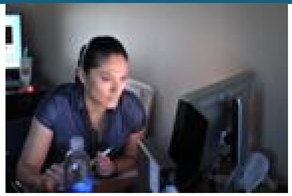
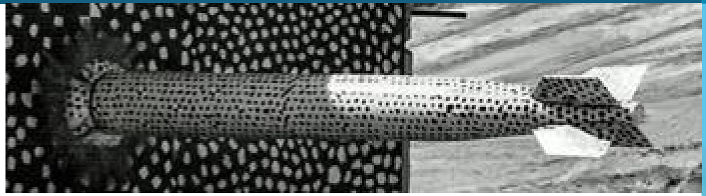
SAND2022-8261C

Sandia National Laboratories

Reduced order modeling with boosting Barlow Twins self-supervised learning for contact problem in a compressible hyperelastic material





T. Kadeethum and H. Yoon

Geomechanics Department, Sandia National Laboratories

Albuquerque, NM, USA

Collaborators: Ida Ang ,Jan Niklas Fuhg ,Nikolaos Bouklas (Cornell)

19th U.S. National Congress on Theoretical and Applied Mechanics

This work was supported by the Laboratory Directed Research and Development program at Sandia National Laboratories and also DOE Office of Fossil Energy project -Science-informed Machine Learning to Accelerate Real

Why reduced order model?

Full order model (FOM) is computationally demanding.

This would take 1-2 hours[1,2].

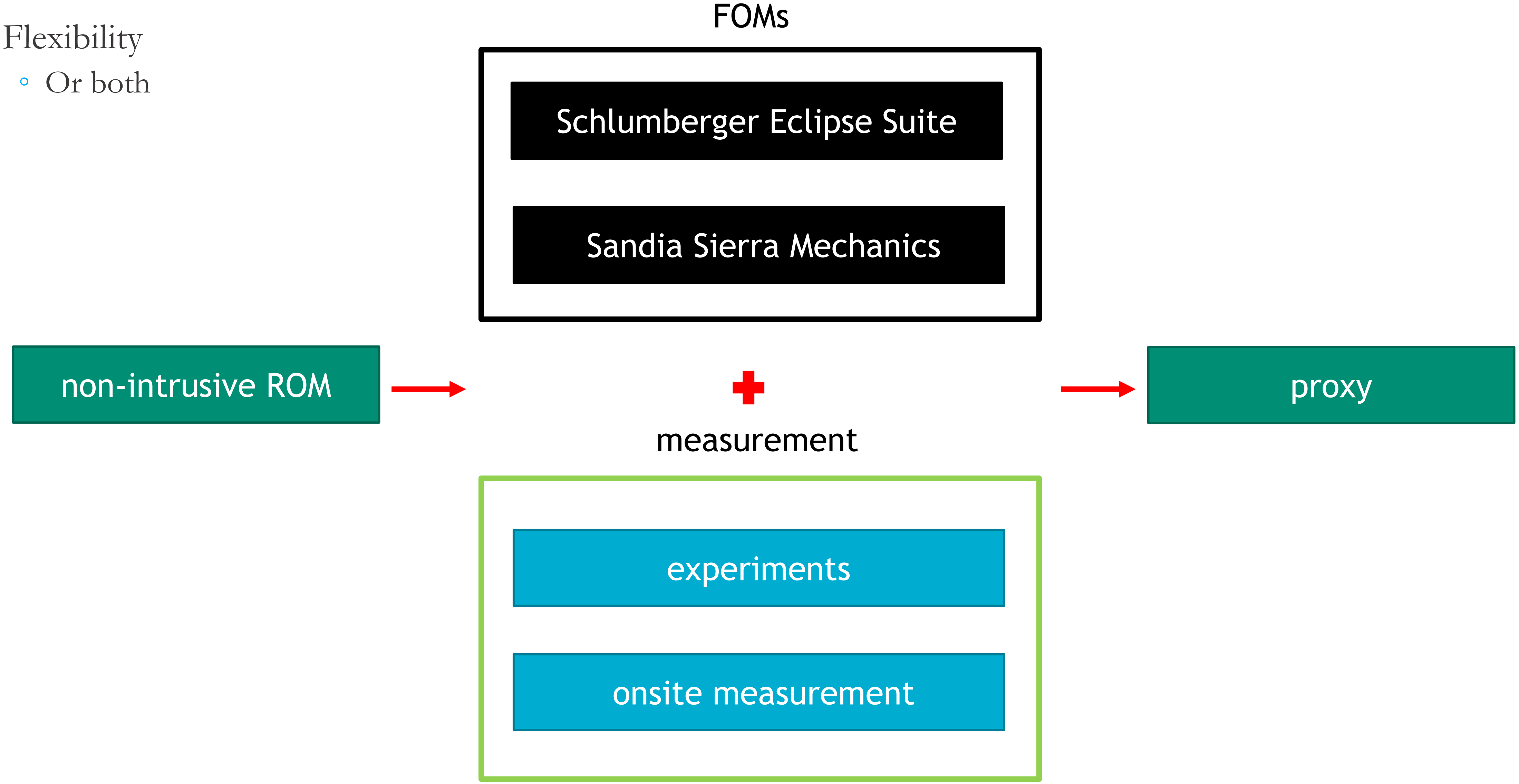
Imagine if you do 100,000 times of this type of simulation.

FOM is computationally very expensive for high fidelity simulations, uncertainty quantification, optimization, or inverse modeling

[1]Kadeethum et al. (2022, Advances in Water Resources)
[2]Kadeethum et al. (2021, Computers & Geosciences)

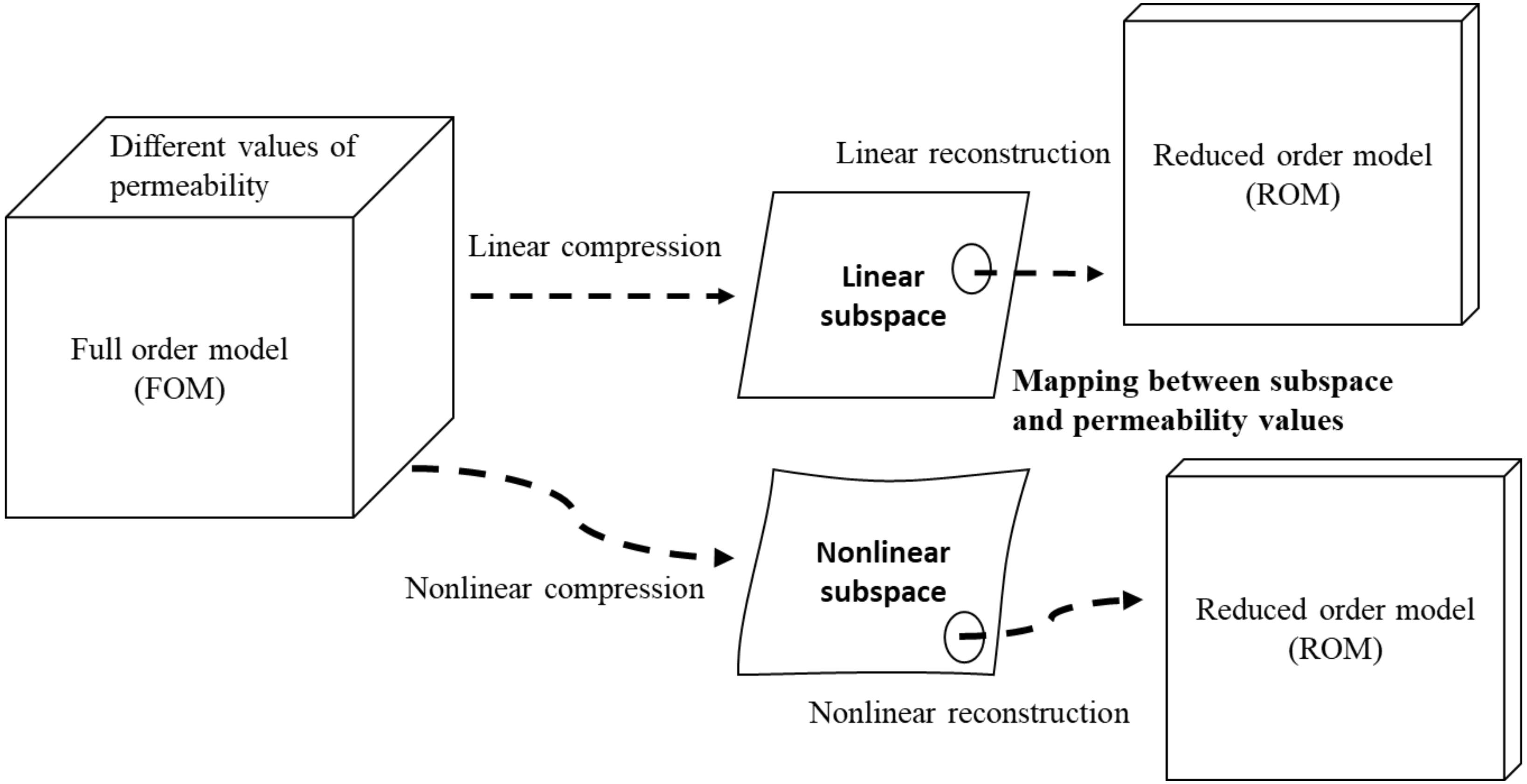
Why non-intrusive approach?

Flexibility
- Or both



Motivation

ROM typically works on ‘parameterized PDEs’ and ‘reduced subspace’

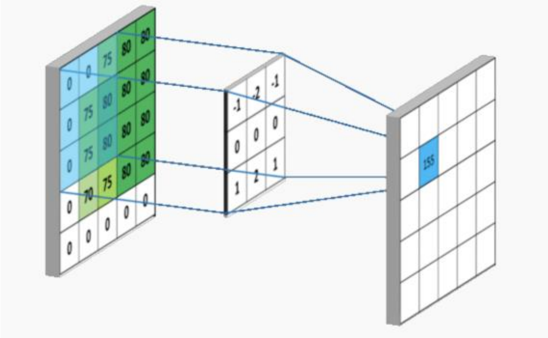


Motivation - continued

1. A unified framework suitable for problems that lie within both **linear** and **nonlinear** manifolds

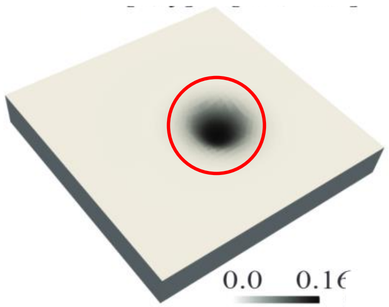
(proper orthogonal decomposition (POD) yields optimal data compression for linear manifolds) [1]

2. A framework that does not rely on 'convolutional layers,' which makes our framework applicable to both **structured** and **unstructured** meshes [1, 2]



[3]

3. Applying machine learning techniques for the physics-based problems with point source (or Dirac delta distribution) such as contact problems or subsurface flow with wells ➔ how to deal with imbalanced training data?



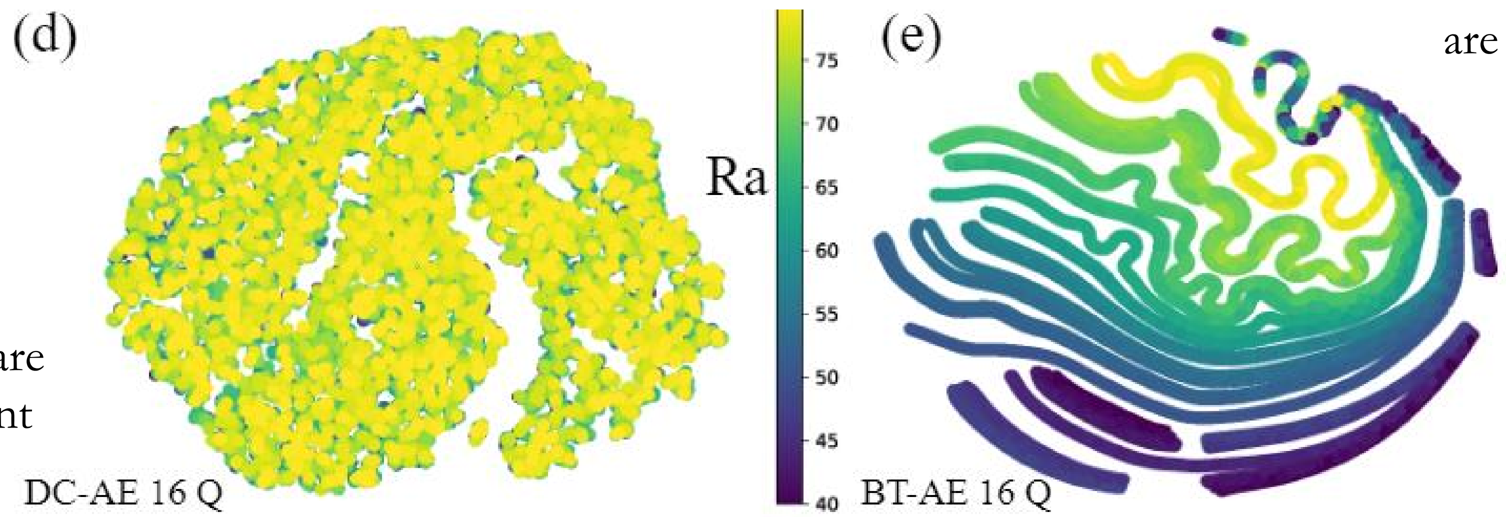
[1]Kadeethum et al. (2022, Advances in Water Resources)
[2]Kadeethum et al. (2021, Nature Computational Science)
[3]https://towardsdatascience.com/simple-introduction-to-convolutional-neural-networks-cdf8d3077bac

Idea

1.A key to develop a good ROM is to produce **better reduced manifolds** [1].

2. We apply Barlow Twins (BT) self-supervised learning [1,2], where BT maximizes the information content of the embedding with the latent space through a **joint embedding architecture**

The nonlinear manifolds are well structure



The nonlinear manifolds are not well structured in latent space

[1]Kadeethum et al. (2022, Scientific Report , accepted)
[2]Zbontar et al. (2021, arXiv:2103.03230)

Idea - continued

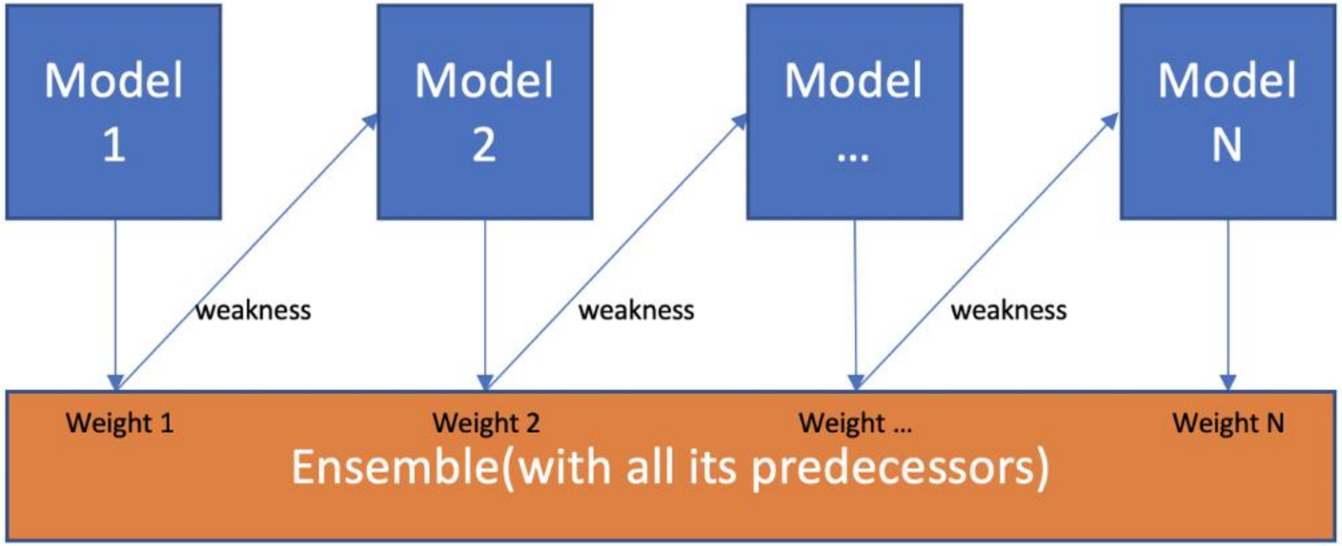
3. We apply a boosting concept for our previous BT-ROM [1]

4. Each model (in general sense) is trained **sequentially** using **subsample** from the training set with **weights**

6. The weights are calculated based on the current model's performance (i.e., **more error more weights**)

5. This way, the $\mathbf{model}^{n+1}$ is forced to learn the samples that $\mathbf{model}^{n}$ fails to mimic

[2]



[1] Kadeethum et al. (2022, Scientific Report, accepted)
[2] https://towardsdatascience.com/boosting-algorithms-explained-d38f56ef3f30

Methodology

1. Initialization

Training set: $\boldsymbol{\mu} = \left[\boldsymbol{\mu}^{(1)}, \boldsymbol{\mu}^{(2)}, \cdots, \boldsymbol{\mu}^{(M-1)}, \boldsymbol{\mu}^{(M)}\right]$

Validation set: $\boldsymbol{\mu}_{\text{validation}}$ = randomly select 5% of MN^t

Testing set: $\boldsymbol{\mu}_{\text{test}} = \left[\boldsymbol{\mu}^{(1)}_{\text{test}}, \boldsymbol{\mu}^{(2)}_{\text{test}}, \cdots, \boldsymbol{\mu}^{(M_{\text{test}}-1)}_{\text{test}}, \boldsymbol{\mu}^{(M_{\text{test}})}_{\text{test}}\right]$

We first initialize training, validation, and testing sets.

These parameters could be material properties, boundary conditions, or parameterized geometry representation.

Methodology

1. Initialization

Training set: $\boldsymbol{\mu} = [\boldsymbol{\mu}^{(1)}, \boldsymbol{\mu}^{(2)}, \cdots, \boldsymbol{\mu}^{(M-1)}, \boldsymbol{\mu}^{(M)}]$

Validation set: $\boldsymbol{\mu}_{\text{validation}} =$ randomly select 5% of MN^t

Testing set: $\boldsymbol{\mu}_{\text{test}} = [\boldsymbol{\mu}_{\text{test}}^{(1)}, \boldsymbol{\mu}_{\text{test}}^{(2)}, \cdots, \boldsymbol{\mu}_{\text{test}}^{(M_{\text{test}}-1)}, \boldsymbol{\mu}_{\text{test}}^{(M_{\text{test}})}]$

2. Full order model (FOM)

FOM = $\boldsymbol{X}_h(\boldsymbol{\mu}^{(1)}), \ldots, \boldsymbol{X}_h(\boldsymbol{\mu}^{(M)})$

Same goes for $\boldsymbol{\mu}_{\text{v}}, \boldsymbol{\mu}_{\text{t}}$

We then build the training set through by querying full order model for each parameter.

*This is the major cost of building data-driven model.

Methodology

Data compression: training BBT-AE model

The machine learning model has one encoder, decoder, and projector.

The main goal is to maximizes the information content of the embedding with the latent space through a joint embedding architecture.

Resulting in a **better reduced manifolds**

<u>If we have 1 encoder, our model is BT-ROM</u>
<u>If we have more than 1 encoders, our model is BBT-ROM</u>

<u>1. Initialization</u>

Training set: $\boldsymbol{\mu} = [\boldsymbol{\mu}^{(1)}, \boldsymbol{\mu}^{(2)}, \cdots, \boldsymbol{\mu}^{(M-1)}, \boldsymbol{\mu}^{(M)}]$

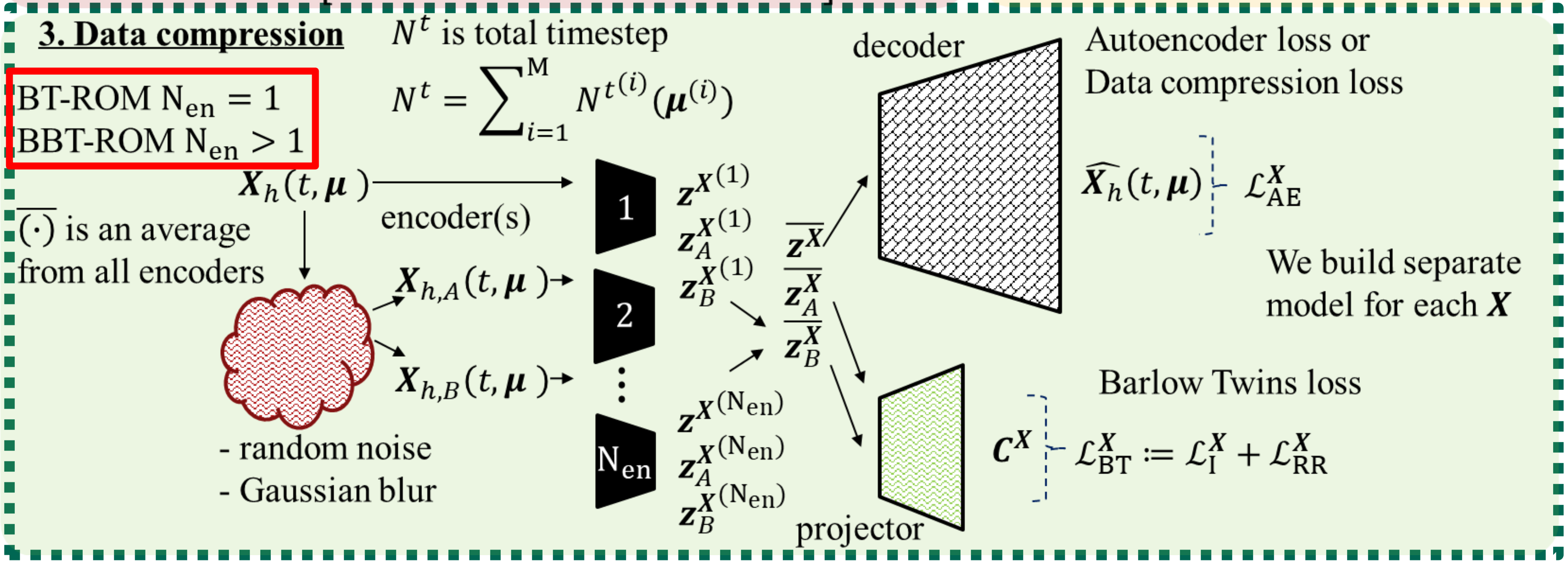
Validation set: $\boldsymbol{\mu}_{\text{validation}}$ = randomly select 5% of MN^t

Testing set: $\boldsymbol{\mu}_{\text{test}} = [\boldsymbol{\mu}^{(1)}_{\text{test}}, \boldsymbol{\mu}^{(2)}_{\text{test}}, \cdots, \boldsymbol{\mu}^{(M_{\text{test}}-1)}_{\text{test}}, \boldsymbol{\mu}^{(M_{\text{test}})}_{\text{test}}]$

<u>2. Full order model (FOM)</u>

FOM = $\boldsymbol{X}_h(\boldsymbol{\mu}^{(1)}), \ldots, \boldsymbol{X}_h(\boldsymbol{\mu}^{(M)})$

Same goes for $\boldsymbol{\mu}_{\text{v}}, \boldsymbol{\mu}_{\text{t}}$



Methodology

We then map our parameters to reduced manifolds using ANN.

*We note that we could use other regressors such as GP or RBF.

1. Initialization

Training set: $\boldsymbol{\mu} = [\boldsymbol{\mu}^{(1)}, \boldsymbol{\mu}^{(2)}, \cdots, \boldsymbol{\mu}^{(M-1)}, \boldsymbol{\mu}^{(M)}]$

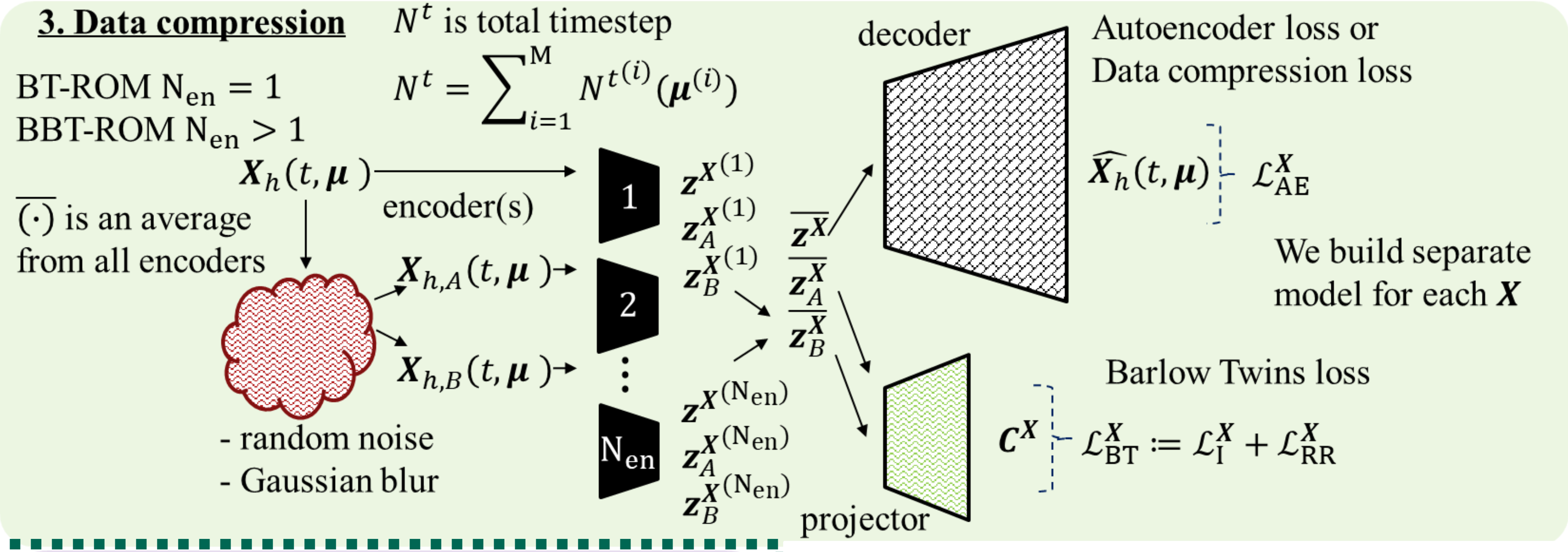
Validation set: $\boldsymbol{\mu}_{\text{validation}}$ = randomly select 5% of MN^t

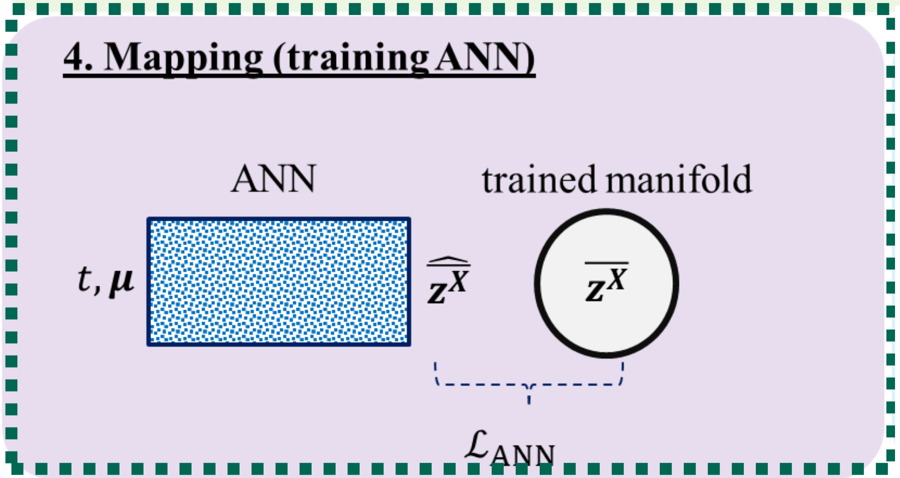
Testing set: $\boldsymbol{\mu}_{\text{test}} = [\boldsymbol{\mu}_{\text{test}}^{(1)}, \boldsymbol{\mu}_{\text{test}}^{(2)}, \cdots, \boldsymbol{\mu}_{\text{test}}^{(M_{\text{test}}-1)}, \boldsymbol{\mu}_{\text{test}}^{(M_{\text{test}})}]$

2. Full order model (FOM)

FOM = $\boldsymbol{X}_h(\boldsymbol{\mu}^{(1)}), \ldots, \boldsymbol{X}_h(\boldsymbol{\mu}^{(M)})$

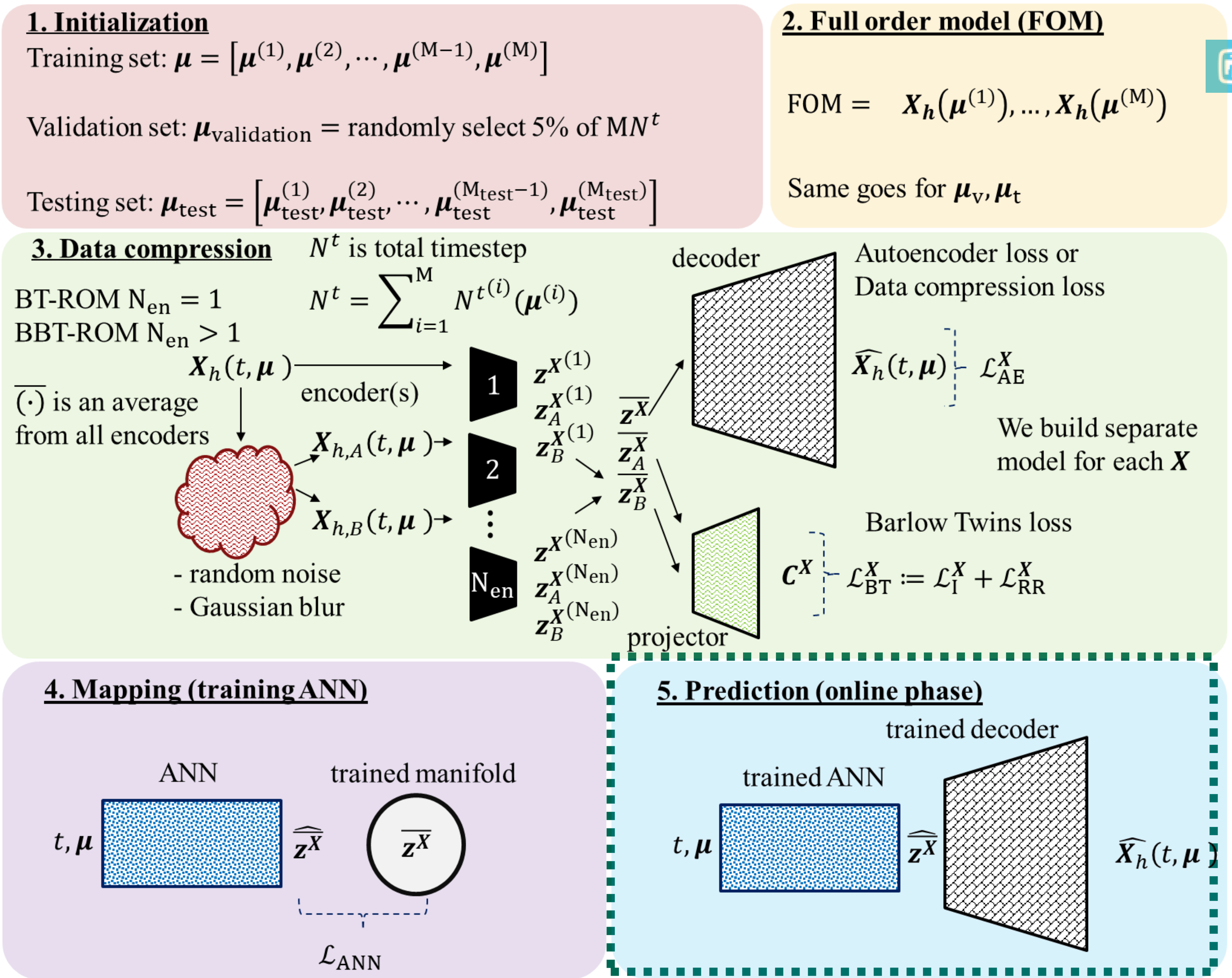
Same goes for $\boldsymbol{\mu}_{\text{v}}, \boldsymbol{\mu}_{\text{t}}$





Methodology

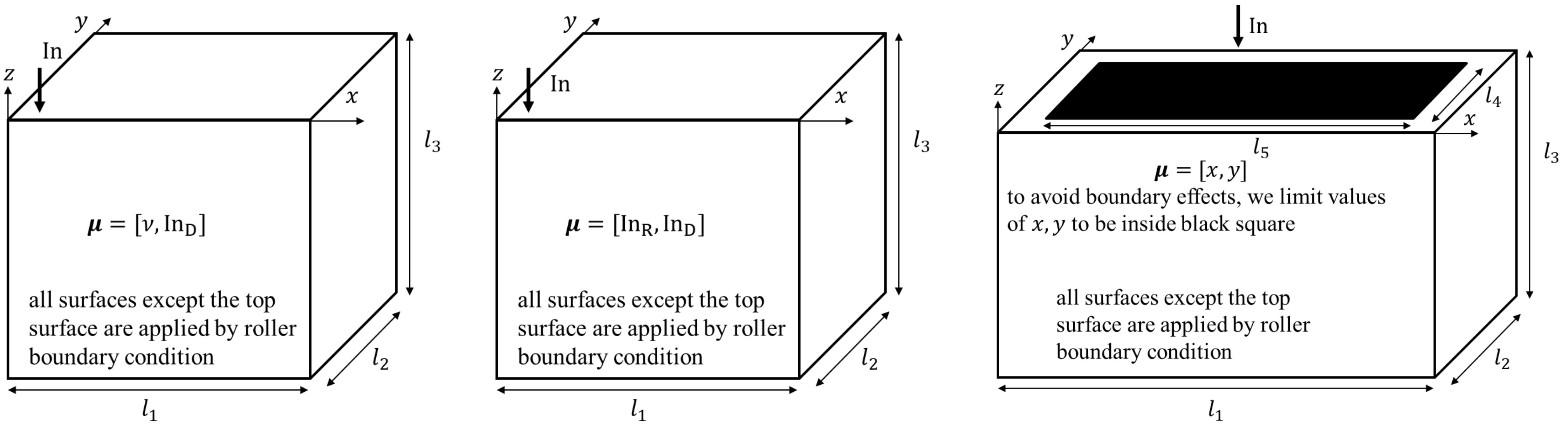
During the online or prediction phase, we approximate our quantities of interest through the **trained ANN** and **trained decoder**.



Physical problems that we test

- **Contact** between a **rigid** indenter and a **hyperelastic** substrate at **finite deformations**
- Goal: To improve ML training with **imbalanced** training data (i.e., only one point of contact)
- A **small area where the deformation occurs** while **most of the domain remains are undeformed**.
- Weak form of contact physics

$$\mathrm{k_{pen}} \int_{\partial\Omega_c} \langle -g_N \rangle \delta u_N \, dS + \int_{\Omega_0} \mathbf{P} : \nabla(\boldsymbol{\delta}\mathbf{u}) \, dV - \int_{\Omega_0} \mathbf{B} \cdot \boldsymbol{\delta}\mathbf{u} \, dV - \int_{\partial\Omega_N} \mathbf{T} \cdot \boldsymbol{\delta}\mathbf{u} \, dS = 0$$



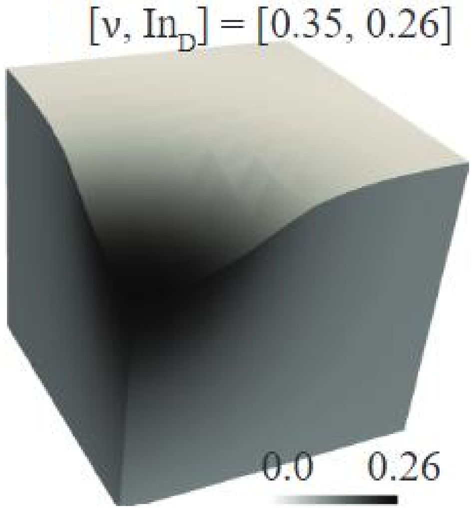
Results – Poisson's ratio and indentation depth as parameters

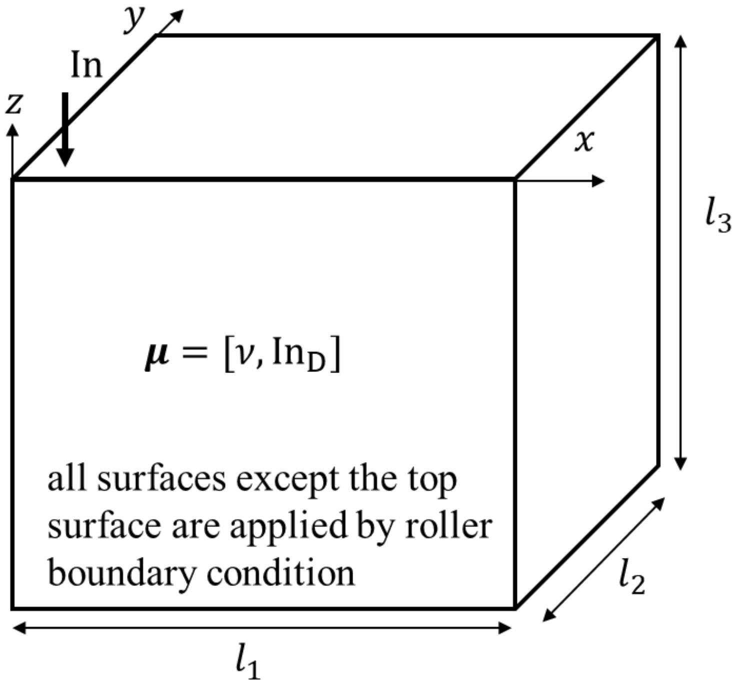
1. It is symmetric; so, we only model a quarter of the full domain (the contact point is in the middle of the material)

Parameters
Poisson's ratio = $[0.1, 0.4]$
indentation depth = $[0.1, 0.3]$

Training: 1600
Validation: 80 (5% of training set)
Testing: 100

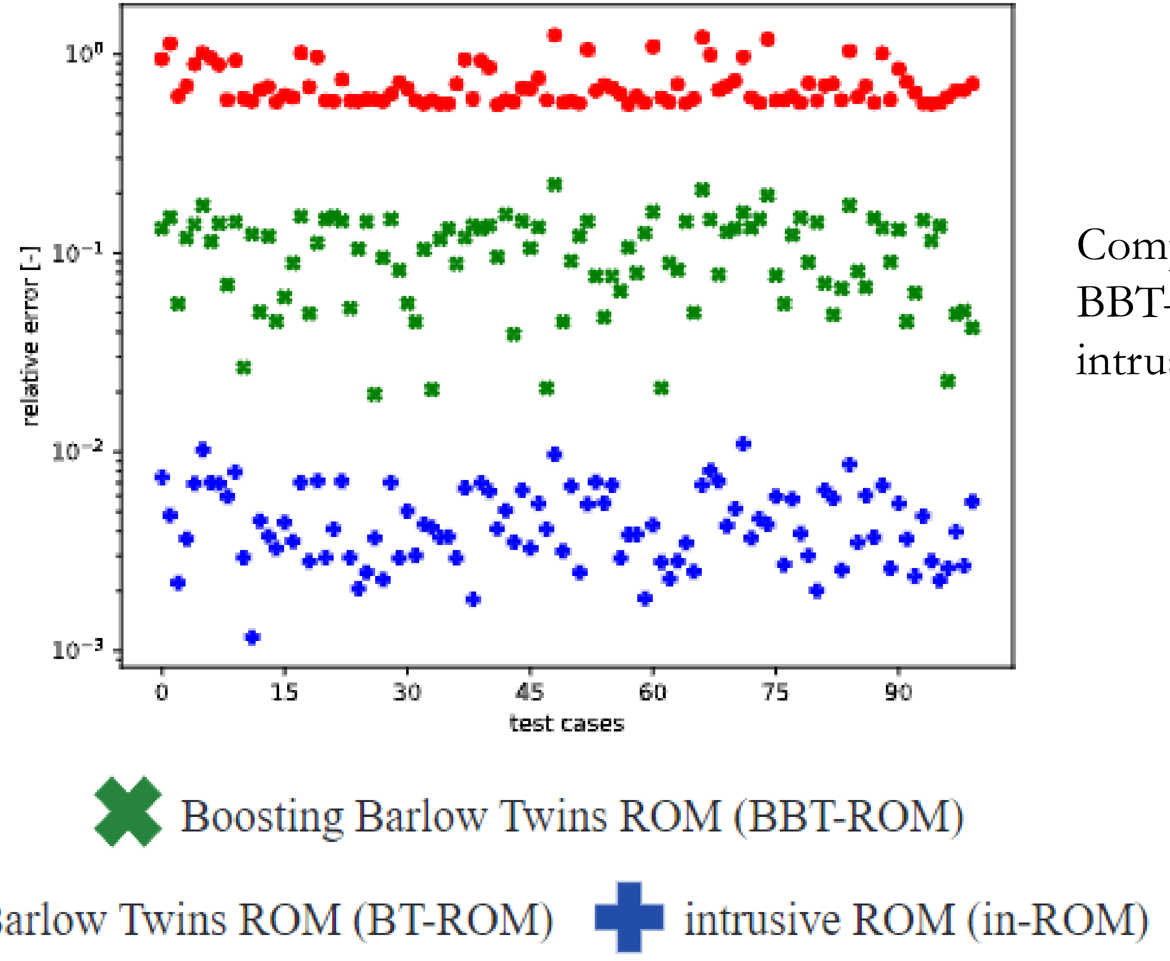
DOFs: 3993





Results – Poisson's ratio and indentation depth as parameters

1. We show here relative error results (compared to full order model)
2. We observe that the proposed model (BBT-ROM) has a better accuracy than our previous model (BT-ROM), but worse than intrusive-ROM.



Computational time:
BBT-ROM = **0.001** s
intrusive-ROM = **8.0** s

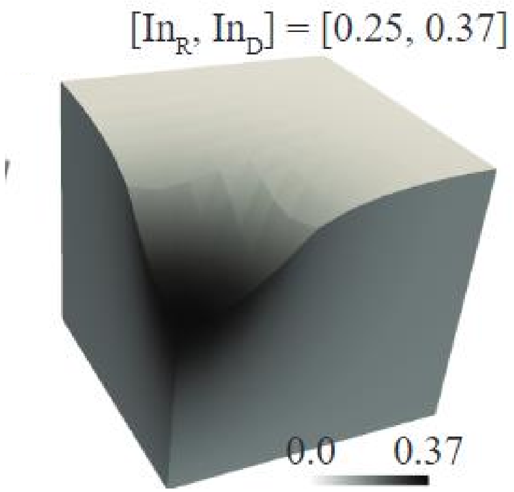
Results – indentation radius and depth as parameters

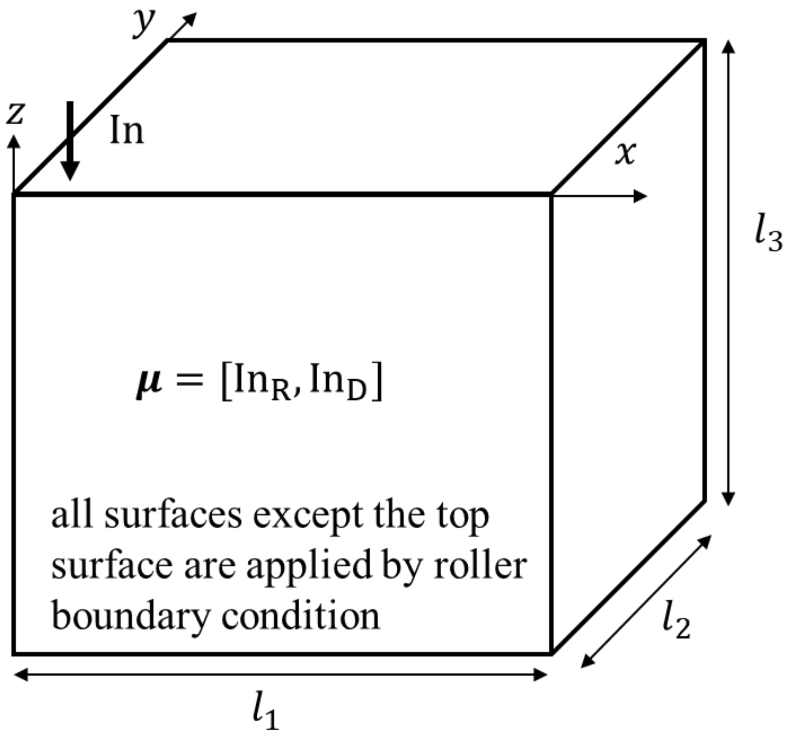
1. It is symmetric; so, we only model a quarter of the full domain (the contact point is in the middle of the material) (the same as in the first problem)

Parameters
indentation radius $= [0.15, 0.4]$
indentation depth $= [0.1, 0.4]$

Training: 1600
Validation: 80 (5% of training set)
Testing: 100

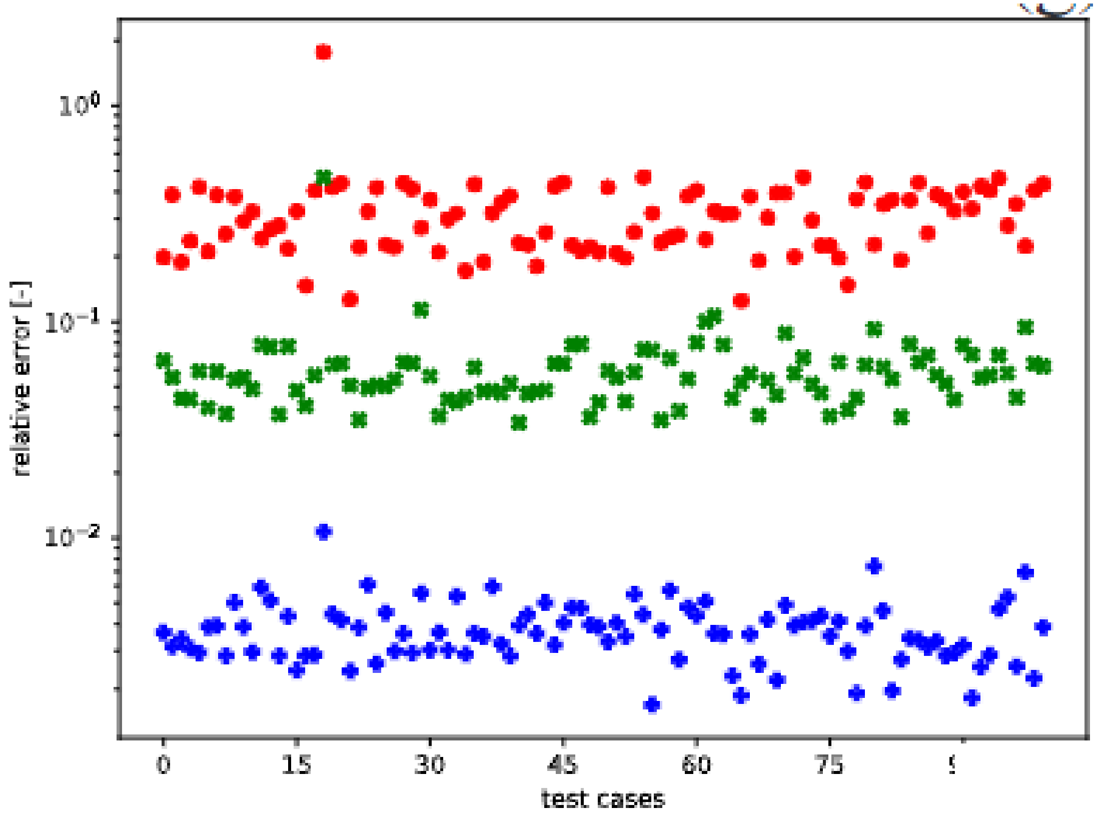
DOFs: 3993





Results – indentation radius and depth as parameters

1. We show here a relative error results (relative to full order model) – we observe that the proposed model (BBT-ROM) has a better accuracy than our previous model (BT-ROM), but worse than in-ROM.



Computational time:
BBT-ROM = **0.001** s
intrusive-ROM = **8.0** s

Boosting Barlow Twins ROM (BBT-ROM)

Barlow Twins ROM (BT-ROM) intrusive ROM (in-ROM)

Results – indentation locations as parameters

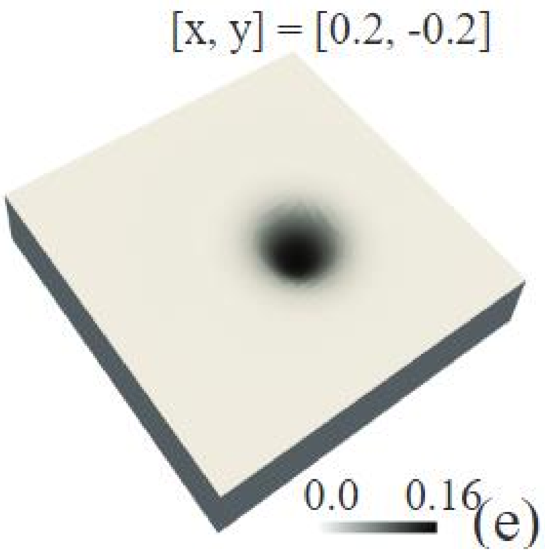
1. We model the whole domain and contact location could occur within the black square

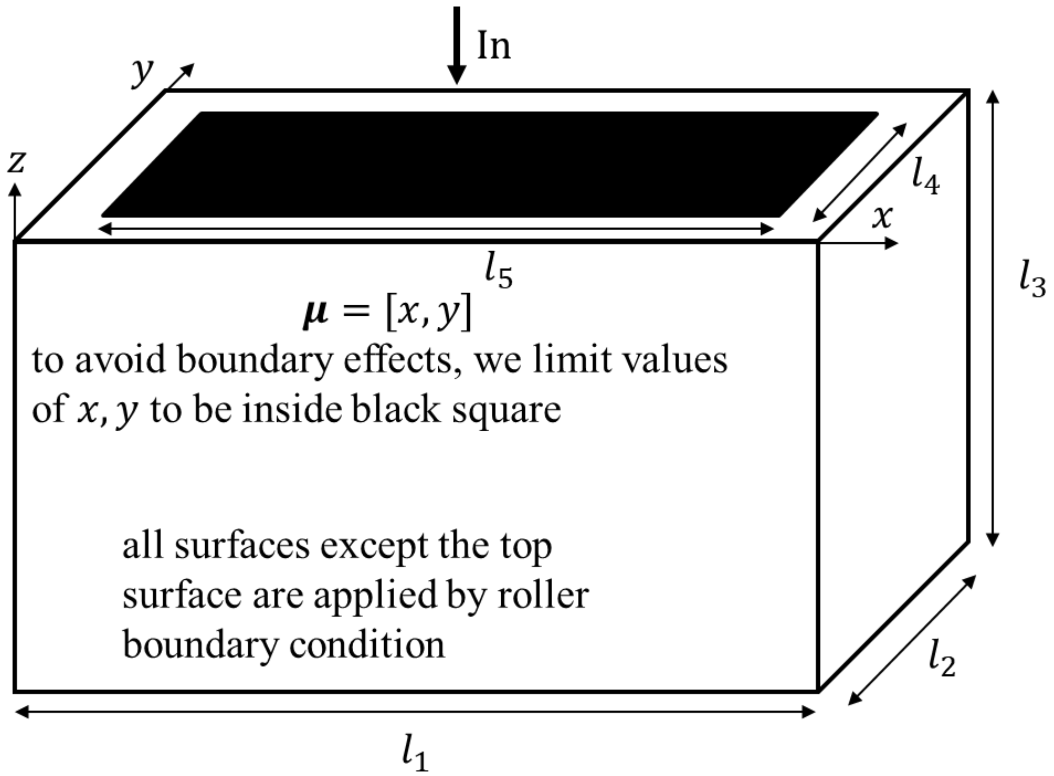
2. The indentation depth and radius are fixed

Parameters
x-coordinate = $[-0.3, 0.3]$
y-coordinate = $[-0.3\ 0.3]$

Training: 1600
Validation: 80 (5% of training set)
Testing: 100

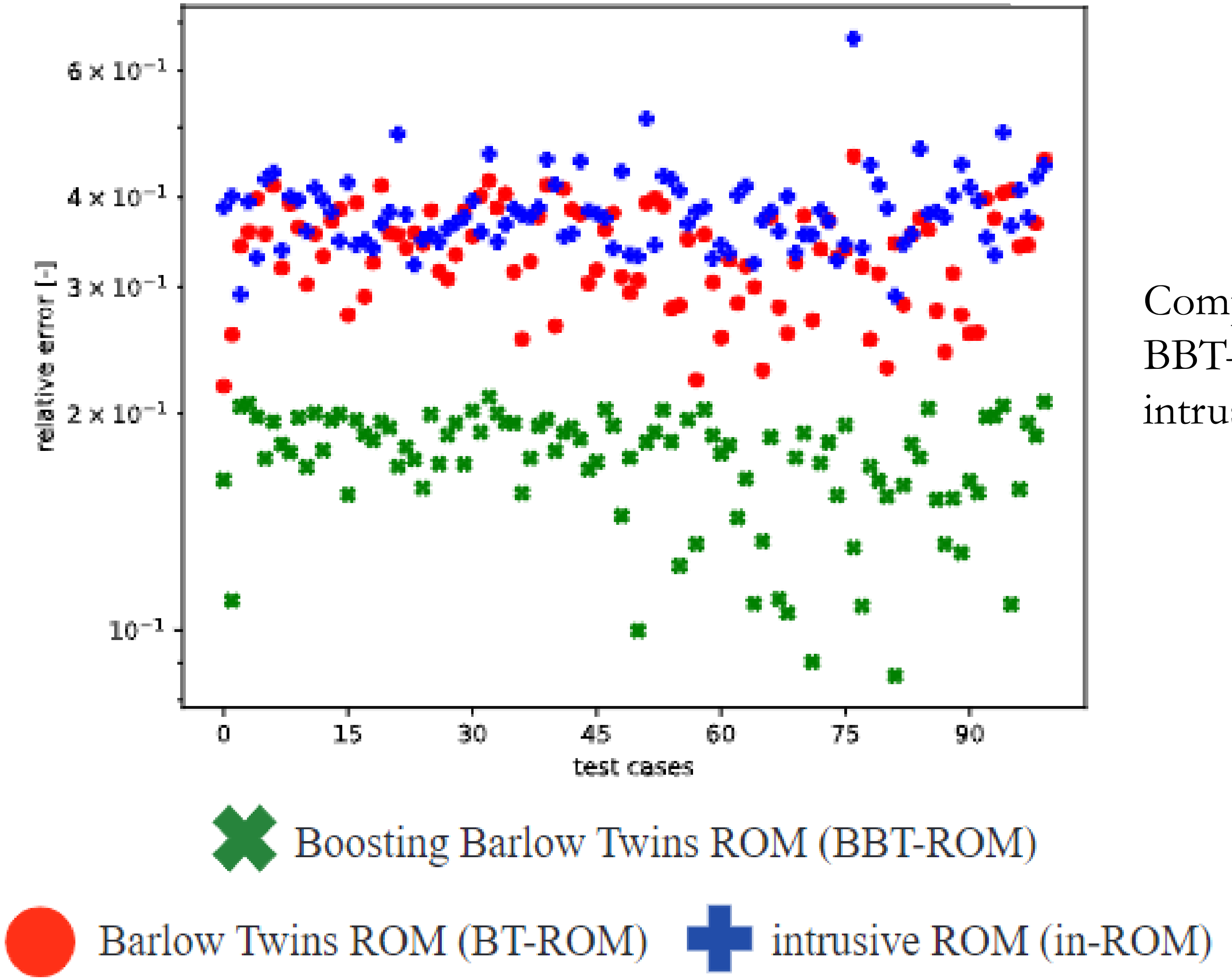
DOFs: 70602





Results – indentation locations as parameters

1. We show here a relative error results (relative to full order model) – we observe that the proposed model (BBT-ROM) has a **best** accuracy than our previous model (BT-ROM) and in-ROM.



Computational time:
BBT-ROM = **0.001** s
intrusive-ROM = ~**20.0** s

Conclusions

1. A ROM framework that works in an optimal way for both **linear** and **nonlinear** manifolds

2. A ROM framework that can be applied for both **structured** and **unstructured** meshes

3. A ROM framework that can handle **data imbalanced** problems

4. An uncertainty-aware BT-ROM is in progress to achieve uncertainty quantification (Neural IPS 2022, in review)

Physical problems that we test - continued

$$k_{pen}\int_{\partial\Omega_c}\langle -g_N\rangle \delta u_N\, dS + \int_{\Omega_0} \mathbf{P} : \nabla(\boldsymbol{\delta}\mathbf{u})\, dV - \int_{\Omega_0} \mathbf{B}\cdot\boldsymbol{\delta}\mathbf{u}\, dV - \int_{\partial\Omega_N} \mathbf{T}\cdot\boldsymbol{\delta}\mathbf{u}\, dS = 0$$

weak form

$$\begin{aligned} \nabla_X \cdot \mathbf{P} + \mathbf{B} &= 0 \quad \text{in } \Omega_0 \\ \mathbf{u} &= \bar{\mathbf{u}} \quad \text{on } \partial\Omega_D \\ \mathbf{P}\cdot\mathbf{N} &= \bar{\mathbf{T}} \quad \text{on } \partial\Omega_N \end{aligned}$$

$$\mathrm{In}(x,y) = -\mathrm{In}_D + \frac{1}{2\mathrm{In}_R}(x^2+y^2) \quad \text{about origin}$$

We approximate the contact profile with a parabolic function

We use PETSc SNES as a nonlinear solver and MUMPS as a linear solver with absolute and relative tolerances of 1×10^{-6} and 1×10^{-16}, respectively. We utilize a backtracking line search with slope descent parameter of 1×10^{-4}, initial step length of 1.0, and quadratic order of the approximation.

'Numerical Tours of Computational Mechanics with FEniCS