensitive to outliers*** that does not follow the overall statistical model.







Multi-Gaussian
(Coifman & Lafon 2006)

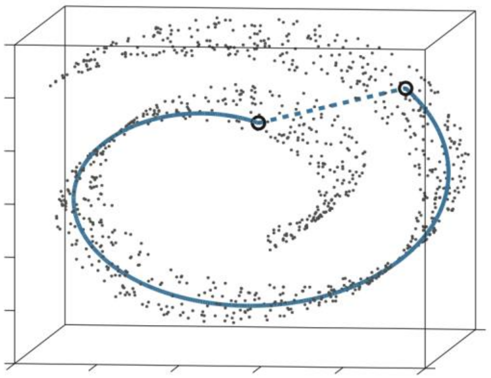
- Knowing that the manifold of the snapshots $\boldsymbol{X}_f$ of nonlinear function is much more “nonlinear” than that of the state snapshots $\boldsymbol{X}_s$, can we design a better projection $\mathbf{Z}$ than that obtained from POD ?

⇨ *Objective: a more robust ROM dealing with* mechanics systems that <u>exhibit a wide variety of parameter-dependent nonlinear behaviors</u>

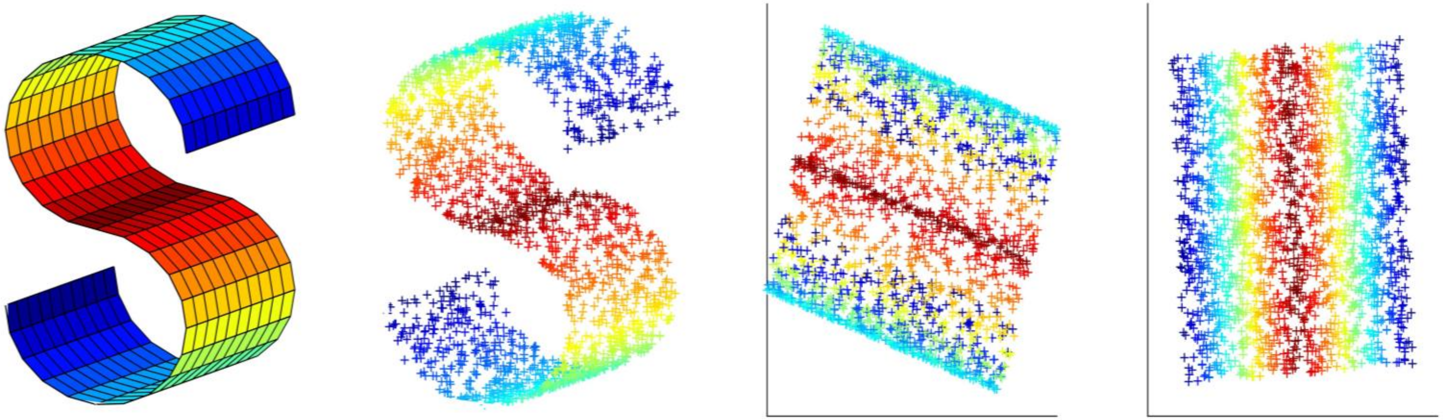
Manifold Learning (Nonlinear Dimensionality Reduction)

- Manifold learning to find the low-dimensional representation

  Given data that lie in a ***non-Euclidean*** space, find an ***embedding*** into Euclidean space that ***preserves as much of the geometry*** as possible



- S-Curve example



(a) Two-dimensional manifold structure represented by the three-dimensional S-curve data set

(b) Two-dimensional embedding obtained by PCA/POD

(c) Two-dimensional embedding obtained by manifold learning

Q. He, J. S. Chen "A Physics-Constrained Data-Driven Approach Based on Locally Convex Reconstruction for Noisy Database," CMAME, 363, 112791, 2020

Manifold Learning: Graph Embedding

"**Spectral Graph Theory**" [Chung 1997]

- Given N points $\{x_i\}_{i=1}^N \in \mathbb{R}^n$, for each x_i , we construct a **weighted graph** $G = (V, E)$ with $\boxed{k \text{ neighbor points, } \{x_j\}_{j=1}^k,\ j \in \mathcal{N}_k(x_i)}$

k-NN

Seek for a "faithful" low dimensional representation $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N] \in \mathbb{R}^{d \times N}$

If possible, provides the nonlinear mapping $f : \mathcal{X} \to \mathcal{Y}$ (if linear, $\mathbf{y}_i = \mathbf{Z}^T(\mathbf{x}_i - c)$)

$\boxed{\text{such that if } x_i \text{ and } x_j \text{ are close to each other, then so are } y_i \text{ and } y_j}$

- Solve minimization problem (Hall's energy [Koren et al. 2002])

$$\boldsymbol{Y}^* = \arg\min_{\boldsymbol{YBY}^T = \boldsymbol{C}} \sum_{i,j}^N \boxed{w_{ij}} \left\| \boldsymbol{y}_i - \boldsymbol{y}_j \right\|^2 = \arg\min_{\boldsymbol{YBY}^T = \boldsymbol{C}} \operatorname{trace}(\boldsymbol{YLY}^T),$$

where $\mathbf{L} = \mathbf{D} - \mathbf{W}$, and $\mathbf{W}$ is a user-defined similarity matrix that better represents the nonlinear structure of input data

$\boldsymbol{D}$ be a diagonal matrix with diagonal entries $d_{jj} = \sum_i w_{ij}$

Intepretation: preserves the locality of data $\{x_i\}_{i=1}^N$ in embedding graph

Belkin and Niyogi, 2002; Yan et al. 2005

Linear Graph Embedding (LGE) Framework

- Assume linear mapping $\mathbf{Y} = \mathbf{Z}^T \mathbf{X}$, we derive the projection from $\mathcal{E}(\boldsymbol{Y}) = \sum_{i,j}^{N} w_{ij} \left\| \boldsymbol{y}_i - \boldsymbol{y}_j \right\|^2,$

$$\Rightarrow \quad \boldsymbol{Z}_{\text{LGE}}^* = \arg\min_{\mathbf{Z}^T\mathbf{Z}=\mathbf{I}} \sum_{i,j}^{N} w_{ij} \left\| \boldsymbol{Z}^T \boldsymbol{x}_i - \boldsymbol{Z}^T \boldsymbol{x}_j \right\|^2 = \arg\max_{\mathbf{Z}^T\mathbf{Z}=\mathbf{I}} \operatorname{trace}(\boldsymbol{Z}^T \boldsymbol{X} \boldsymbol{L} \boldsymbol{X}^T \boldsymbol{Z}),$$

- Some optional choices of weight function
  - Inverse distance weight $\quad w(dist(\boldsymbol{x}_i, \boldsymbol{x}_j)) = \dfrac{1}{dist(\boldsymbol{x}_i, \boldsymbol{x}_j)^p},$
  - Gaussian weight $\quad w(dist(\boldsymbol{x}_i, \boldsymbol{x}_j)) = \exp\left(-\dfrac{dist(\boldsymbol{x}_i, \boldsymbol{x}_j)^2}{2\sigma^2}\right),$

- Link to POD (or Principal component analysis)

$$\boldsymbol{Z}_{\text{PCA}}^* = \arg\max_{\mathbf{Z}^T\mathbf{Z}=\mathbf{I}} \operatorname{trace}(\boldsymbol{Z}^T \boldsymbol{X} \boldsymbol{H} \boldsymbol{X}^T \boldsymbol{Z}), \quad \Leftrightarrow \quad \boldsymbol{Z}_{\text{PCA}}^* = \arg\min_{\mathbf{Z}^T\mathbf{Z}=\mathbf{I}} \sum_{i \neq j}^{N} \frac{1}{N} \left\| \boldsymbol{Z}^T \boldsymbol{x}_i - \boldsymbol{Z}^T \boldsymbol{x}_j \right\|^2.$$

$$\boldsymbol{H} = \boldsymbol{I} - \frac{1}{N} \boldsymbol{1}\boldsymbol{1}^T = \text{mean operator}$$

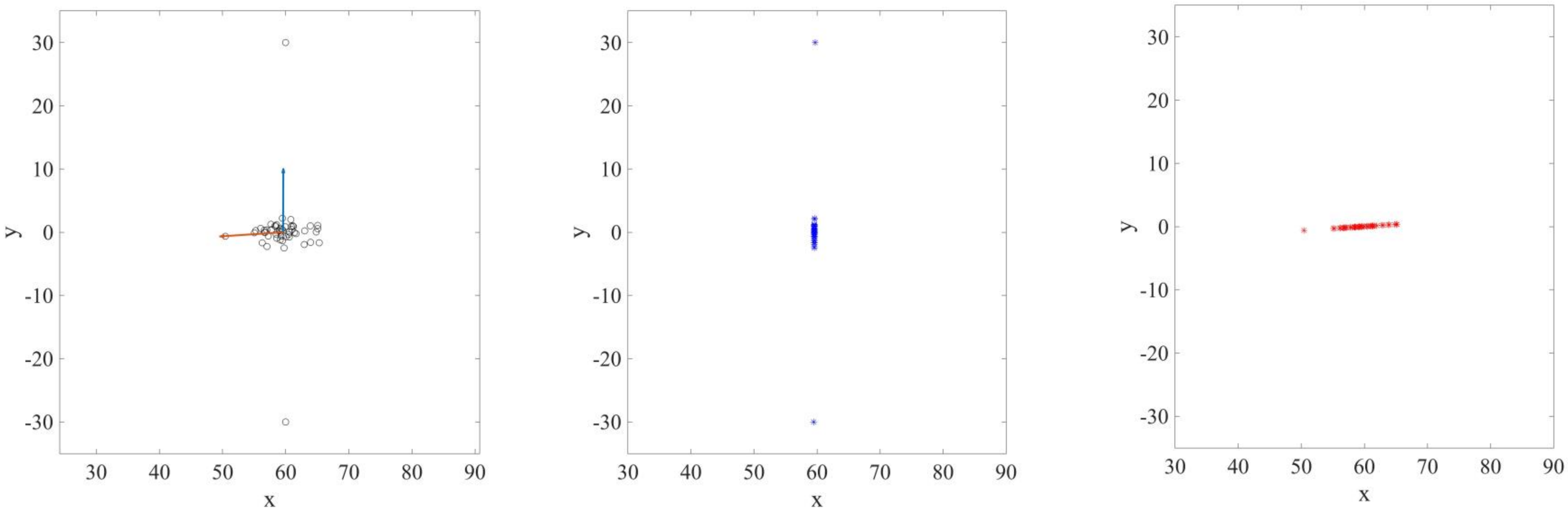
Observations

✓ *POD is a <u>special case</u> of LGE with uniform weights*

✓ *LGE provides a <u>general</u> framework (locality and weights) to consider a priori knowledge of data to enhance the resulting reduced-order projection*

Belkin and Niyogi, 2002; Yan et al. 2005

Example: POD vs. LGE Projection



(a) original data

(b) POD reconstruction

$\varepsilon_r = 3.57 \times 10^{-2}$

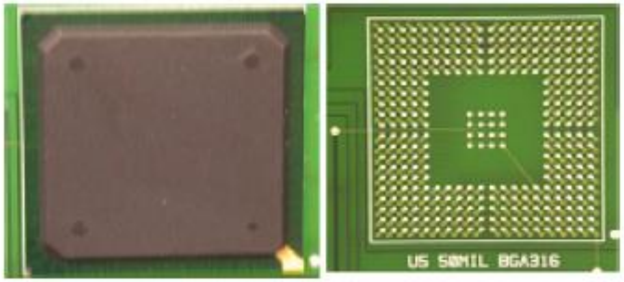
(c) LGE reconstruction

$\varepsilon_r = 3.09 \times 10^{-2}$

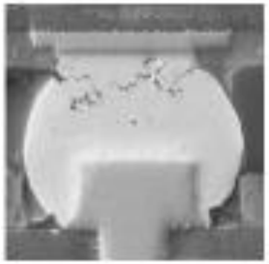
Remarks:

1. One drawback of the POD learning methods is that it is based on **least squares estimation** techniques and known for its **extreme sensitivity to "outliers"**
2. A "robust" learning method is the one that can tolerate some percentage of outlying data without having the solution severely skewed from the desired solution.
3. It motives the development of LGE to **improve the outlier robustness**, which better **preserves the nonlinear data structure**.

Thermal fatigue of solder joints



IC package [1]



Crack path [2]

Main reason of failure
→thermal fatigue

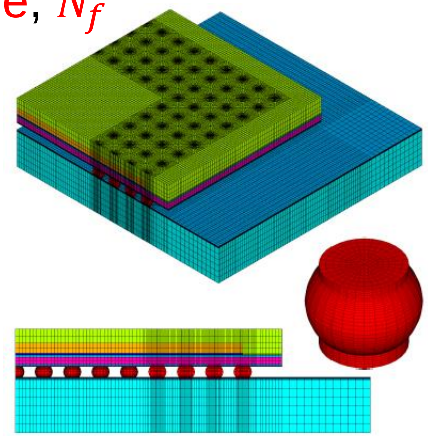
→Mismatch of thermal expansion induces thermal stress in solder joints

Fatigue analysis : estimation of numbers of loading cycles to failure, N_f

Standard procedures

1. Numerical simulation
2. Stress/strain calculation
3. Estimate N_f

Relation is defined by a fatigue model



Typical model of IC package [2]

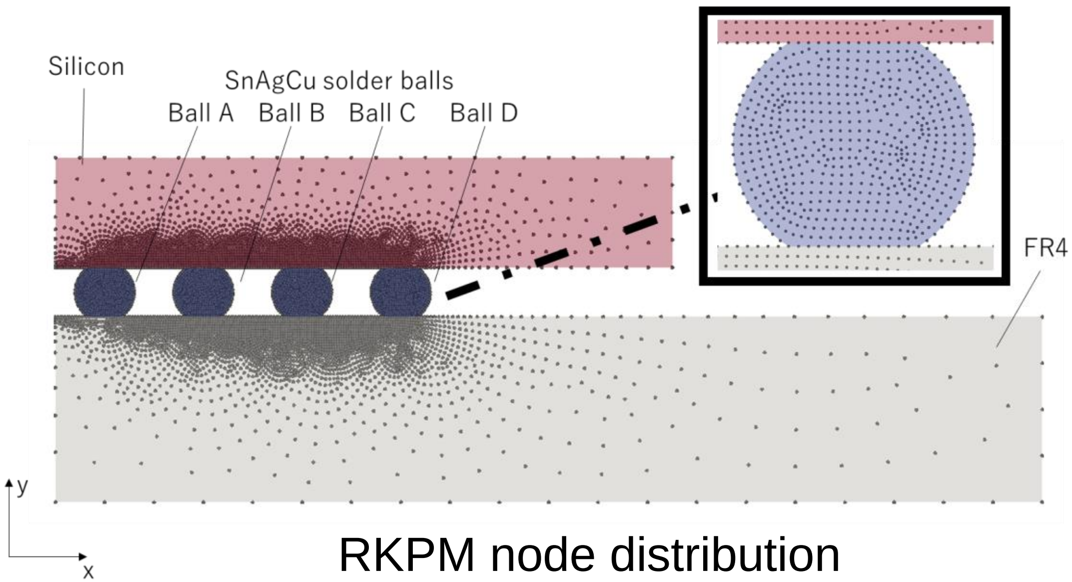
The first procedure is time consuming

- Model solder joints as nonlinear material (Visco-plastic)
- 3D large-scale simulations for many thermal cycles

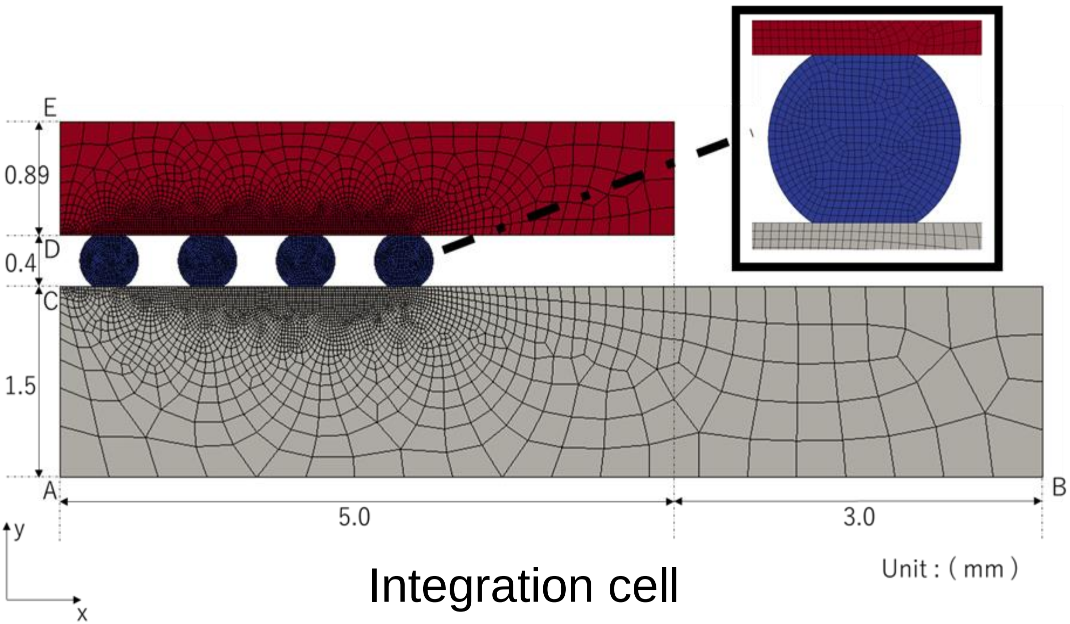
Employment of reduced-order modeling (ROM) with Hyper Reduction techniques to enhance efficiency

Q. He, J. S. Chen "A Physics-Constrained Data-Driven Approach Based on Locally Convex Reconstruction for Noisy Database," CMAME, 363, 112791, 2020

Example of fatigue analysis with ROM techniques



RKPM node distribution



Integration cell

<u>Flip chip assembly</u>

<u>Nodal distribution</u>

-number of nodes 7,419

<u>Integration cell</u>

-number of cells 7,159

-- 2,738 in silicon

-- 2,231 in SnAgCu solder balls

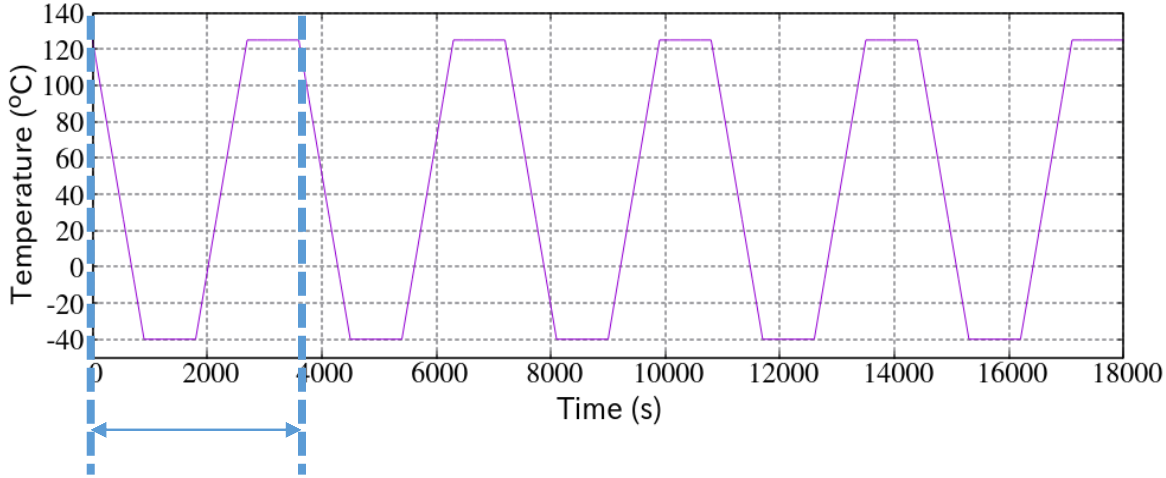
-- 2,190 in FR4

<u>Essential boundary condition</u>

-AB ... $u_y = 0$

-AC and DE ... $u_x = 0$

Problem Statement



Dwell time is 15 min
Ramp rate is 11 ^{o}C/min

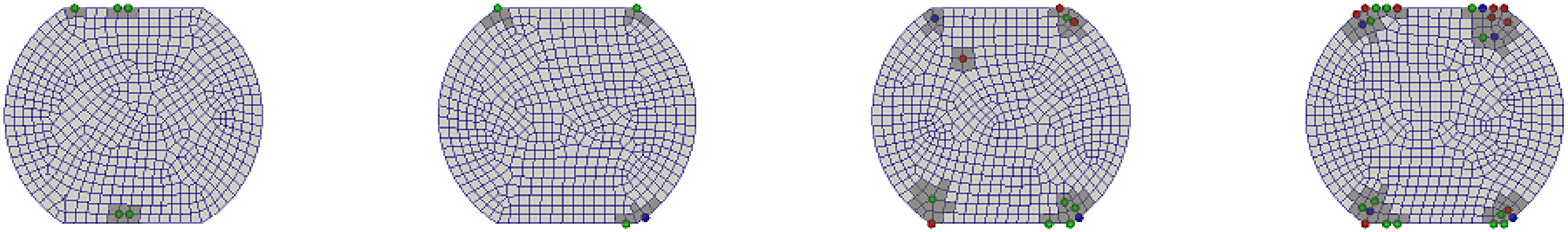
The time step size is 180 s

1^{st} Cycle (20 timesteps, 0 ~ 3,600 s)

1)For collecting snapshot matrices $\mathbf{X}_s$ and $\mathbf{X}_Q$, we run high-fidelity analysis for the 1st thermal cycle (0 s ~ 3,600 s, 20 timesteps).
2)Apply SVD to $\mathbf{X}_s$ and $\mathbf{X}_Q$ and select the first k and l basis from left singular vectors, we obtain $\mathbf{V}_k$ and $\mathbf{Z}_l$
In this study, $k = 6, l = 30$
3)Define index matrix $\mathbf{P}_{\mathrm{D}} \in \mathbb{R}^{n \times m}$ by using greedy algorithm
In this study, the number of the selected DOFs $m = 60$

For comparison, we run **high-fidelity analysis**, **POD-Galerkin-based analysis**, and **Gappy-POD-based analysis** for the 5 cycles (0 s ~ 18,000 s, 100 timesteps)

Visualization of the selected DoFs



Ball A Ball B Ball C Ball D

blue points represent nodes for which x-component of $\boldsymbol{Q}_{NL}$ is evaluated
green points represent nodes for which y-component of $\boldsymbol{Q}_{NL}$ is evaluated
red points represent nodes for which both x- and y-components of $\boldsymbol{Q}_{NL}$ are evaluated

In this study, $m = 60$, so 60 components are chosen in Gappy-POD procedure
→ During the non-linear N-R iteration, we do not need to perform domain integration over the entire domain (2,231 cells)
→Dark gray cells (131 cells) are used to evaluate the non-linear internal force and tangent stiffness

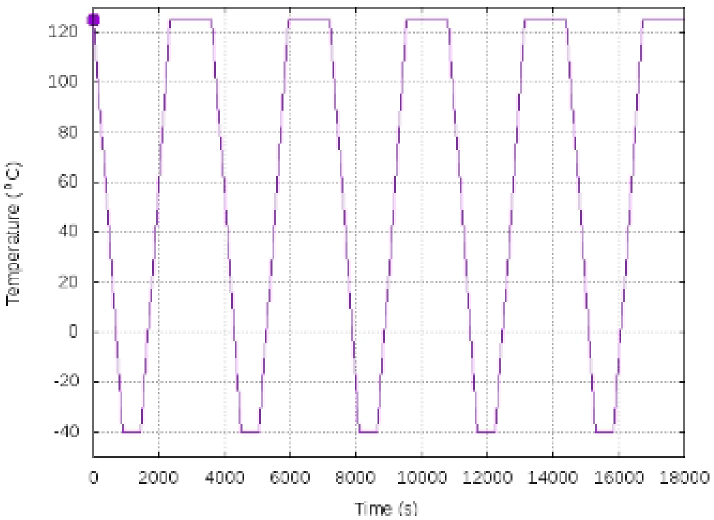
Verification

Comparison of shear stress

High-fidelity

POD Galerkin

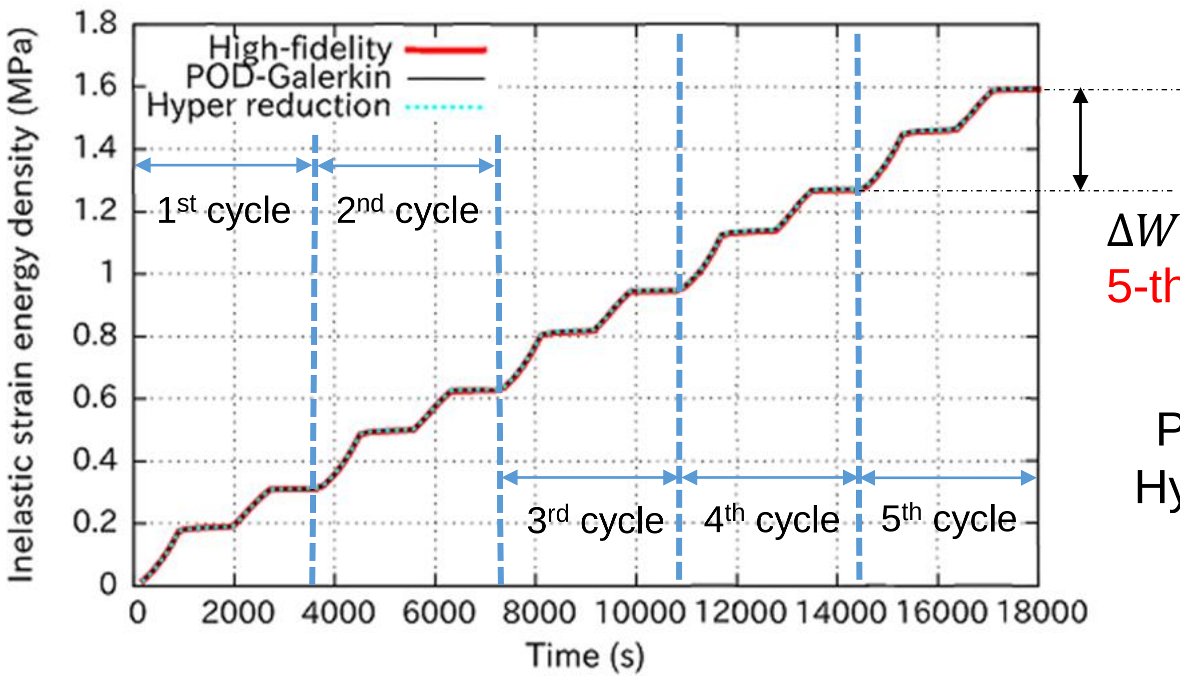
Hyper-reduction

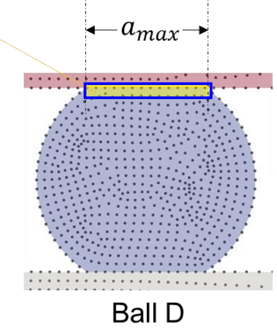


Lifetime estimation

Darveaux model (Darveaux, R., 2002, Basit, M et al., 2015): Lifetime N_f is estimated by the increment of inelastic strain energy density during one cycle at critical area.

Critical area Ω_{cr} : Crack will initiate here





ΔW during the 5-th cycle

High-fidelity : 0.3238 MPa
POD-Galerkin : 0.3236 MPa
Hyper-reduction : 0.3231 MPa

Based on ΔW during the 5-th cycle, lifetime N_f is estimated

Unit : cycle

High-fidelity	POD-Galerkin	Hyper-reduction
1,198	1,200	1,206

Computational Efficiency

In Newton-Raphson loop, the following 3 procedures are time consuming

(i) computing $\boldsymbol{J}$ and $\boldsymbol{Q}_{NL}$ (in Gappy-POD, $\mathbf{P}_{\mathrm{D}}{}^{\mathrm{T}}\boldsymbol{J}$ and $\mathbf{P}_{\mathrm{D}}{}^{\mathrm{T}}\boldsymbol{Q}_{NL}$),

(ii) matrix and vector operation

(iii) solving the algebraic equations

	CPU time per iteration Unit: s				Unit: s
	(i)	(ii)	(iii)	Total iteration number	Total CPU time
High-fidelity	0.16	0.0016	0.12	489	134.67
POD-Galerkin	0.17	0.18	0.00059	480	167.02
Hyper reduction	0.0039	0.0027	0.00037	486	3.53

Due to $\mathbf{P}_{\mathrm{D}}$

$\mathbf{V}_k{}^{\mathrm{T}}\boldsymbol{J}^{(i)}\mathbf{V}_k$

The number of equations
6 << 14,838 (much less than the original total DOF)

Summary

Physics-preserving **Model Order Reduction** (MOR) for fracture mechanics and nonlinear materials

- Integrated Singular Basis Function Method (ISBFM) is used to allow low order domain integration and yield a sparser discrete system
- A Decomposed Subspace Reduction (DSR) method is developed to preserve the near-tip singularity and discontinuities in the low-dimensional reduced model for the cracked region
- A robust reduced-order model for parameterized nonlinear systems characterized by a wide variety of nonlinear behaviors in terms of parameter changes.
- Manifold learning for a given data that lie in a non-Euclidean space finds an embedding into Euclidean space with maximal geometry preservation.
- A linear graph projection (LGP) based on the Graph-embedding framework that is a general framework to construct reduced order basis.
- By using the LGP with localized weighted relationship between pair-wise data points leads to a robust reduced order model (insensitive to outliers); whereas the POD is easily mislead by certain faraway data and ignoring the whole data structure.
- Reduced-order modeling (ROM) with Hyper Reduction techniques is effective to enhance efficiency in nonlinear materials modeling.