Manifold Learning and Autoencoders in Nonlinear Materials Modeling

J. S. Chen, Xiaolong He, Karan Taneja
University of California, San Diego

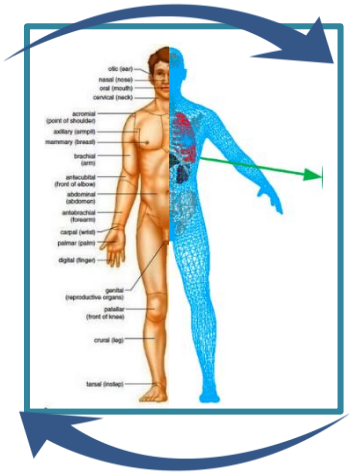
Qizhi He
San Diego State University

Chung-Hao Lee, Devin Laurence
The University of Oklahoma, Norman

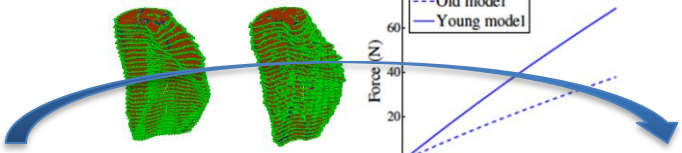
Digital-Twin and Innovations in Cyber Technologies

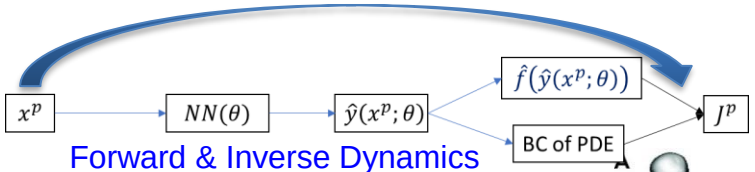
Digital Twin informing actions and interventions for the Human

1. Prehabilitation

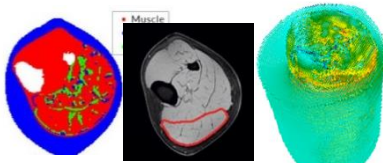


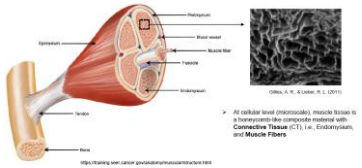
2. Rehabilitation

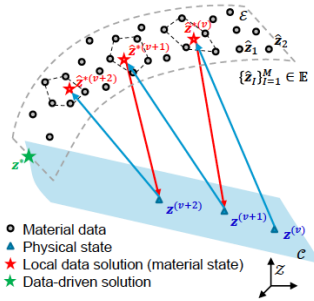


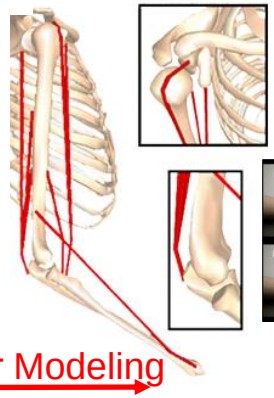


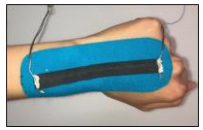
Forward & Inverse Dynamics

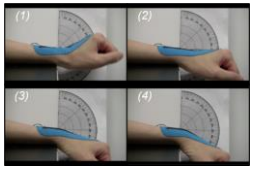










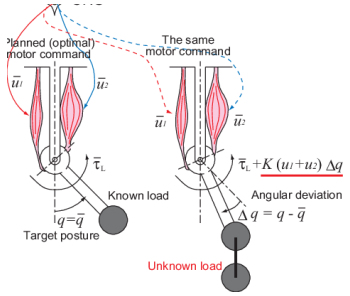


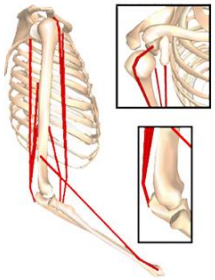
Manifold Learning & Reduced Order Modeling

Deep Learning & Parameters Identification

Component /Whole-Body Scale

Multi-body Dynamics Simulation

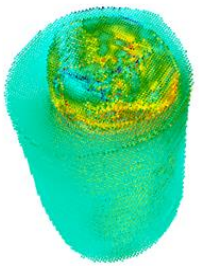




Reduced Order

Tissue (Continuum) Scale

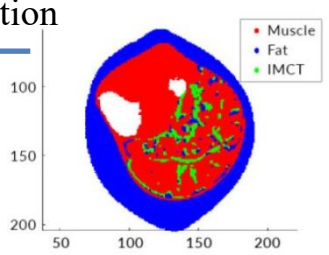
Continuum Mechanics Modeling

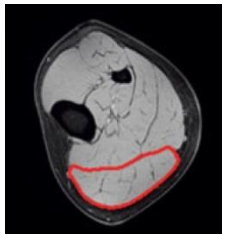


Homogenization

Cellular (Meso) Scale

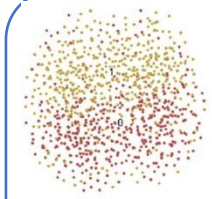
Meshfree Discretization





Data-Enhanced Physics-based Simulator

DATA-DRIVEN ALGORITHMS



High Dimensional data

Supervised

- Regression
- *k*-NN
- Random Forest
- Neural network
- ….

Unsupervised

- PCA (or NLDM)
- Clustering
- Outlier detection
- Structure discovery
- ....

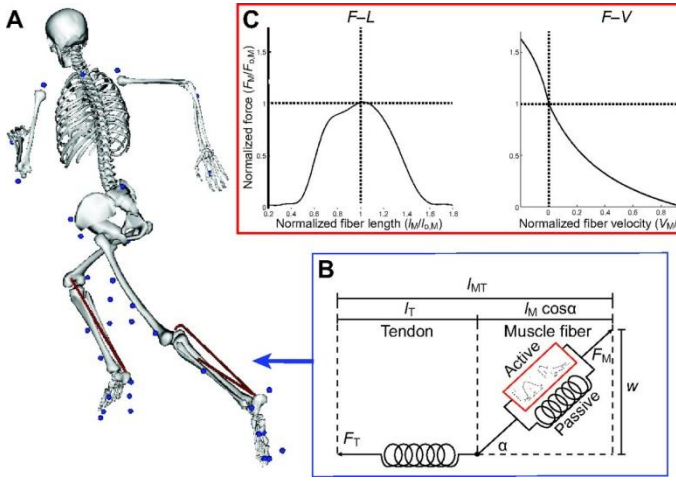
DATA WAREHOUSE

- Spatial/temporal sensing data
- MRI/DTI imaging data
- Laboratory testing data
- Medical history data
- ....

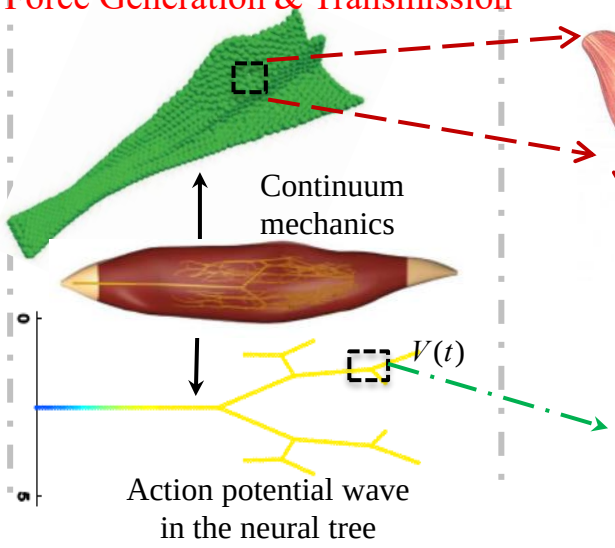
PHYSICAL MODELS

- Conservation laws
- Compatibility
- Thermodynamics
- Reaction-diffusion systems
- ....

Mobility & Performance

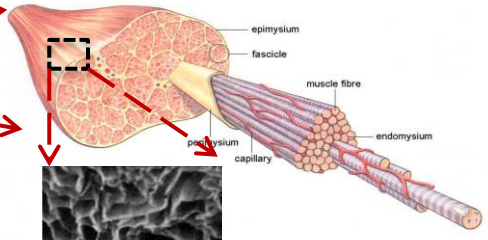


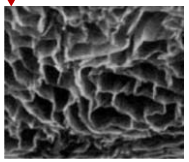
Force Generation & Transmission



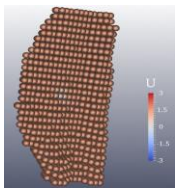
Action potential wave in the neural tree

Muscle Fiber Contraction





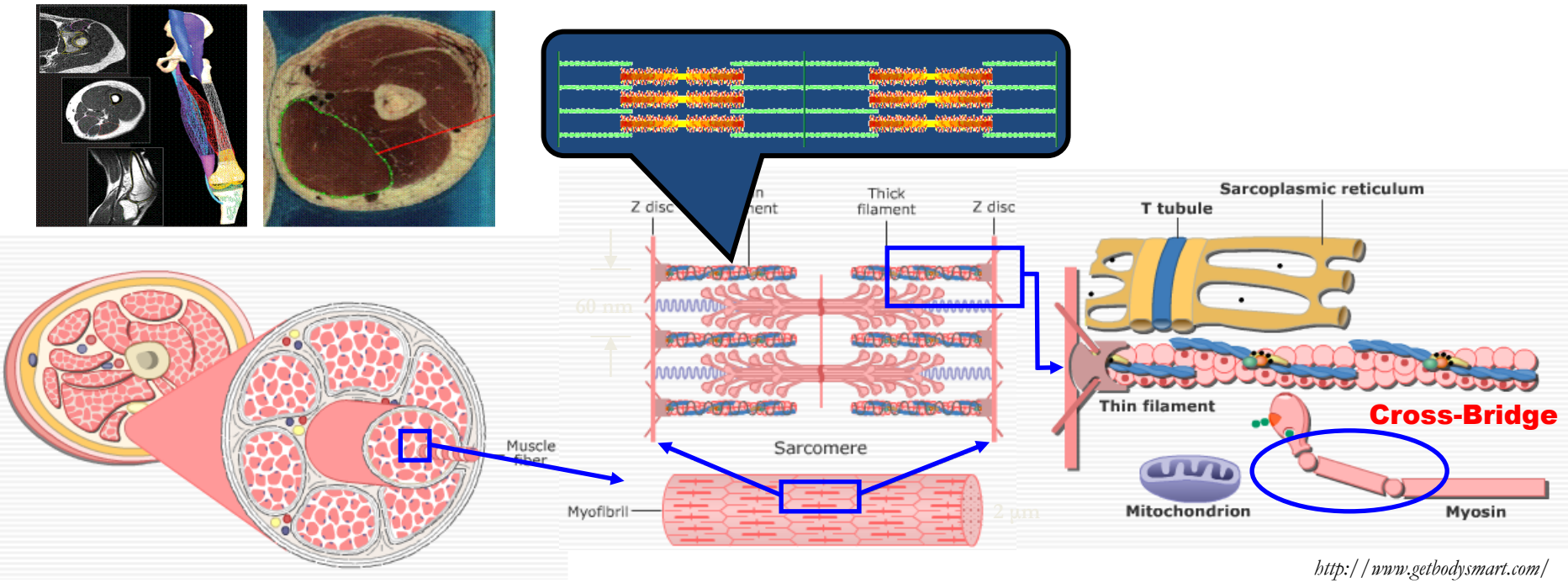
Extracellular matrix (ECM)



Action potential wave in muscle tissue

Reduced order, homogenization — *Structural decomposition* - - *Mechanical info* —· *Electrical info*

Challenges in Computational Biomechanics



- Geometry complexity
- Model construction from images

⇒ Image Based RKPM with Nodal Integration

- Heterogeneity
- Multiple length scales

- Complexities in material models

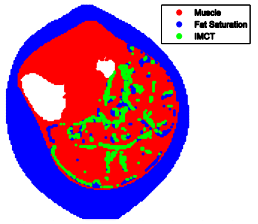
⇒ Data-Driven Computing Enhanced with Machine Learning Approach

- Multiple physics/chemistry

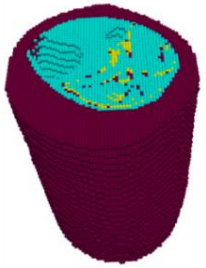
Model Reduction and Data-Driven Modeling for Computational Biomechanics

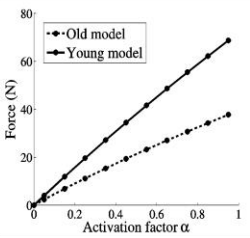
—*Physics info* - - *Data info* —· *Empirical info*

MULTI-SCALE IMAGE DATA



NUMERICAL MODELING





(RKPM)

MATERIAL MODELS

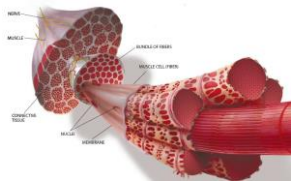
Phenomenological

Multiscale

Cellular, subcellular…

Patient-specific

Randomness



PHYSICAL MODELS

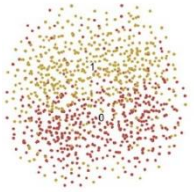
Equilibrium: e.g. $\nabla \cdot \boldsymbol{\sigma} + \boldsymbol{b} = \boldsymbol{0}$

Compatibility: e.g. $\boldsymbol{\varepsilon} = \frac{1}{2}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T)$

… other physical laws (constraints)

- Surrogate modeling
- Data-driven modeling
- …

DATA-DRIVEN MODELS



High Dimensional data

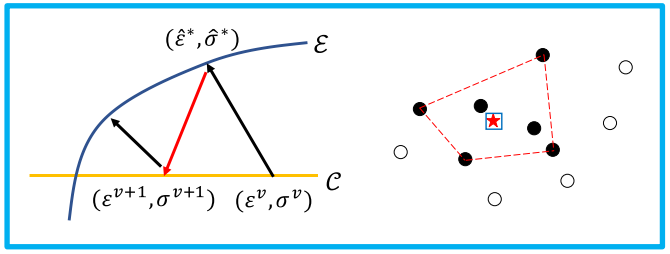
Supervised $x \longrightarrow y$

- Regression
- *k*-NN
- Random Forest
- Neural network
- ….

Unsupervised x

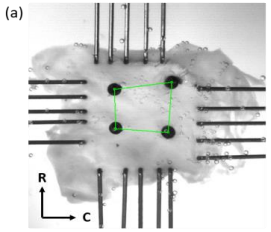
- PCA (or NLDM)
- Clustering
- Outlier detection
- Structure discovery
- ….

Data-enhanced physics-based modeling paradigm

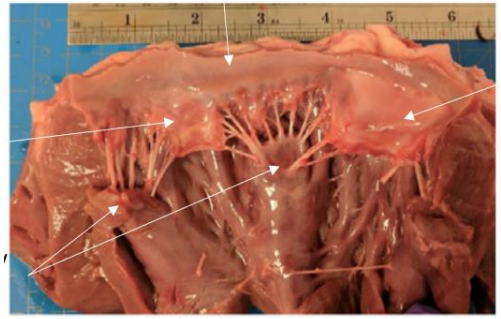


Procedures of Data-driven solver

Experimental setup

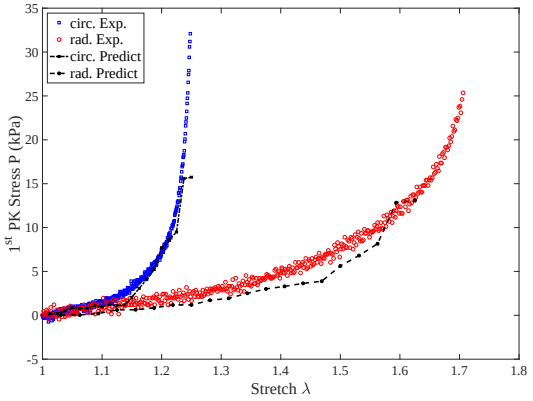


Mitral Annulus



MVAL

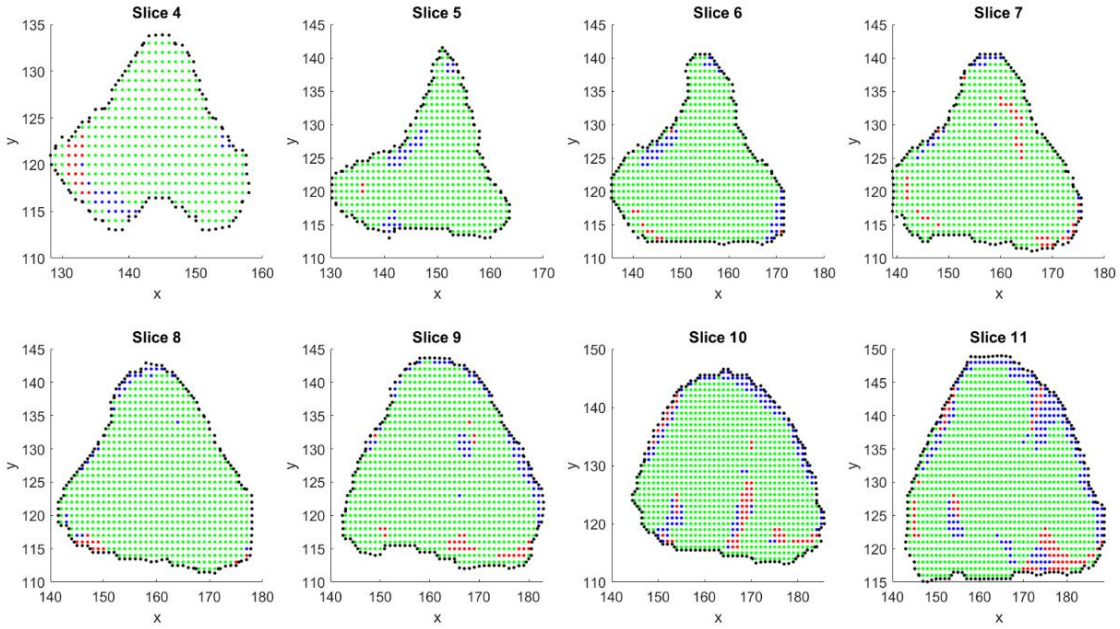
Experiment test for ***heart valve tissue***



Data-driven prediction

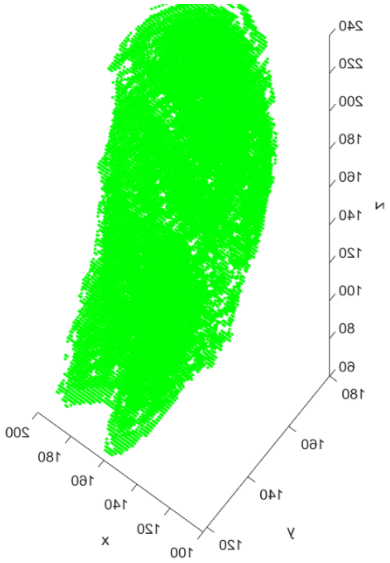
Cellular Scale Models

MRI

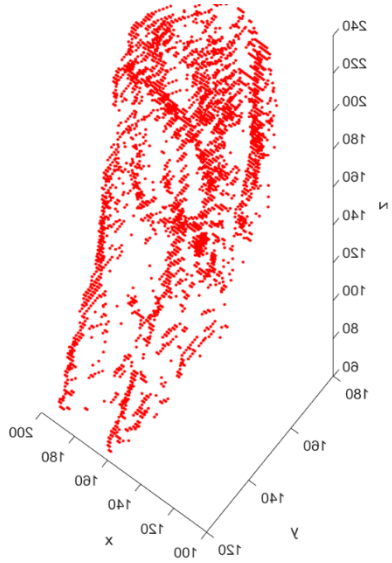


DTI

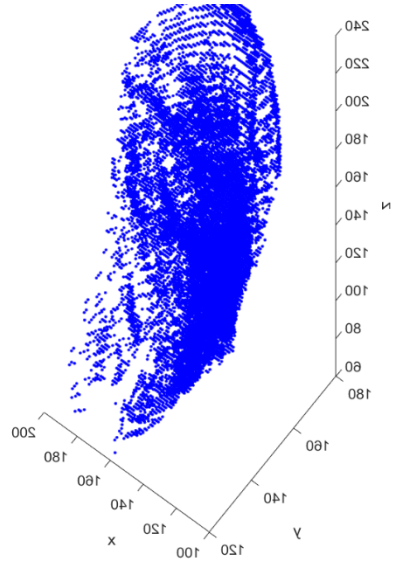
Muscle



Fat



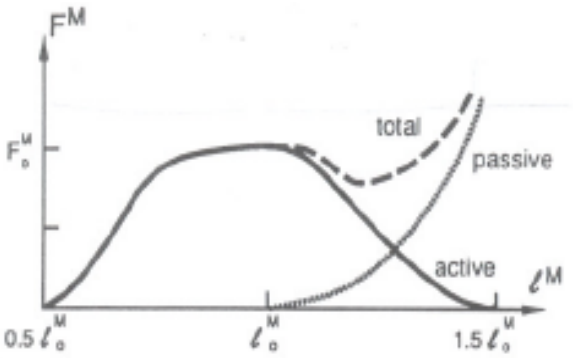
Connective Tissue

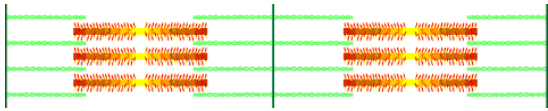


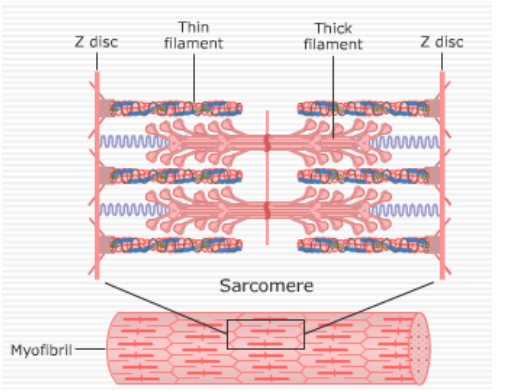
Stress in Muscle Fibers

Passive Stress + Active Stress

$$\sigma_{fiber}^{total} = \sigma_{fiber}^{act} + \sigma_{fiber}^{pass}$$



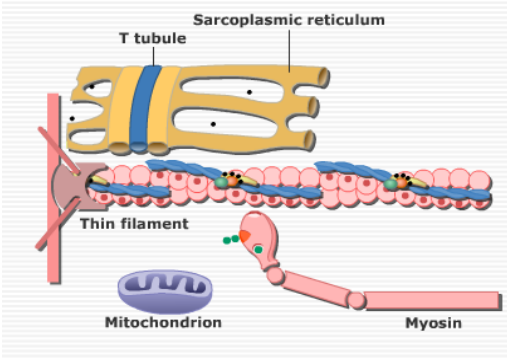


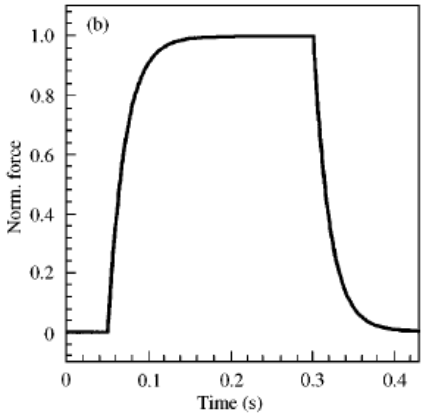


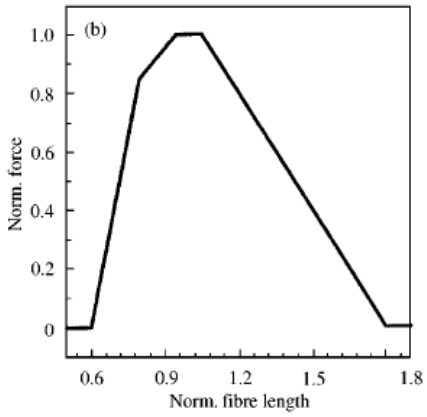
Active Stress -Hill 1939, Zajac 1989, van Leeuwen 1992

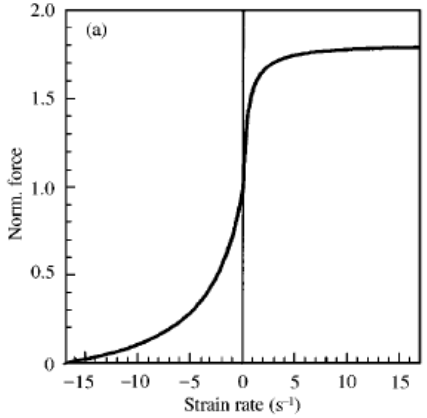
$$\sigma_{fiber}^{act} = \sigma_{iso} f_a f_l f_v$$

where σ_{iso} : maximum isometric stress









f_a : activation function

f_l : length dependent function

$f_{\dot{v}}$ velocity dependent function

S. W. Chi, J. Hodgson, J. S. Chen, et al., Journal of Biomechanics, 2010

Data-Driven Computational Framework

Physical Laws

Equilibrium

$$\nabla \cdot \boldsymbol{\sigma} + \boldsymbol{b} = \boldsymbol{0}, \text{ in } \Omega$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \bar{\boldsymbol{t}}, \text{ on } \Gamma_t$$

Compatibility

$$\boldsymbol{\varepsilon} = \frac{1}{2}\left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T\right) \text{ in } \Omega$$

$$\boldsymbol{u} = \bar{\boldsymbol{u}}, \text{ on } \Gamma_u$$

Physical Manifold Set $\mathcal{C}$

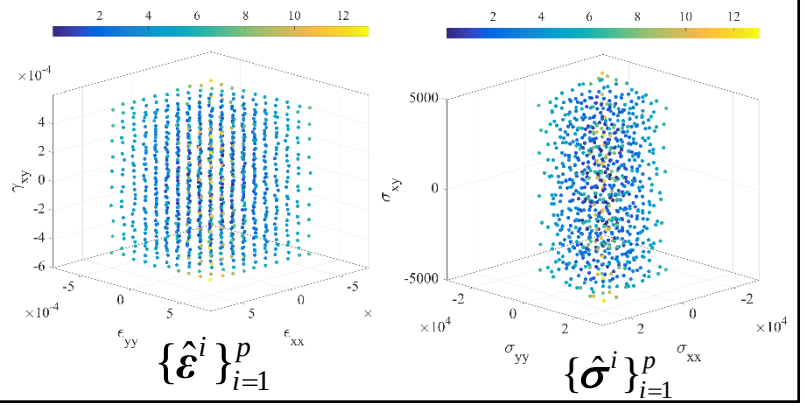
$$\mathcal{C} = \{(\boldsymbol{\varepsilon}[\boldsymbol{u}], \boldsymbol{\sigma}) \mid \boldsymbol{u} \in \mathcal{V}_u, \boldsymbol{\sigma} \in \mathcal{V}_\sigma\}$$

$$\mathcal{C}_\sigma = \{\boldsymbol{\tau} \in \mathcal{V}_\sigma \mid \nabla \cdot \boldsymbol{\tau} + \boldsymbol{b} = \boldsymbol{0} \text{ in } \Omega, \text{ and } \boldsymbol{\tau} \cdot \boldsymbol{n} = \bar{\boldsymbol{t}} \text{ on } \Gamma_t\}$$

$$\mathcal{C}_u = \{\boldsymbol{v} \in \mathcal{V}_u \mid \boldsymbol{v} = \bar{\boldsymbol{u}} \text{ on } \Gamma_u\}$$

Material Data Set $\mathbb{E}_{em}$

$$\mathbb{E}_{em} = \{(\hat{\boldsymbol{\varepsilon}}^i, \hat{\boldsymbol{\sigma}}^i), i = 1, \ldots, p\}$$



Data-Driven Solution: $\mathbb{E}_{em} \cap \mathcal{C}$

Data-Driven Problem *(Bonnet 2005; Avril et al. 2008; Kirchdoerfer & Ortiz 2016)*

$$\mathcal{H}(\boldsymbol{u}, \boldsymbol{\sigma}, \hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}}) = \int_\Omega \left(\frac{1}{2}(\boldsymbol{\varepsilon}(\boldsymbol{u}) - \hat{\boldsymbol{\varepsilon}}) : \boldsymbol{M}^\varepsilon : (\boldsymbol{\varepsilon}(\boldsymbol{u}) - \hat{\boldsymbol{\varepsilon}}) + \frac{1}{2}(\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}) : \boldsymbol{M}^\sigma : (\boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}})\right) d\Omega$$

$$\min_{(\hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}}) \in \mathbb{E}_{em}} \min_{(\boldsymbol{u}, \boldsymbol{\sigma}) \in \mathcal{C}_u \times \mathcal{C}_\sigma} \mathcal{H}(\boldsymbol{u}, \boldsymbol{\sigma}, \hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}}) = \min_{(\boldsymbol{u}, \boldsymbol{\sigma}) \in \mathcal{C}_u \times \mathcal{C}_\sigma} \min_{(\hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}}) \in \mathbb{E}_{em}} \mathcal{H}(\boldsymbol{u}, \boldsymbol{\sigma}, \hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}})$$

Two-Step Data-Driven Approach

$$\min_{(\hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}}) \in \mathbb{E}} \min_{(\boldsymbol{u}, \boldsymbol{\sigma}) \in \mathcal{C}_u \times \mathcal{C}_\sigma} \mathcal{H}(\boldsymbol{u}, \boldsymbol{\sigma}, \hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}}) \Rightarrow$$

$$\text{Global Step}: \; J(\hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}}) = \min_{(\boldsymbol{u}, \boldsymbol{\sigma}) \in \mathcal{C}_u \times \mathcal{C}_\sigma} \mathcal{H}(\boldsymbol{u}, \boldsymbol{\sigma}, \hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}}),$$

$$\text{Local Step}: \; (\hat{\boldsymbol{\varepsilon}}^*, \hat{\boldsymbol{\sigma}}^*) = \arg\min_{(\hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}}) \in \mathbb{E}_{em}} J(\hat{\boldsymbol{\varepsilon}}, \hat{\boldsymbol{\sigma}})$$

Data-Driven Two-Stage Minimization

Global Step (Physical Manifold)

$$\min_{\boldsymbol{u}\in\mathcal{C}_u,\boldsymbol{\sigma}\in\mathcal{V}_\sigma} \mathcal{H}(\boldsymbol{u},\boldsymbol{\sigma},\hat{\boldsymbol{\varepsilon}},\hat{\boldsymbol{\sigma}})$$

$$\text{subject to:} \quad \operatorname{div}\boldsymbol{\sigma}+\boldsymbol{b}=\boldsymbol{0} \ \text{ in } \Omega$$

$$\boldsymbol{\sigma}\cdot\boldsymbol{n}=\bar{\boldsymbol{t}} \quad \text{ on } \Gamma_t$$

Find physically admissible state $\boldsymbol{s}=(\boldsymbol{\varepsilon}[\boldsymbol{u}],\boldsymbol{\sigma})\in\mathcal{C}$ closest to a given experimental data $\hat{\boldsymbol{s}}=(\hat{\boldsymbol{\varepsilon}},\hat{\boldsymbol{\sigma}})\in\mathbb{E}_{em}$ such that the following functional is minimized

$$\mathcal{L}_{DD}(\boldsymbol{u},\boldsymbol{\sigma},\boldsymbol{\lambda},\boldsymbol{\eta})=\mathcal{H}(\boldsymbol{u},\boldsymbol{\sigma},\hat{\boldsymbol{\varepsilon}},\hat{\boldsymbol{\sigma}})+\int_\Omega \boldsymbol{\lambda}\cdot(\operatorname{div}\boldsymbol{\sigma}+\boldsymbol{b})\mathrm{d}\Omega+\int_{\Gamma_t}\boldsymbol{\eta}\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{n}-\bar{\boldsymbol{t}})\mathrm{d}\Gamma$$

The Euler-Lagrange equations reveals that $\boldsymbol{\eta}=-\boldsymbol{\lambda}$ on Γ_t

$$\delta\mathcal{L}_{DD}(\boldsymbol{u},\boldsymbol{\sigma},\boldsymbol{\lambda})=\int_\Omega\Big(\delta\boldsymbol{\varepsilon}[\boldsymbol{u}]:\boldsymbol{M}^{\varepsilon}:(\boldsymbol{\varepsilon}[\boldsymbol{u}]-\hat{\boldsymbol{\varepsilon}})+\delta\boldsymbol{\sigma}:(\boldsymbol{M}^{\sigma}:(\boldsymbol{\sigma}-\hat{\boldsymbol{\sigma}})\Big)\mathrm{d}\Omega-\int_\Omega\delta\boldsymbol{\sigma}:\boldsymbol{\varepsilon}[\boldsymbol{\lambda}]\mathrm{d}\Omega$$
$$+\int_{\Gamma_u}(\delta\boldsymbol{\sigma}\cdot\boldsymbol{n})\cdot\boldsymbol{\lambda}\mathrm{d}\Gamma-\int_\Omega\delta\boldsymbol{\varepsilon}[\boldsymbol{\lambda}]:\boldsymbol{\sigma}\mathrm{d}\Omega+\int_\Omega\delta\boldsymbol{\lambda}\cdot\boldsymbol{b}\mathrm{d}\Omega+\int_{\Gamma_t}\delta\boldsymbol{\lambda}\cdot\bar{\boldsymbol{t}}\mathrm{d}\Gamma$$

$$\int_\Omega\delta\boldsymbol{\varepsilon}[\boldsymbol{\lambda}]:\boldsymbol{\sigma}\mathrm{d}\Omega=\int_\Omega\delta\boldsymbol{\lambda}\cdot\boldsymbol{b}\mathrm{d}\Omega+\int_{\Gamma_t}\delta\boldsymbol{\lambda}\cdot\bar{\boldsymbol{t}}\mathrm{d}\Gamma$$

$$\int_\Omega\delta\boldsymbol{\varepsilon}[\boldsymbol{u}]:\boldsymbol{M}^{\varepsilon}:\boldsymbol{\varepsilon}[\boldsymbol{u}]\mathrm{d}\Omega=\int_\Omega\delta\boldsymbol{\varepsilon}[\boldsymbol{u}]:\boldsymbol{M}^{\varepsilon}:\hat{\boldsymbol{\varepsilon}}\mathrm{d}\Omega$$

$$\int_\Omega\delta\boldsymbol{\sigma}:(\boldsymbol{M}^{\sigma}:\boldsymbol{\sigma}-\boldsymbol{\varepsilon}[\boldsymbol{\lambda}])\mathrm{d}\Omega=\int_\Omega\delta\boldsymbol{\sigma}:\boldsymbol{M}^{\sigma}:\hat{\boldsymbol{\sigma}}\mathrm{d}\Omega$$

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

Discretization

$$\boldsymbol{u}(\boldsymbol{x}) \approx \boldsymbol{u}^h(\boldsymbol{x}) = \sum_{I=1}^{N} \Psi_I(\boldsymbol{x}) \boldsymbol{d}_I,$$

$$\boldsymbol{\lambda}(\boldsymbol{x}) \approx \boldsymbol{\lambda}^h(\boldsymbol{x}) = \sum_{I=1}^{N} \Psi_I(\boldsymbol{x}) \boldsymbol{\Lambda}_I,$$

$$\boldsymbol{\sigma}(\boldsymbol{x}) \approx \boldsymbol{\sigma}^h(\boldsymbol{x}) = \sum_{\alpha=1}^{m} \chi_\alpha(\boldsymbol{x}) \boldsymbol{\sigma}_\alpha,$$

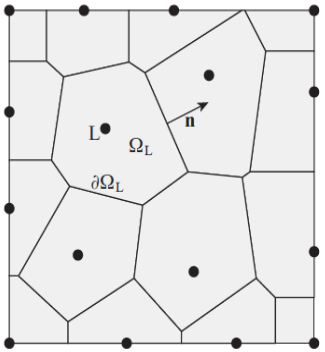
$\{\Psi_I\}_{I=1}^{N}$: Reproducing kernel shape functon

$\chi_\alpha(\boldsymbol{x})$: indicator function, $\chi_\alpha(\boldsymbol{x}) = 1$ if $\boldsymbol{x} \in \Omega_\alpha$, $\chi_\alpha(\boldsymbol{x}) = 0$ if $\boldsymbol{x} \notin \Omega_\alpha$

Nodal Integration + SCNI

$$\int_\Omega \delta\boldsymbol{\varepsilon}[\boldsymbol{u}] : \boldsymbol{M}^\varepsilon : \boldsymbol{\varepsilon}[\boldsymbol{u}] \mathrm{d}\Omega = \int_\Omega \delta\boldsymbol{\varepsilon}[\boldsymbol{u}] : \boldsymbol{M}^\varepsilon : \hat{\boldsymbol{\varepsilon}} \mathrm{d}\Omega$$

$$\Rightarrow \sum_{J=1}^{N} \left(\sum_{\alpha=1}^{m} V_\alpha \tilde{\boldsymbol{B}}_{\alpha I}^T \boldsymbol{M}_\alpha^\varepsilon \tilde{\boldsymbol{B}}_{\alpha J} \right) \boldsymbol{d}_J = \sum_{\alpha=1}^{m} V_\alpha \tilde{\boldsymbol{B}}_{\alpha I}^T \boldsymbol{M}_\alpha^\varepsilon \hat{\boldsymbol{\varepsilon}}_\alpha, \; I = 1, \ldots, N$$



$$\tilde{\boldsymbol{B}}_{LI} \equiv \tilde{\boldsymbol{B}}_I(\boldsymbol{x}_L) = \begin{bmatrix} \tilde{b}_{I1}(\boldsymbol{x}_L) & 0 \\ 0 & \tilde{b}_{I2}(\boldsymbol{x}_L) \\ \tilde{b}_{I2}(\boldsymbol{x}_L) & \tilde{b}_{I1}(\boldsymbol{x}_L) \end{bmatrix}$$

$$\tilde{b}_{Ii}(\boldsymbol{x}_L) = \frac{1}{V_L} \int_{\partial\Omega_L} \Psi_I(\boldsymbol{x}) n_i(\boldsymbol{x}) \mathrm{d}\Gamma$$

$$\int_\Omega \delta\boldsymbol{\varepsilon}[\boldsymbol{\lambda}] : \boldsymbol{\sigma} \mathrm{d}\Omega = \int_\Omega \delta\boldsymbol{\lambda} \cdot \boldsymbol{b} \mathrm{d}\Omega + \int_{\Gamma_t} \delta\boldsymbol{\lambda} \cdot \bar{\boldsymbol{t}} \mathrm{d}\Gamma$$

$$\Rightarrow \sum_{\alpha=1}^{m} V_\alpha \tilde{\boldsymbol{B}}_{\alpha I}^T \boldsymbol{\sigma}_\alpha = \boldsymbol{f}_I, \; I = 1, \ldots, N \qquad \Rightarrow \sum_{J=1}^{N} \left(\sum_{\alpha=1}^{m} V_\alpha \tilde{\boldsymbol{B}}_{\alpha I}^T \boldsymbol{M}_\alpha^{\sigma-1} \tilde{\boldsymbol{B}}_{\alpha J} \right) \boldsymbol{\Lambda}_J = \boldsymbol{f}_I - \sum_{\alpha=1}^{m} V_\alpha \tilde{\boldsymbol{B}}_{\alpha I}^T \hat{\boldsymbol{\sigma}}_\alpha, \; I = 1, \ldots, N$$

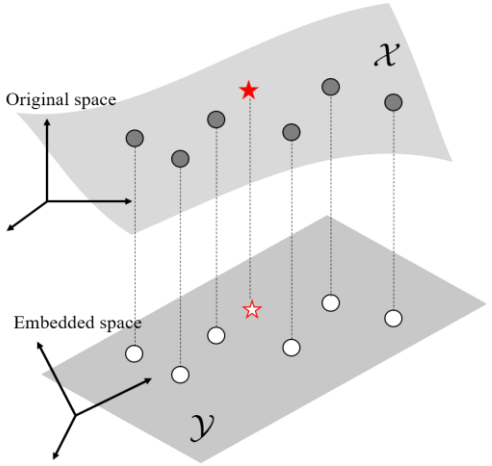
Collocation + SCNI

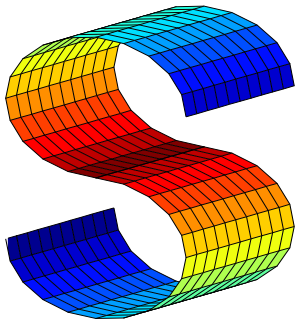
$$\int_\Omega \delta\boldsymbol{\sigma} : (\boldsymbol{M}^\sigma : \boldsymbol{\sigma} - \boldsymbol{\varepsilon}[\boldsymbol{\lambda}]) \mathrm{d}\Omega = \int_\Omega \delta\boldsymbol{\sigma} : \boldsymbol{M}^\sigma : \hat{\boldsymbol{\sigma}} \mathrm{d}\Omega$$

$$\Rightarrow \boldsymbol{M}_\alpha^\sigma \boldsymbol{\sigma}_\alpha - \sum_{I=1}^{N} \tilde{\boldsymbol{B}}_{\alpha I} \boldsymbol{\Lambda}_I = \boldsymbol{M}_\alpha^\sigma \hat{\boldsymbol{\sigma}}_\alpha, \; \alpha = 1, \ldots, m \quad \Rightarrow \boldsymbol{\sigma}_\alpha = \hat{\boldsymbol{\sigma}}_\alpha + \boldsymbol{M}_\alpha^{\sigma-1} \sum_{I=1}^{N} \tilde{\boldsymbol{B}}_{\alpha I} \boldsymbol{\Lambda}_I, \; \alpha = 1, \ldots, m$$

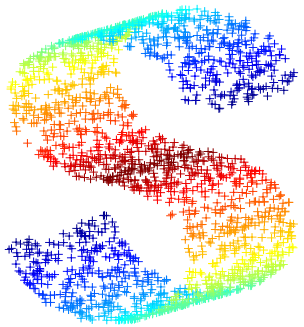
Manifold Learning (Nonlinear Dimensionality Reduction)

Observation: Naturally occurring data usually reside on a lower dimensional submanifold (Lee and Verleysen 2007; Ibañez et al. 2016)

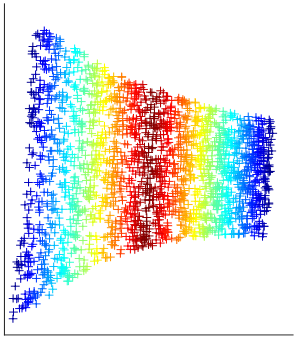




Synthetic manifold



Raw data



Low-dim. representation

Locally linear embedding (LLE) approach

Let $\hat{\boldsymbol{s}}^i = (\hat{\boldsymbol{\varepsilon}}^i, \hat{\boldsymbol{\sigma}}^i)$

$$\{w_{ij}\}_{i,j=1,\ldots,p} = \arg\min \sum_{i=1}^{p} \| \hat{\boldsymbol{s}}^i - \sum_{j=1, j\neq i}^{p} w_{ij}^* \hat{\boldsymbol{s}}^j \|^2$$

$$\text{subject to: } , \sum_{j=1, j\neq i}^{p} w_{ij}^* = 1, \; i = 1, \ldots, p$$

$$w_{ij}^* = 0 \;\text{ if }\; j \notin \mathcal{N}_k(\hat{\boldsymbol{s}}^i)$$

$$\hat{\boldsymbol{s}}^i \leftarrow \sum_{j \in \mathcal{N}_k(\hat{\boldsymbol{s}}^i)} w_{ij} \hat{\boldsymbol{s}}^j$$

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

Local Step: Local Convexity Data-Driven Solver

$$\text{Global Step}: \min_{(\boldsymbol{u},\boldsymbol{\sigma})\in\mathcal{C}_u\times\mathcal{C}_\sigma} \mathcal{H}(\boldsymbol{u},\boldsymbol{\sigma},\hat{\boldsymbol{\varepsilon}},\hat{\boldsymbol{\sigma}}) \rightarrow s_\alpha = (\boldsymbol{\varepsilon}_\alpha(\boldsymbol{u}),\boldsymbol{\sigma}_\alpha)$$

Project the local state onto the convex hull of $\{\hat{\boldsymbol{s}}^i\}_{i\in\mathcal{N}_k(\boldsymbol{s}_\alpha)}$ associated with $\boldsymbol{s}_\alpha$

$$\mathcal{E}(\boldsymbol{s}_\alpha) = \text{Conv}(\{\hat{\boldsymbol{s}}^i\}_{i\in\mathcal{N}_k(\boldsymbol{s}_\alpha)}) = \left\{ \sum_{i\in\mathcal{N}_k(\boldsymbol{s}_\alpha)} w_i\hat{\boldsymbol{s}}^i \,\middle|\, \sum_{i\in\mathcal{N}_k(\boldsymbol{s}_\alpha)} w_i = 1, \text{ and } w_i \geq 0, \forall i \in \mathcal{N}_k(\boldsymbol{s}_\alpha) \right\},$$

Optimal Reconstruction

$$\boldsymbol{w}_\alpha^* = \arg\min_{\boldsymbol{w}\in\mathbb{R}^k} \| \boldsymbol{s}_\alpha - \sum_{i\in\mathcal{N}_k(\boldsymbol{s}_\alpha)} w_i\hat{\boldsymbol{s}}^i \|_M^2$$

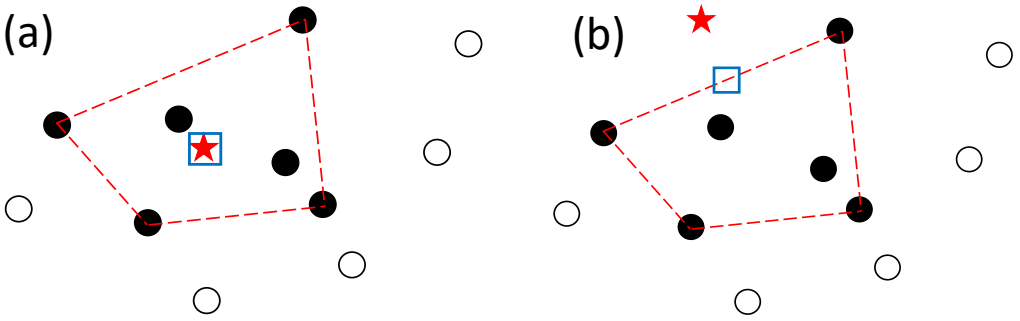
$$\text{subject to: } \sum_{i\in\mathcal{N}_k(\boldsymbol{s}_\alpha)} w_i = 1,$$

$$w_i \geq 0, \ \forall i \in \mathcal{N}_k(\boldsymbol{s}_\alpha),$$

Given the data-set neighbors $\{\hat{\boldsymbol{s}}^i\}_{i\in\mathcal{N}_k(\boldsymbol{s}_\alpha)} \subset \mathbb{E}$ associated with $\boldsymbol{s}_\alpha$, construct $\mathcal{E}(\boldsymbol{s}_\alpha)$ such that

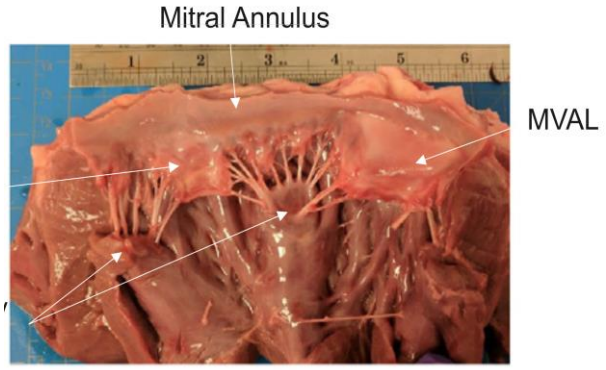
$$\hat{\boldsymbol{s}}_\alpha^* = \arg\min_{\hat{\boldsymbol{s}}_\alpha\in\mathcal{E}_\alpha} \| \boldsymbol{s}_\alpha - \hat{\boldsymbol{s}}_\alpha \|_M^2,$$

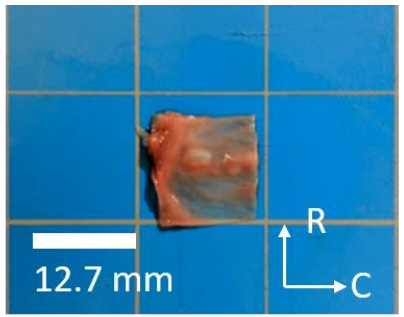
- Inside material manifold
- Outside material manifold

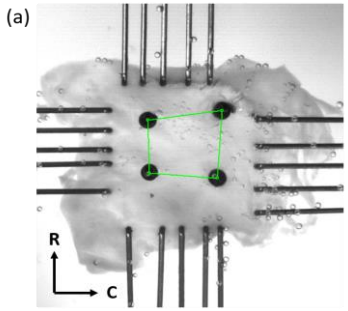


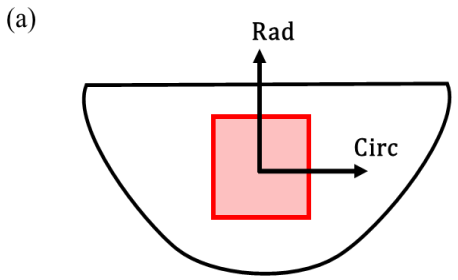
- ✓ Enhance Robustness
- ✓ Avoid across-cluster instability

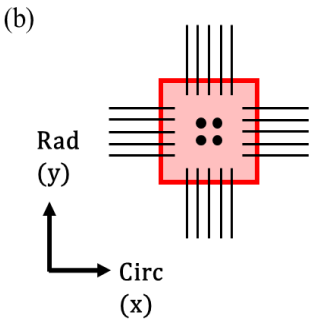
X. He and J. S. Chen, CMAME, 2019

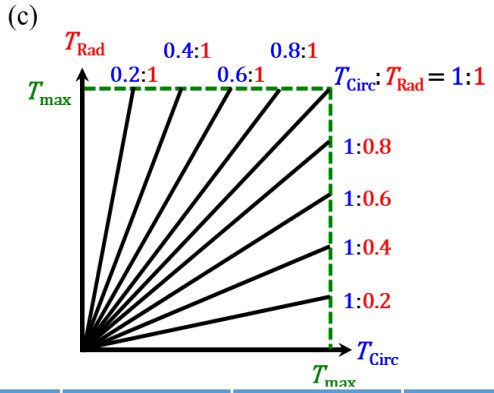


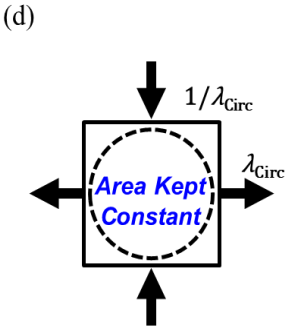












Protocol ID	Testing	λ_{Circ}	λ_{Rad}	u_x (mm)	u_y (mm)
1	Biaxial Tension T_{Circ}:T_{Rad} = 1:1	1.333	1.525	2.498	3.938
2	Biaxial Tension T_{Circ}:T_{Rad} = 1:0.8	1.342	1.499	2.564	3.744
3	Biaxial Tension T_{Circ}:T_{Rad} = 1:0.6	1.355	1.466	2.662	3.498
4	Biaxial Tension T_{Circ}:T_{Rad} = 1:0.4	1.369	1.415	2.770	3.110
5	Biaxial Tension T_{Circ}:T_{Rad} = 1:0.2	1.388	1.326	2.913	2.442
6	Biaxial Tension T_{Circ}:T_{Rad} = 0.8:1	1.313	1.541	2.344	4.055
7	Biaxial Tension T_{Circ}:T_{Rad} = 0.6:1	1.275	1.562	2.064	4.215
8	Biaxial Tension T_{Circ}:T_{Rad} = 0.4:1	1.213	1.588	1.596	4.409
9	Biaxial Tension T_{Circ}:T_{Rad} = 0.2:1	1.109	1.618	0.820	4.635
10	Pure Shear in x (x: tension, y: compression)	1.387	0.721	2.903	-2.093
11	Pure Shear in y (x: compression, y: tension)	0.620	1.612	-2.847	4.590

Tong-Fung model (1976)

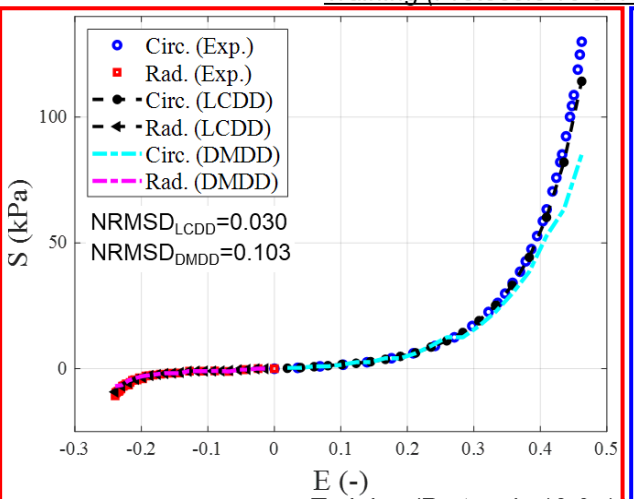
$$W = \frac{c}{2}\left[\exp\left(a_1 E_{CC}^2 + a_2 E_{RR}^2 + 2a_3 E_{CC} E_{RR}\right) - 1\right]$$

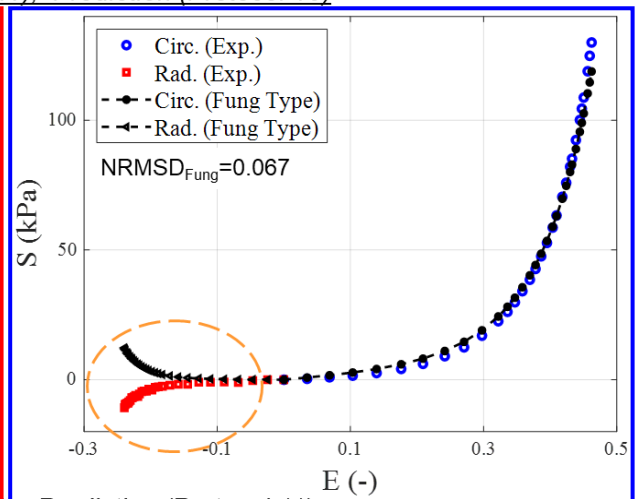
Lee-Sacks model (2014)

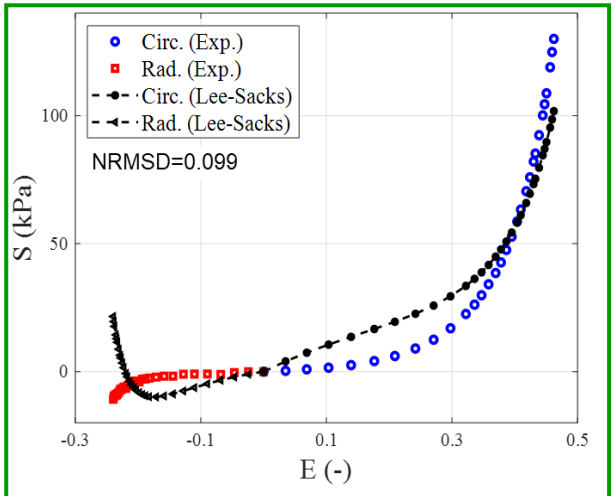
$$W = C_{10}\left(I_1 - 3\right) + \frac{c_0}{2}\left\{\delta \exp\left[c_1\left(I_1 - 3\right)^2\right] + \left(1-\delta\right)\exp\left[c_2\left(I_4 - 1\right)^2\right] - 1\right\}$$

Pure Shear Training and Prediction

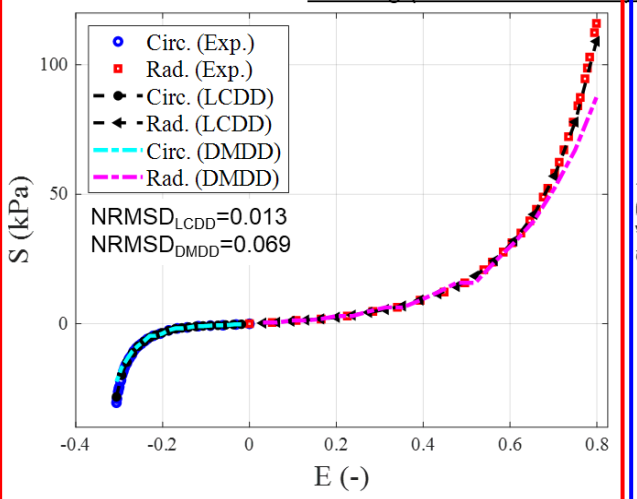
Training (Protocols 10 & 11); Prediction (Protocol 10)

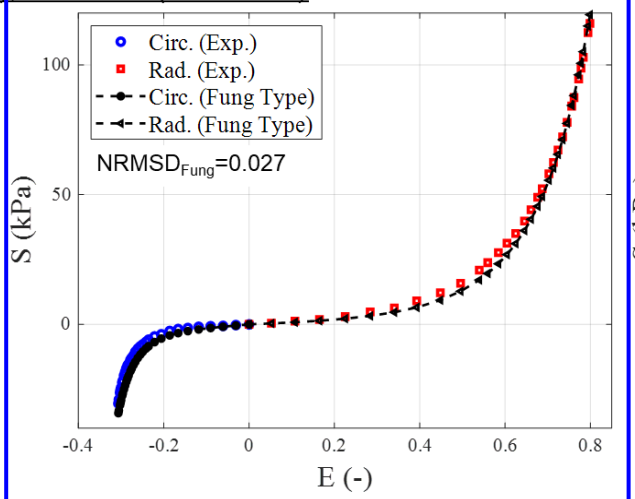


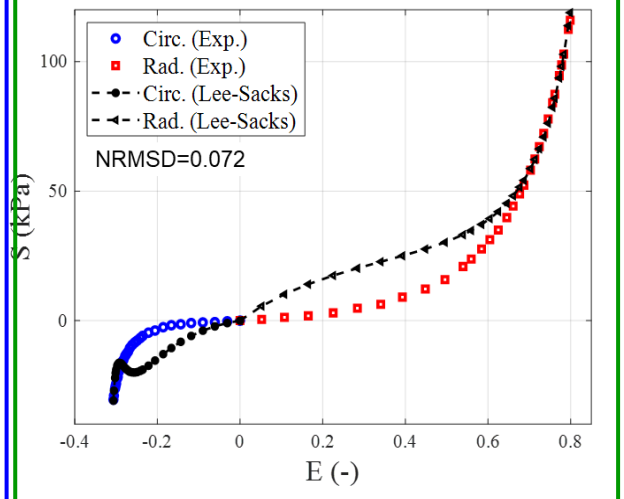




Training (Protocols 10 & 11); Prediction (Protocol 11)







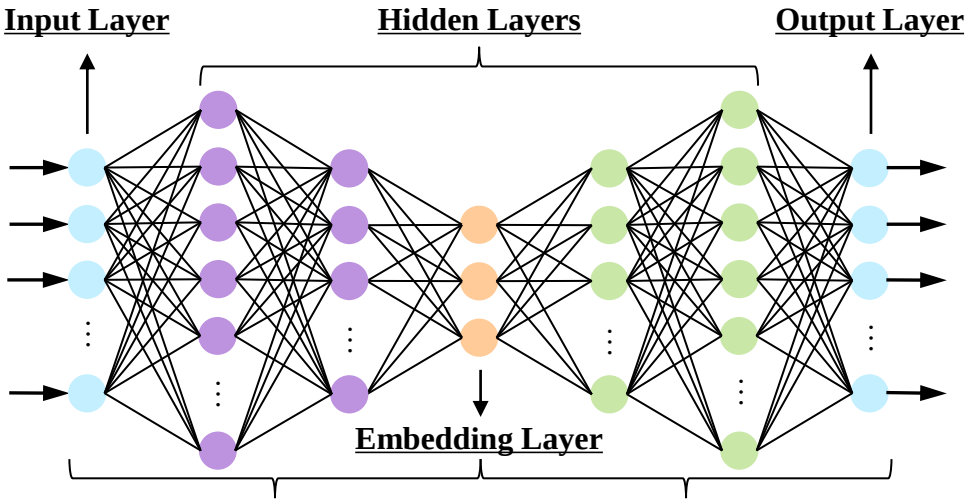
Model Free Data-Driven

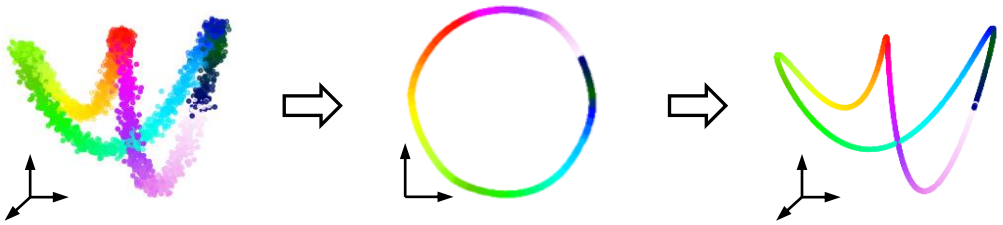
Tong-Fung Model

Lee-Sacks model

He, Q., Laurence, D. W., Lee, C. H. and Chen, J. S., "Manifold Learning Based Data-Driven Modeling for Soft Biological Tissues," Journal of Biomechanics, Vol 117, 110124, 2021, https://doi.org/10.1016/j.jbiomech.2020.110124

Deep Autoencoders



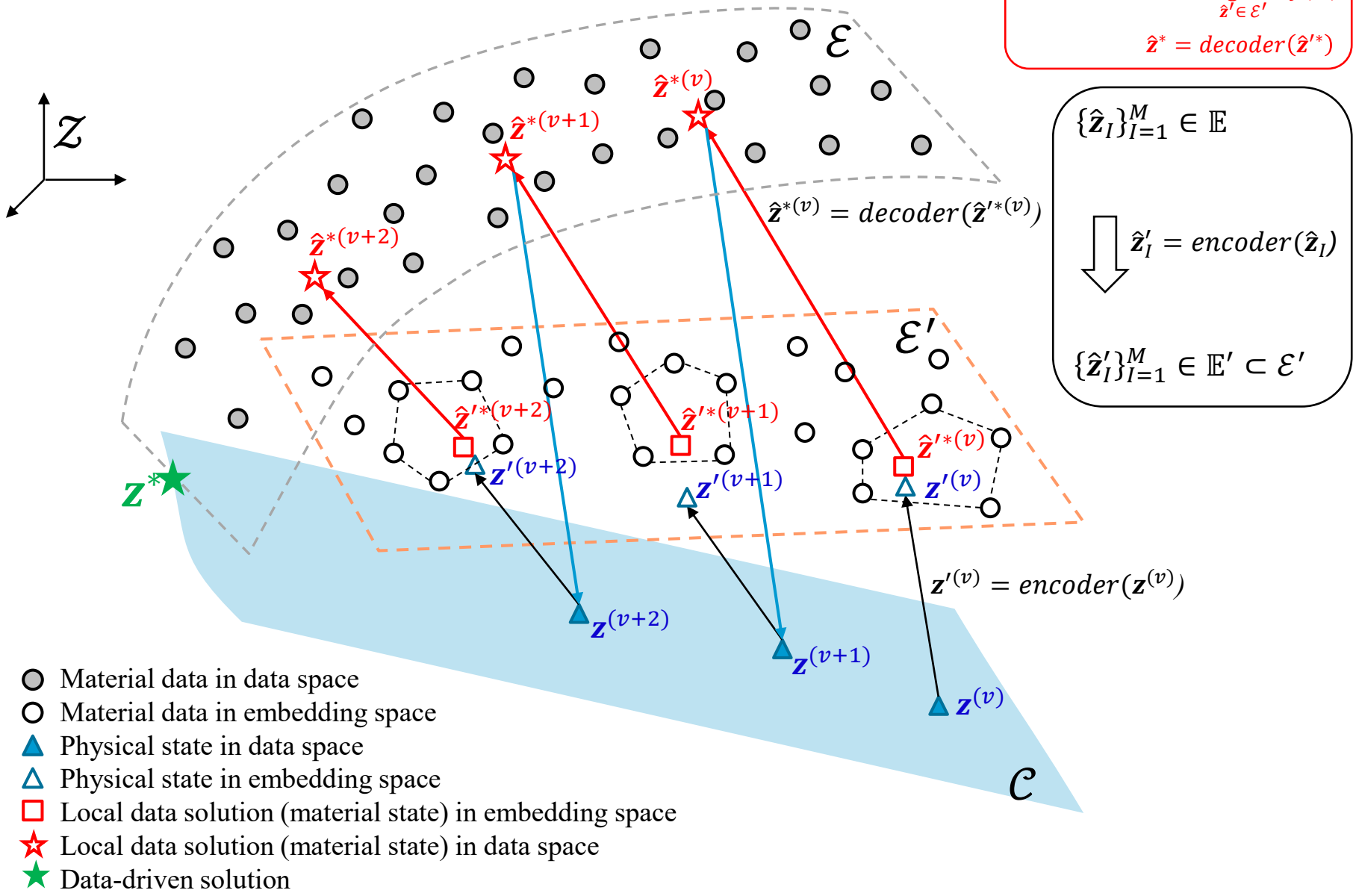


High-Dimensional Object | **Low-Dimensional Embedding** | **Reconstructed Object**

- $dim(\text{Embedding}) < dim(\text{Input})$:
  - Noise filtering
  - Underlying data structure
- Deep network architectures:
  - Strong learning and generalization capability
  - Learning nonlinear embedding with exponentially less data
  - Efficient data projection:
    - *Encoder*: data space → embedding space
    - *Decoder*: embedding space → data space

He, X., He, Q., Chen, J. S., "Deep autoencoders for physics-constrained data-driven nonlinear materials modeling," Under Revision, Computer Methods in Applied Mechanics and Engineering, https://doi.org/10.1016/j.cma.2021.114016, Vol. 385, 2021.

Auto-Embedding Data-Driven (AEDD)



He, X., He Q., Chen, J. S., Deep autoencoders for physics-constrained data-driven nonlinear modeling. Under review

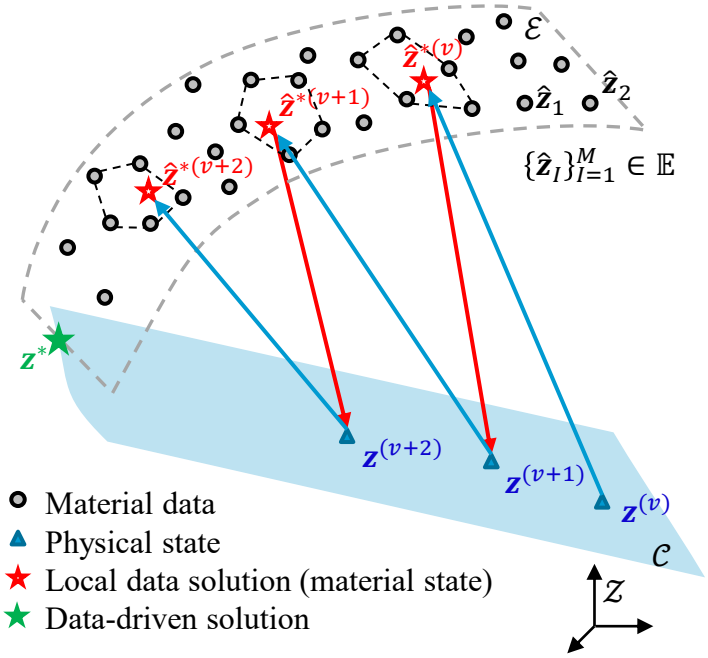
Local Convexity Data-Driven (LCDD) vs Auto-Embedding Data-Driven (AEDD)

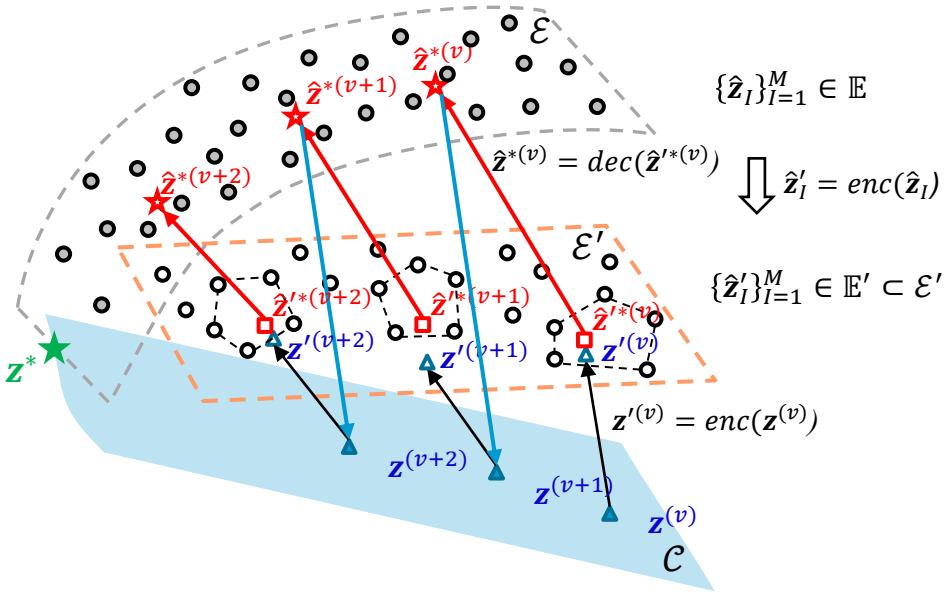
LCDD

- Online learning local convex manifold of material data
- Online local convexity-preserving reconstruction (non-negative least-square solver) of optimal data points in high-dim. data space

AEDD

- Offline learning a global manifold of material data by autoencoders
- Online local convexity-preserving reconstruction of optimal data points in low-dim. embedding space
- Higher computational efficiency
- Stronger generalization ability





He, X., He, Q., Chen, J. S., "Deep autoencoders for physics-constrained data-driven nonlinear materials modeling," Under Revision, Computer Methods in Applied Mechanics and Engineering, https://doi.org/10.1016/j.cma.2021.114016, Vol. 385, 2021.

Auto-Embedding Data-Driven (AEDD) Computing

Physical Constraints

Equilibrium:
$$\mathrm{DIV}(\boldsymbol{F}(\boldsymbol{u})\cdot\boldsymbol{S})+\boldsymbol{b}=\boldsymbol{0} \quad \text{in } \Omega^X$$
$$(\boldsymbol{F}(\boldsymbol{u})\cdot\boldsymbol{S})\cdot\boldsymbol{N}=\boldsymbol{t} \quad \text{on } \Gamma_t^X$$

Compatibility:
$$\boldsymbol{E}=\boldsymbol{E}(\boldsymbol{u})=(\boldsymbol{F}^T\boldsymbol{F}-\boldsymbol{I})/2 \quad \text{in } \Omega^X$$
$$\boldsymbol{u}=\boldsymbol{g} \quad \text{on } \Gamma_u^X$$

Distance Functional:

$$\mathcal{H}(\boldsymbol{z},\hat{\boldsymbol{z}})=\int_{\Omega^X} d^2(\boldsymbol{z},\hat{\boldsymbol{z}})\,d\Omega=\int_{\Omega^X} d_E^2\left(\boldsymbol{E},\hat{\boldsymbol{E}}\right)+d_S^2\left(\boldsymbol{S},\hat{\boldsymbol{S}}\right)d\Omega$$

Iteration

Global Step: $J(\hat{\boldsymbol{z}})=\min_{\boldsymbol{z}\in\mathcal{C}}\mathcal{H}(\boldsymbol{z},\hat{\boldsymbol{z}})$

Material Database

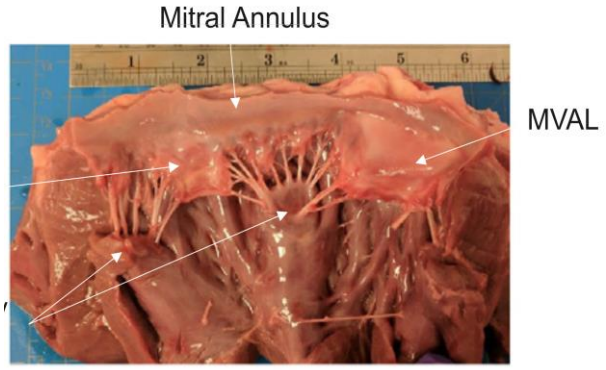
Autoencoder-based Manifold Learning

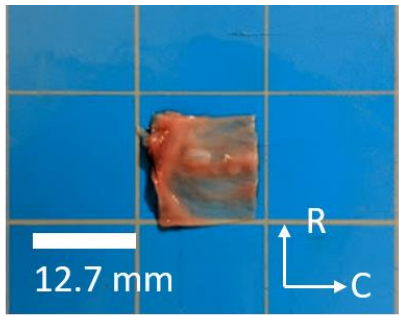
Local Step:
$$\boldsymbol{z}'=encoder(\boldsymbol{z})$$
$$\hat{\boldsymbol{z}}'^{*}=\arg\min_{\hat{\boldsymbol{z}}'\in\mathcal{E}'} J(\hat{\boldsymbol{z}}')$$
$$\hat{\boldsymbol{z}}^{*}=decoder(\hat{\boldsymbol{z}}'^{*})$$

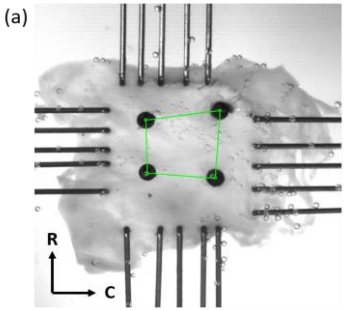
High-Dim. Data Space ($\mathcal{C}$ or $\mathcal{E}$) $\xrightarrow{encoder}$ $\xleftarrow{decoder}$ Low-Dim. Embedding Space ($\mathcal{E}'$)

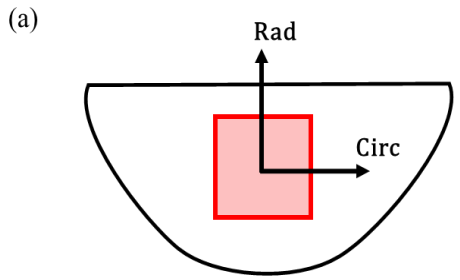
	Physical State	Material State
High-Dim. Data Space	$\boldsymbol{z}=(\boldsymbol{E}(\boldsymbol{u}),\boldsymbol{S})\in\mathcal{C}$	$\hat{\boldsymbol{z}}=\left(\hat{\boldsymbol{E}},\hat{\boldsymbol{S}}\right)\in\mathbb{E}=\left\{\left(\hat{\boldsymbol{E}}^I,\hat{\boldsymbol{S}}^I\right)\right\}_{I=1}^{M}$
Low-Dim. Embedding Space	$\boldsymbol{z}'=encoder(\boldsymbol{z})\in\mathcal{E}'$	$\hat{\boldsymbol{z}}'=encoder(\hat{\boldsymbol{z}})\in\mathbb{E}'=\left\{\left(\hat{\boldsymbol{E}}'^I,\hat{\boldsymbol{S}}'^I\right)\right\}_{I=1}^{M}\subset\mathcal{E}'$

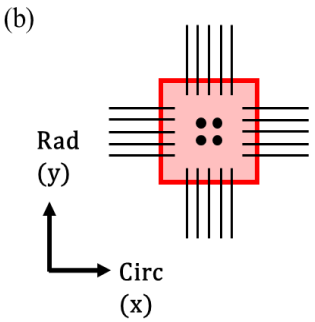
He, X., He, Q., Chen, J. S., "Deep autoencoders for physics-constrained data-driven nonlinear materials modeling," Under Revision, Computer Methods in Applied Mechanics and Engineering, https://doi.org/10.1016/j.cma.2021.114016, Vol. 385, 2021.

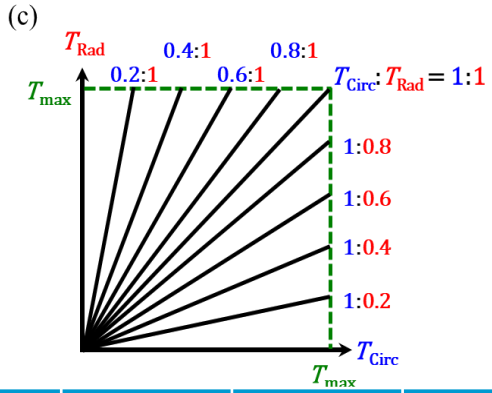


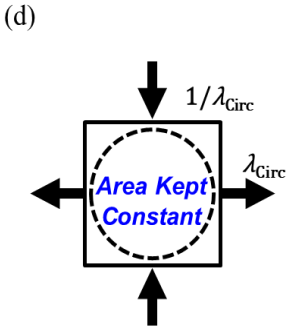












Protocol ID	Testing	λ_{Circ}	λ_{Rad}	u_x (mm)	u_y (mm)
1	Biaxial Tension T_{Circ}:T_{Rad} = 1:1	1.333	1.525	2.498	3.938
2	Biaxial Tension T_{Circ}:T_{Rad} = 1:0.8	1.342	1.499	2.564	3.744
3	Biaxial Tension T_{Circ}:T_{Rad} = 1:0.6	1.355	1.466	2.662	3.498
4	Biaxial Tension T_{Circ}:T_{Rad} = 1:0.4	1.369	1.415	2.770	3.110
5	Biaxial Tension T_{Circ}:T_{Rad} = 1:0.2	1.388	1.326	2.913	2.442
6	Biaxial Tension T_{Circ}:T_{Rad} = 0.8:1	1.313	1.541	2.344	4.055
7	Biaxial Tension T_{Circ}:T_{Rad} = 0.6:1	1.275	1.562	2.064	4.215
8	Biaxial Tension T_{Circ}:T_{Rad} = 0.4:1	1.213	1.588	1.596	4.409
9	Biaxial Tension T_{Circ}:T_{Rad} = 0.2:1	1.109	1.618	0.820	4.635
10	Pure Shear in x (x: tension, y: compression)	1.387	0.721	2.903	-2.093
11	Pure Shear in y (x: compression, y: tension)	0.620	1.612	-2.847	4.590

Tong-Fung model (1976)

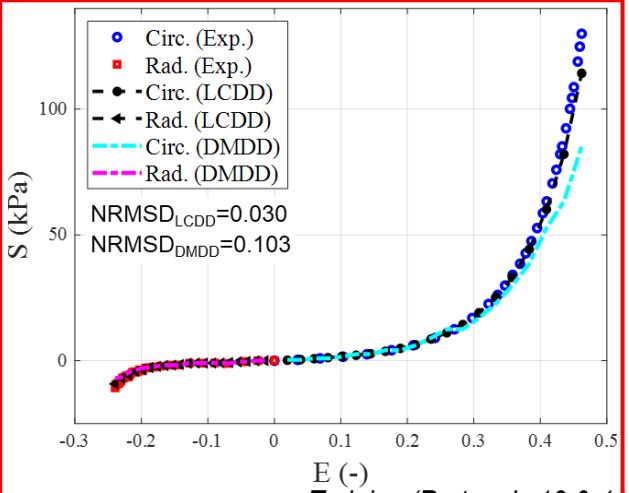
$$W = \frac{c}{2}\left[\exp\left(a_1 E_{CC}^2 + a_2 E_{RR}^2 + 2a_3 E_{CC} E_{RR}\right) - 1\right]$$

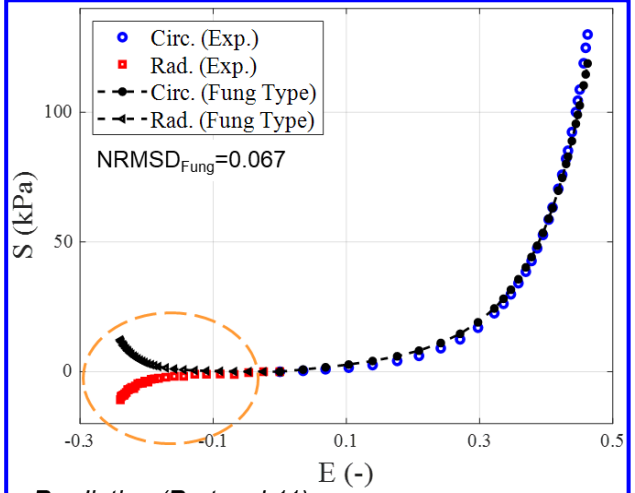
Lee-Sacks model (2014)

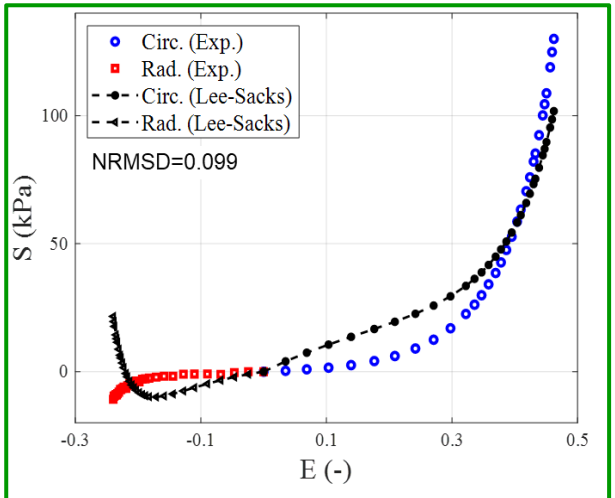
$$W = C_{10}\left(I_1 - 3\right) + \frac{c_0}{2}\left\{\delta \exp\left[c_1\left(I_1 - 3\right)^2\right] + \left(1-\delta\right)\exp\left[c_2\left(I_4 - 1\right)^2\right] - 1\right\}$$

Pure Shear Training and Prediction

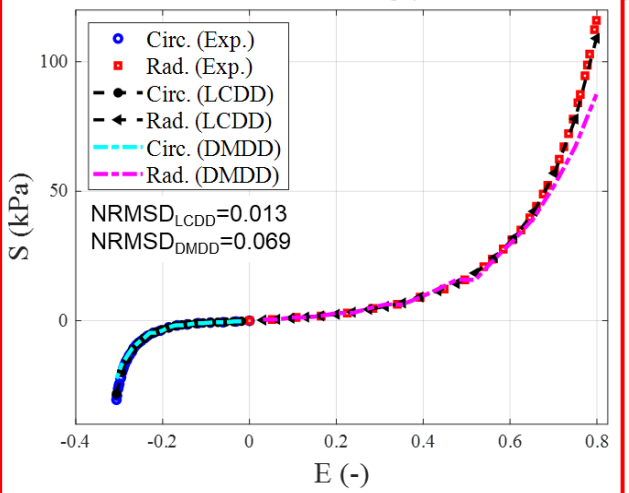
Training (Protocols 10 & 11); Prediction (Protocol 10)

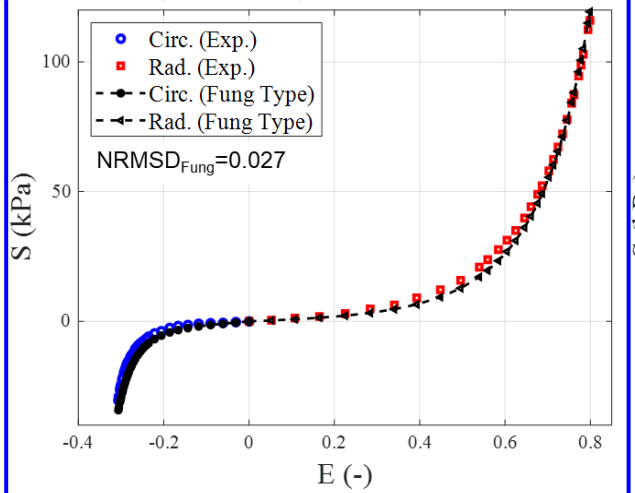


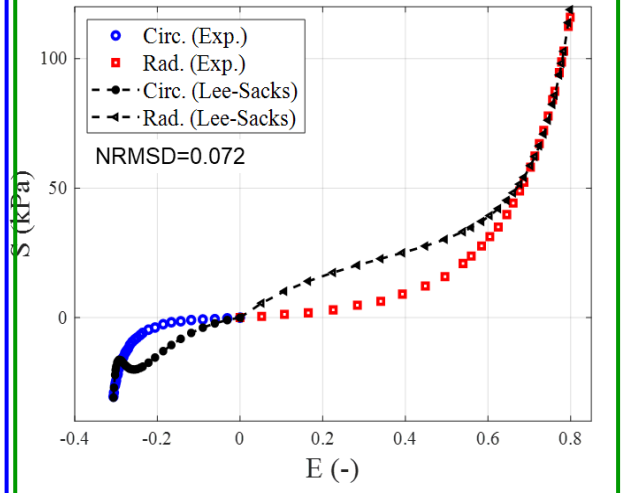




Training (Protocols 10 & 11); Prediction (Protocol 11)







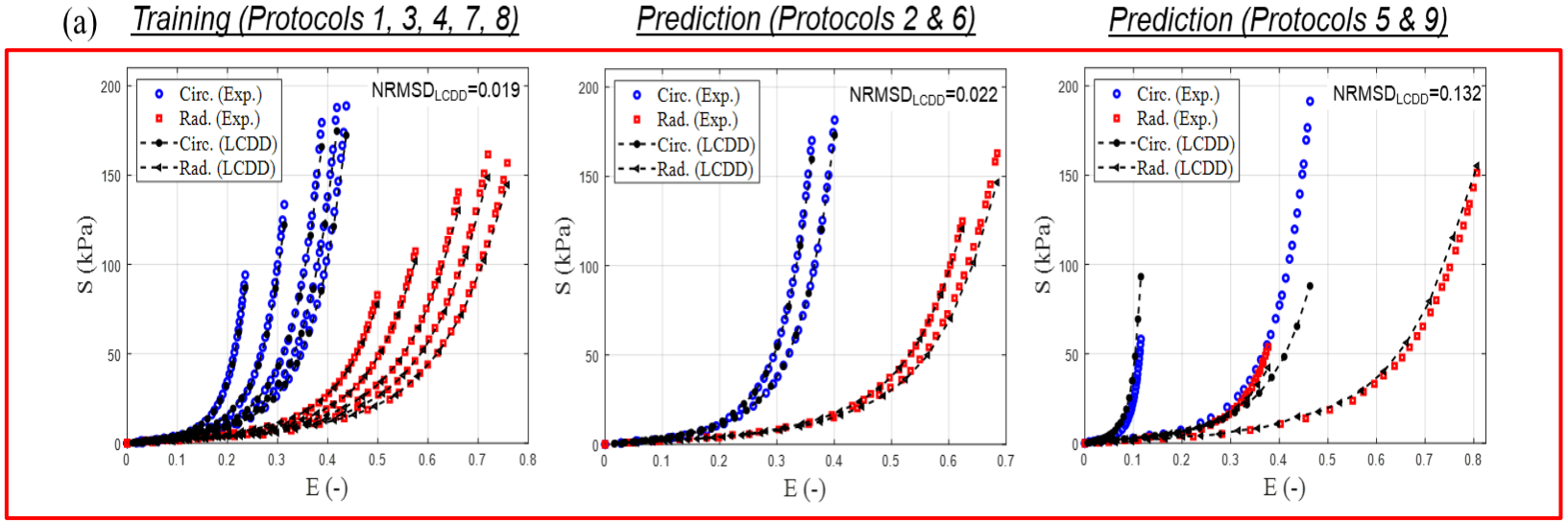
Model Free Data-Driven

Tong-Fung Model

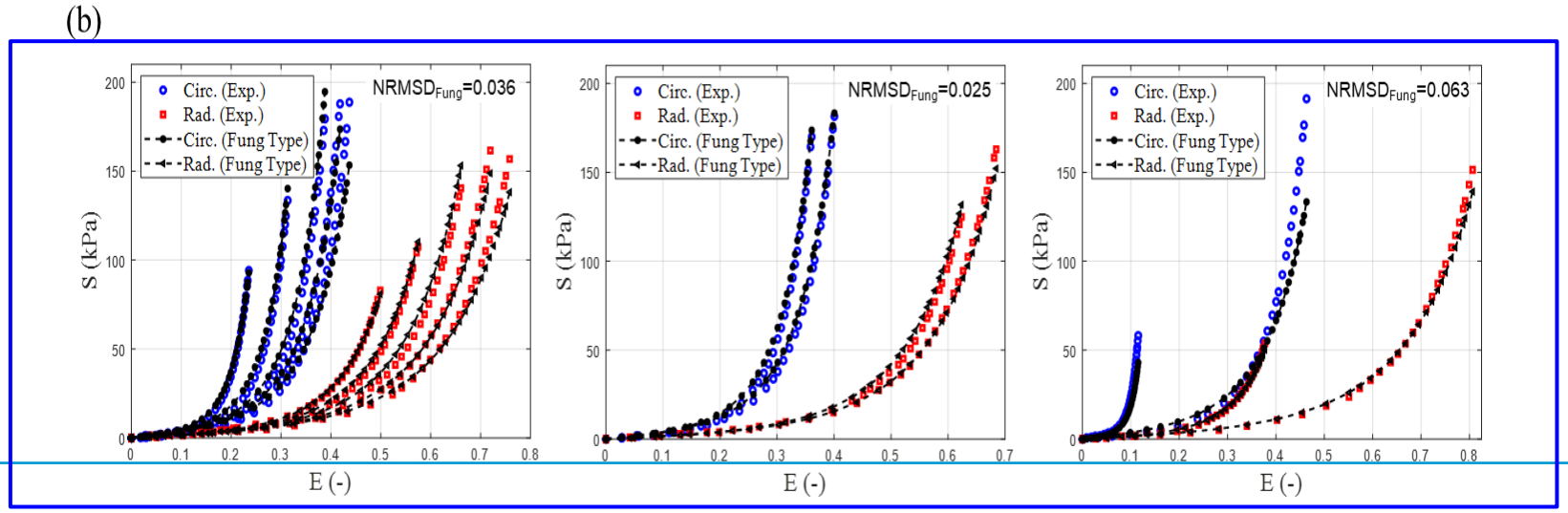
Lee-Sacks model

Training: Biaxials
Prediction: Interpolated & Exterpolated Biaxials

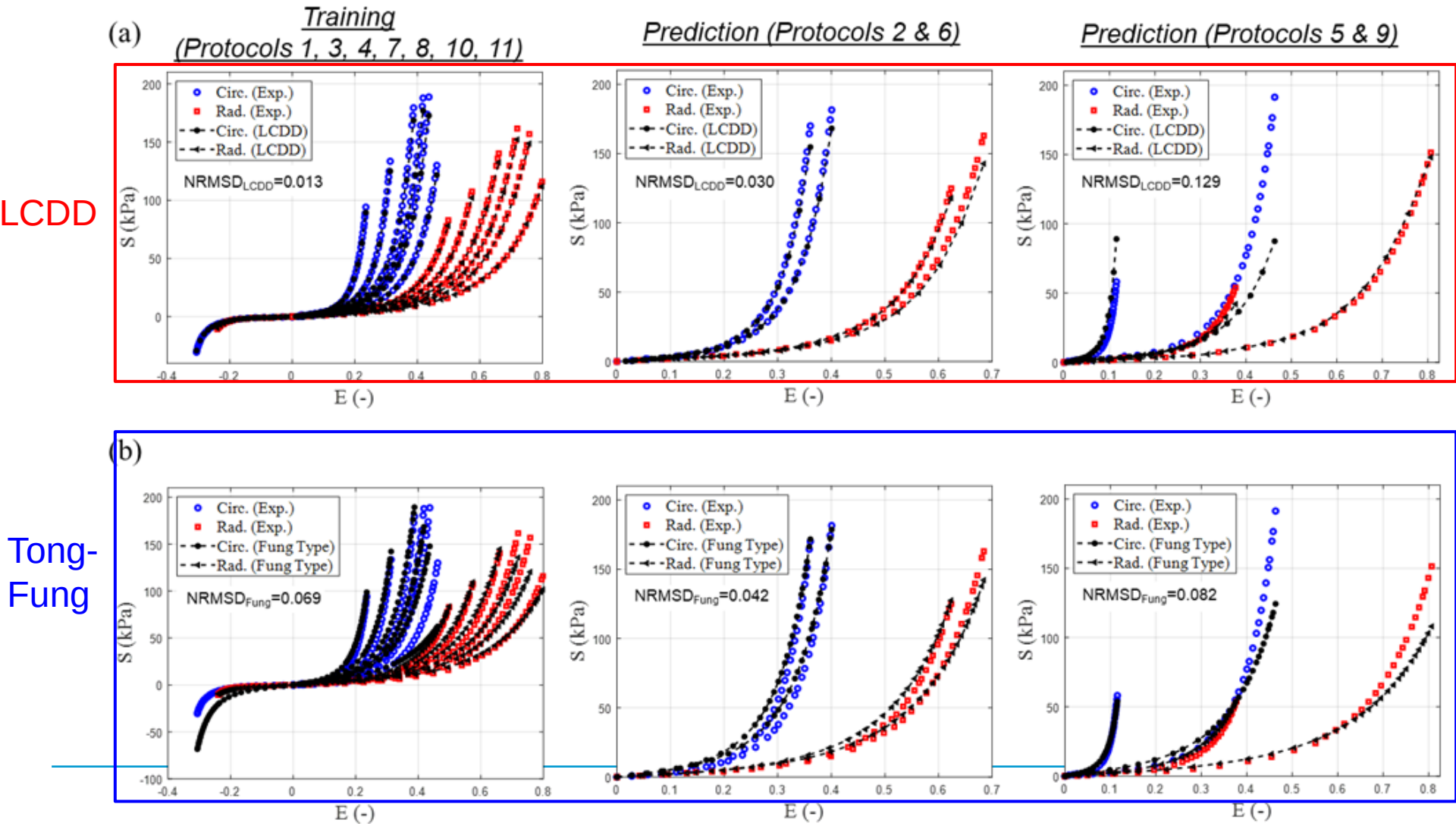
LCDD



Tong-Fung



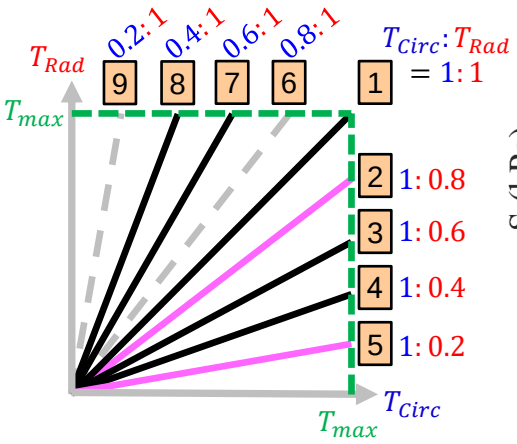
Training: Biaxials & Pure Shears
Prediction: Interpolated & Exterpolated Biaxials



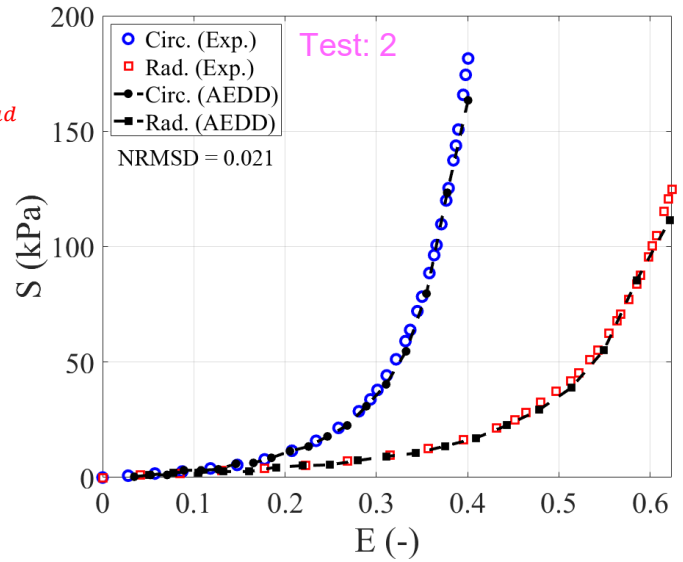
Biological Tissue Modeling

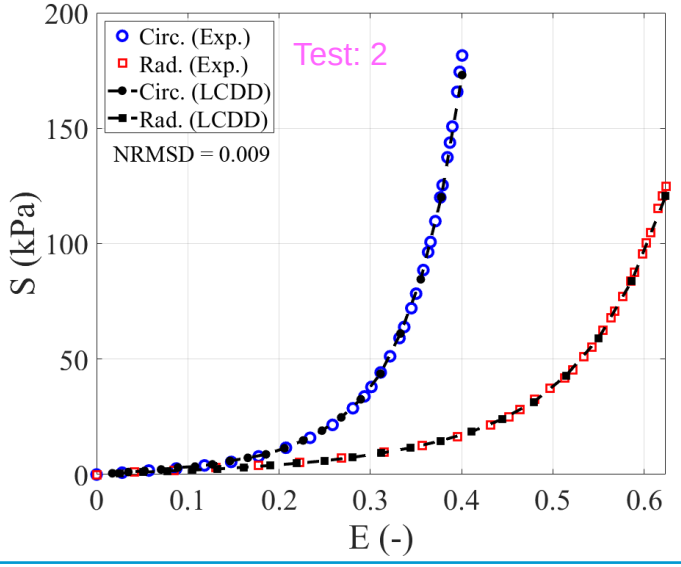
Train: 1, 3, 4, 7, 8

Test: 2, 5

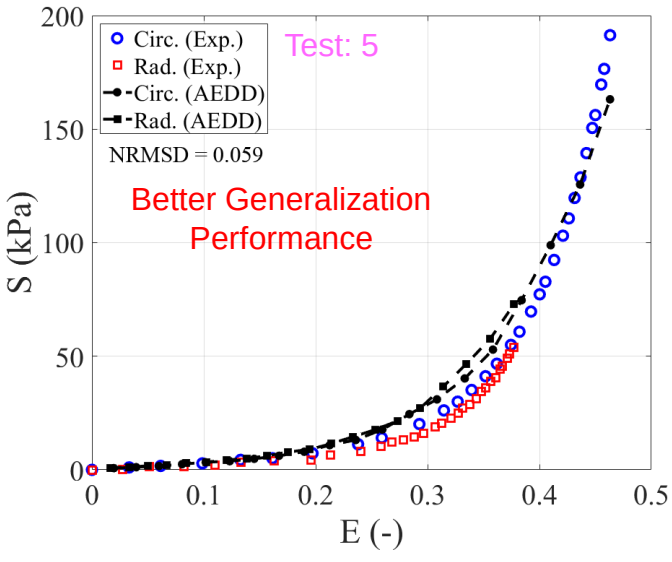


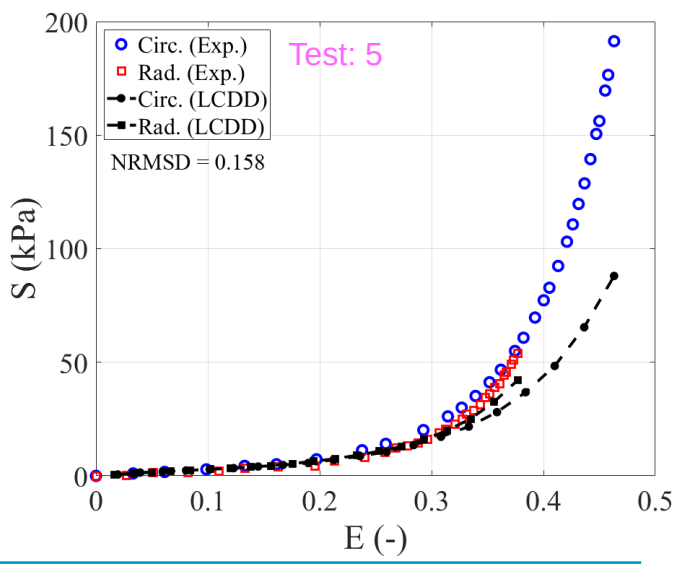
Interpolative prediction





Extrapolative prediction





He, Q., Laurence, D. W., Lee, C. H., Chen, J. S., Manifold learning based data-driven modeling for soft biological tissues. J. Biomech, 117, 110124, 2021

He, X., He Q., Chen, J. S., Deep autoencoders for physics-constrained data-driven nonlinear modeling. Under review.

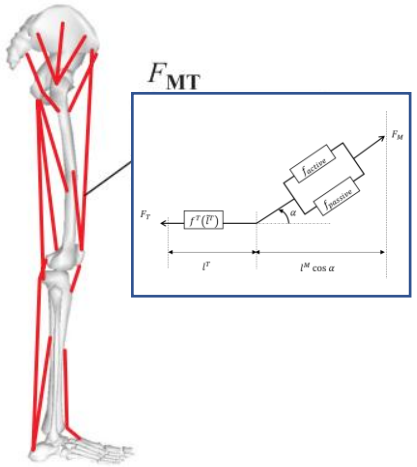
DT of motion of Human Musculo-Skeletal (MSK) system: Key Equations

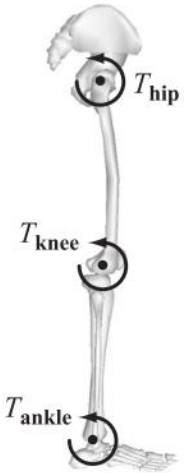
Muscle excitation $\{u(t)\}$

Muscle activation $\{a(t)\}$

Muscle & Tendon Force production $\{F_M, F_T\}$

Dynamic Human Motion $\{q(t), \dot{q}(t)\}$





MSK Dynamics:

$$\boldsymbol{M(q)\,\ddot{q} + C(q,\dot{q}) + G(q) - T_{MT} - E = 0}$$

where,

$\boldsymbol{q}$ → generalized coordinates

$\boldsymbol{\dot{q}}$ → generalized velocity

$\boldsymbol{\ddot{q}}$ → generalized acceleration

$\boldsymbol{E}$ → External force

$\boldsymbol{M(q)}$ → Inertial mass matrix

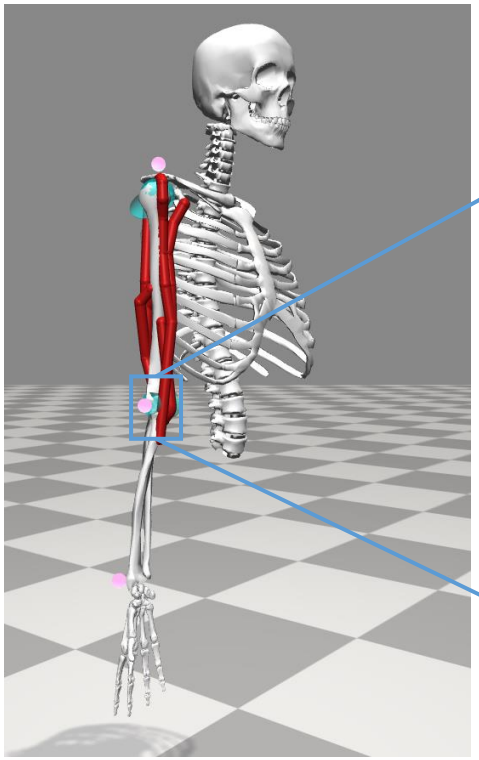
$\boldsymbol{C(q,\dot{q})}$ → Coriolis force

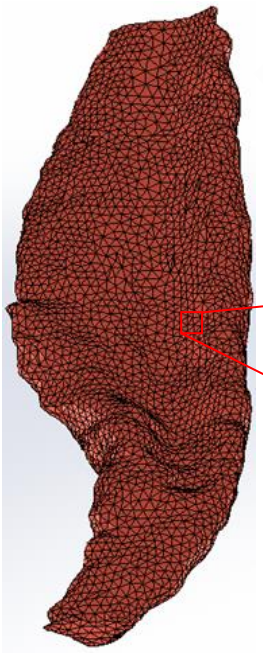
$\boldsymbol{G(q)}$ → Gravitational loading

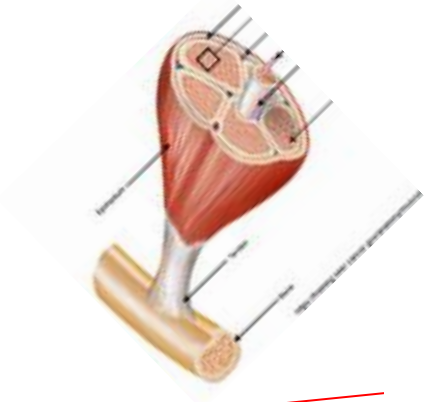
$\boldsymbol{T_{MT} = R(q)F_{MT}}$ → Joint torques

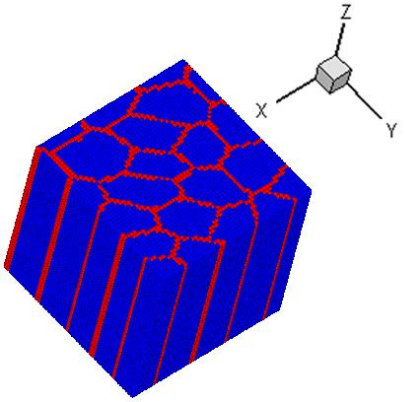
$\boldsymbol{F_{MT}}$ → Musculotendon force

$\boldsymbol{R(q)}$ → Moment arm matrix









Component Scale

Continuum Scale

Microscale

Dynamic Human Motion $q(t)$

Muscle Force Production F_{MT}

Muscle Activation $a(t)$

Muscle Excitation $u(t)$

Musculotendon Material Model

Active Material

Passive Material

Muscle Matrix Model

Fibre Passive Part

Connective Tissue

Solving Musculoskeletal Inverse Problem through Machine Learning

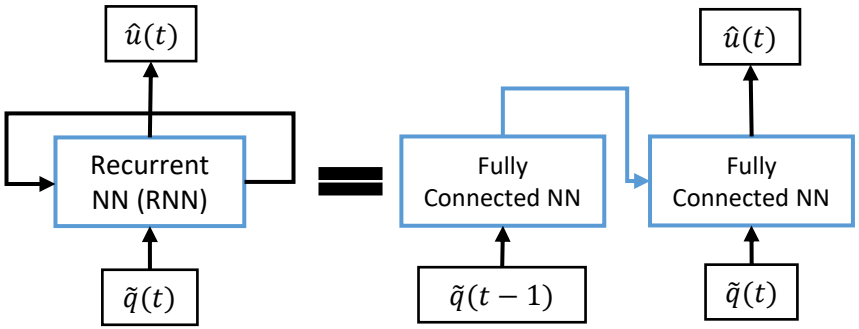
Musculoskeletal Forward Dynamics

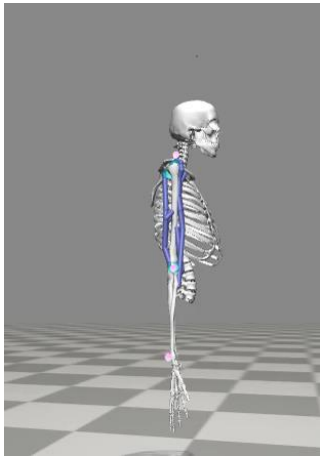
1. Muscle Excitation $u(t)$
2. Muscle Activation $a(t)$
3. Muscle Force production $F_{MT}(t)$
4. Dynamic Human Motion $q(t)$

$$\dot{a} = f(u, a) \quad (1)$$
$$\dot{F}_{MT} = f(F_{MT}, a, l^{MT}, v^{MT}) \quad (2)$$
$$M(q)\,\ddot{q} + C(q, \dot{q}) + R(q)F_{MT} = E \quad (3)$$

- $q(t)$ known from *motion capture* data
- unknown $u(t)$ leading to $q(t)$
- Inverse problem solved through *optimization*

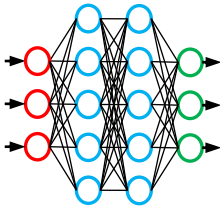
Approach 1: Recurrent Neural Networks (RNNs)



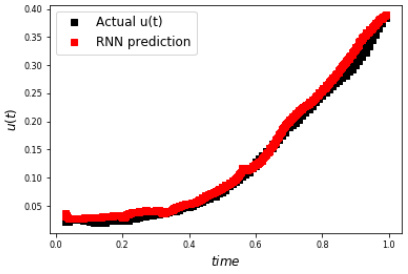


Training: Simulation[4] based data,
In: $\tilde{q}(t)$, Out: $\hat{u}(t)$

Usage: In: $q^{motion}(t)$, Out: $u^{motion}(t)$



RNN



Approach 2: Physics Informed NN (PINN)

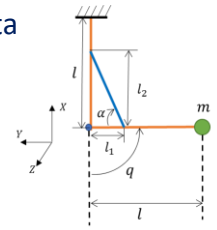
Identify f_0^M from simulated motion data

Given: $t, a(t), q(t)$ and ODE + IC

Unknown: NN parameters $\boldsymbol{\theta}$ &

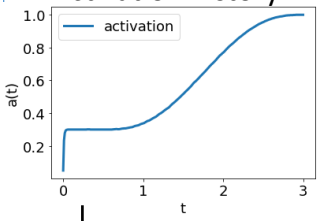
Max Isometric Muscle force f_0^M

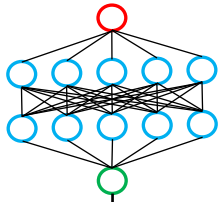
Goal: $\boldsymbol{\theta}^*, f_0^{M*} = \underset{\boldsymbol{\theta}, f_0^M}{\operatorname{argmin}} J(\boldsymbol{\theta}, f_0^M)$

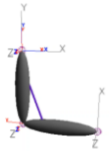


$I\ddot{q} = T(a, f_0^M, q)$

Activation history







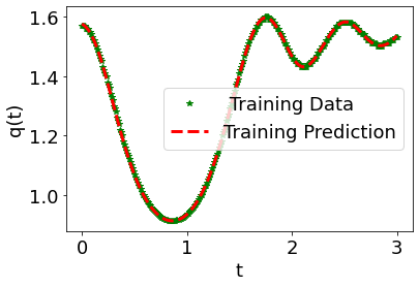
$\hat{q}(t;\boldsymbol{\theta})$

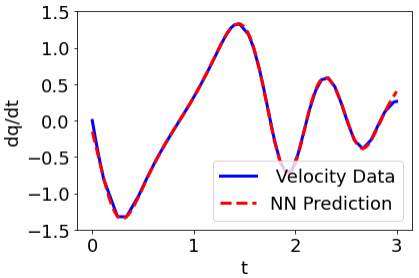
ODE Residual

$$R(\boldsymbol{\theta}, f_0^M) = I\ddot{\hat{q}}(t, \boldsymbol{\theta}) - T\left(a, f_0^M, \hat{q}(t;\boldsymbol{\theta})\right)$$

PINN Loss

$$J(\boldsymbol{\theta}, f_0^M) = \beta_1 \left|\left|q(t) - \hat{q}(t;\boldsymbol{\theta})\right|\right|^2 + \beta_2 \left|\left|R(\boldsymbol{\theta}, f_0^M)\right|\right|^2$$



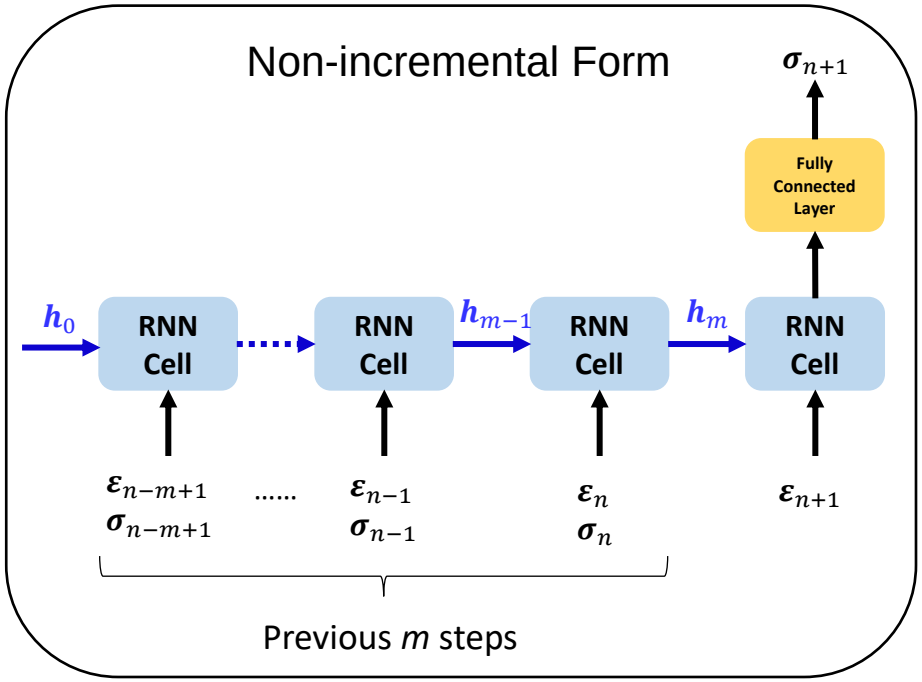


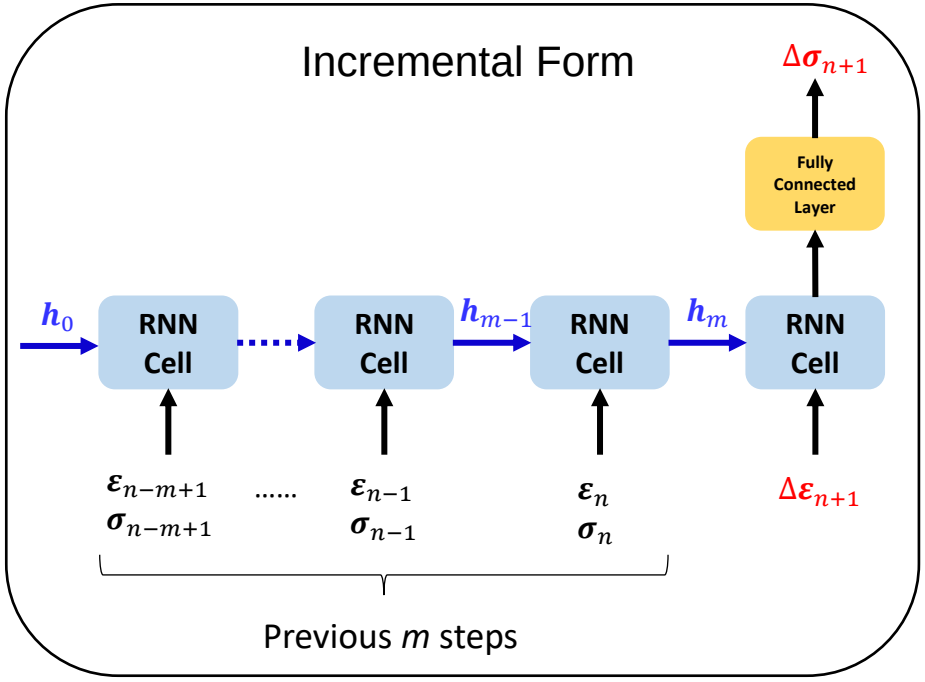
Karan Taneja, "Physics Informed System Identification of Digital Twins of Human Musculoskeletal System".

MS 3-1, Session 5, from 15:40 PM, Tuesday

Recurrent Neural Networks (RNN) for Path-Dependent Nonlinearity

- Non-Incremental Form vs. Incremental Form





$$\boldsymbol{\sigma}_{n+1} = RNN\{\boldsymbol{\varepsilon}_{n+1}, \boldsymbol{x}^h_{n+1}\}$$

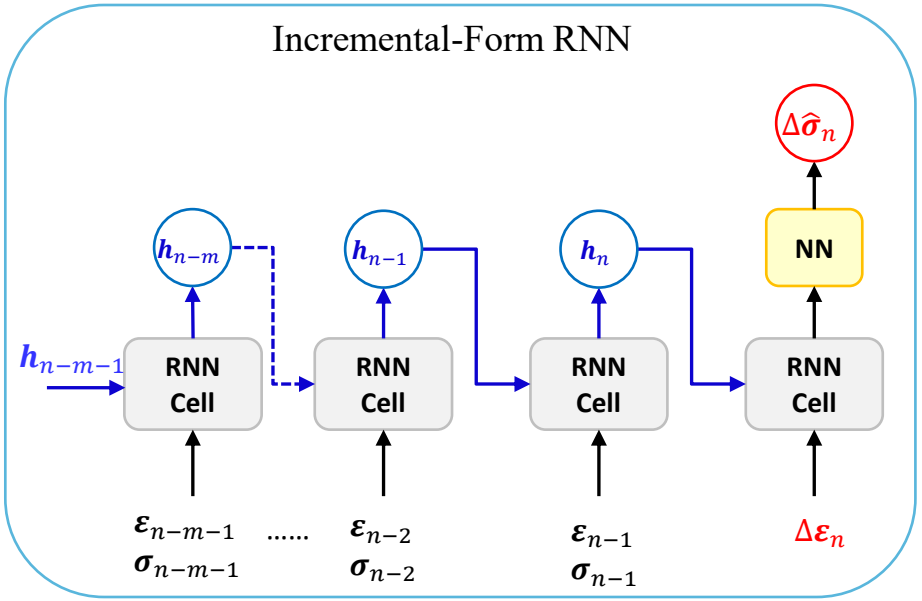
$$\downarrow$$

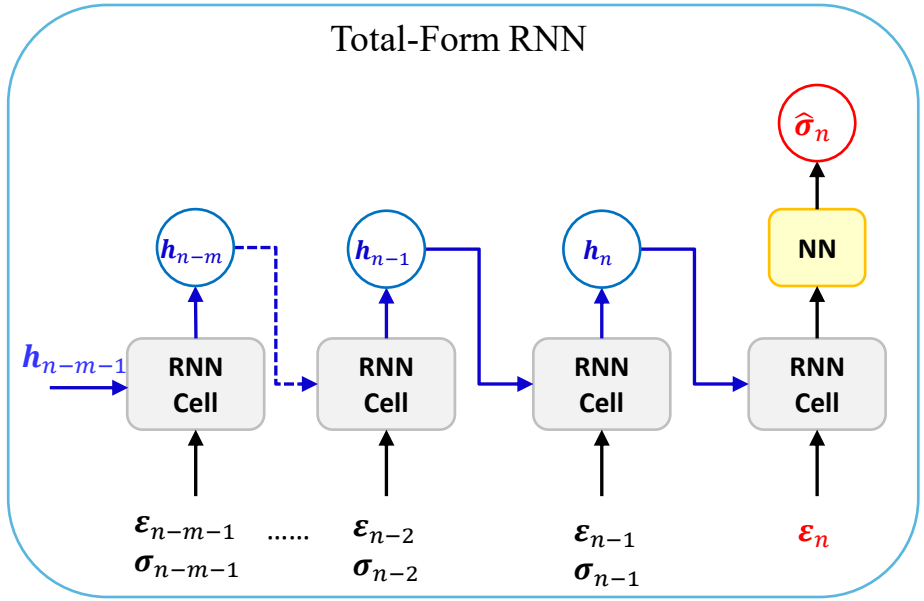
$$\boldsymbol{x}^h_{n+1} = [\boldsymbol{x}_n, \boldsymbol{x}_{n-1}, \dots] = [(\boldsymbol{\varepsilon}_n, \boldsymbol{\sigma}_n), (\boldsymbol{\varepsilon}_{n-1}, \boldsymbol{\sigma}_{n-1}), \dots]$$

$$\Delta\boldsymbol{\sigma}_{n+1} = RNN\{\Delta\boldsymbol{\varepsilon}_{n+1}, \boldsymbol{x}^h_{n+1}\}$$

$$\boldsymbol{\sigma}_{n+1} = \boldsymbol{\sigma}_n + \Delta\boldsymbol{\sigma}_{n+1}$$

Recurrent Neural Networks (RNNs): Incremental Form vs. Total Form



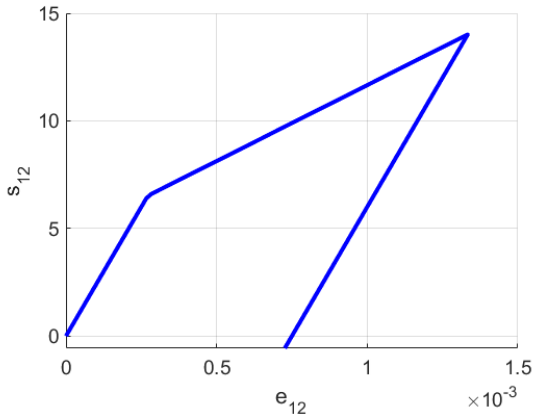


$$\Delta\hat{\boldsymbol{\sigma}}_n = RNN\{\Delta\boldsymbol{\varepsilon}_n, (\boldsymbol{\varepsilon}_{n-1}, \boldsymbol{\sigma}_{n-1}), (\boldsymbol{\varepsilon}_{n-2}, \boldsymbol{\sigma}_{n-2}), \dots\}$$

$$\hat{\boldsymbol{\sigma}}_n = RNN\{\boldsymbol{\varepsilon}_n, (\boldsymbol{\varepsilon}_{n-1}, \boldsymbol{\sigma}_{n-1}), (\boldsymbol{\varepsilon}_{n-2}, \boldsymbol{\sigma}_{n-2}), \dots\}$$

- Data generated by plane-strain J2 plastic model
  - Same loading path with 6 different strain-increment sizes
  - $E = 3 \times 10^4$
  - $E^{pl} = 1 \times 10^4$
  - $\nu = 0.25$
  - Linear isotropic hardening

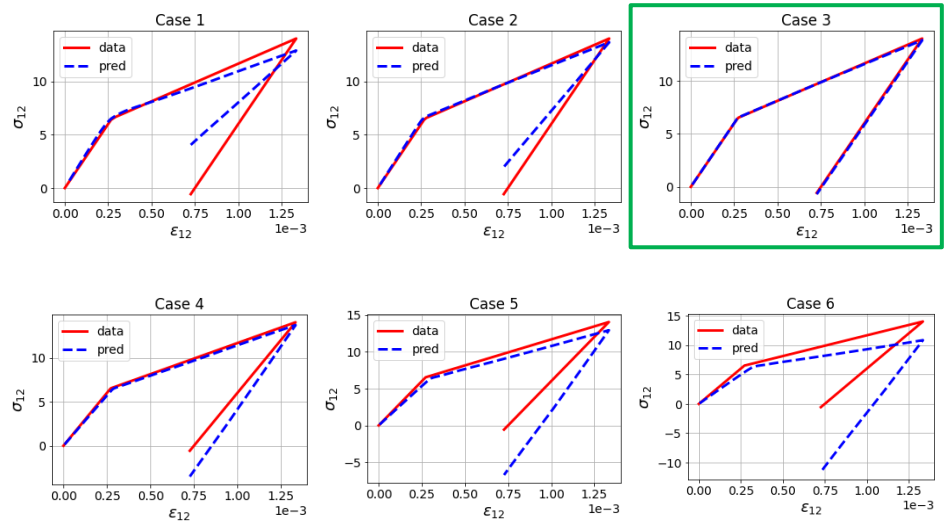
Case ID	1	2	3	4	5	6
# Increments	100	200	300	400	500	600



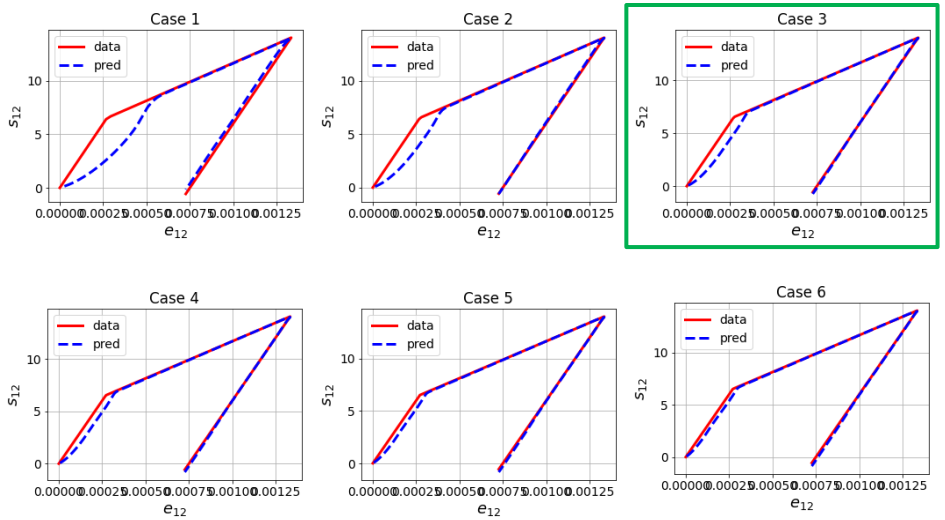
He, X., Chen, J. S., Physics-informed data-driven constitutive modeling for path-dependent materials, to be submitted

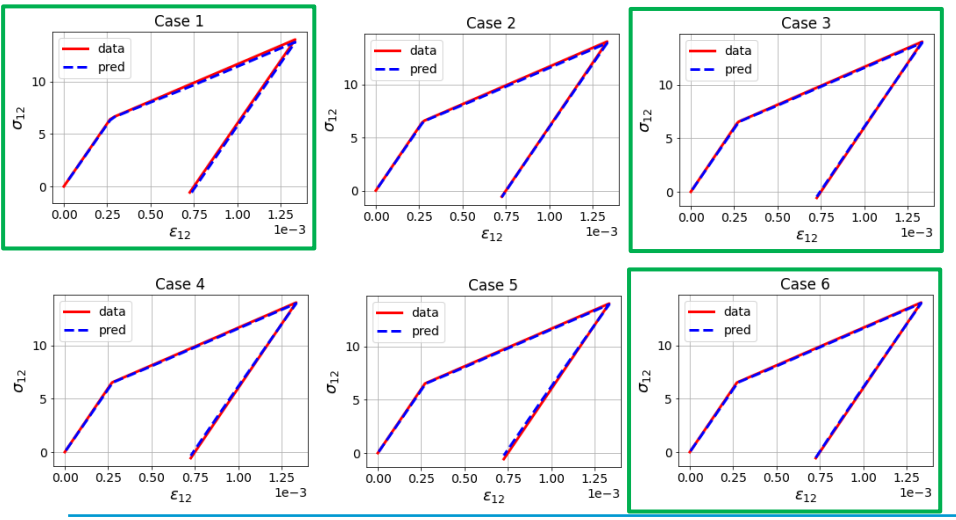
Recurrent Neural Networks (RNNs): Effects of Strain Increments

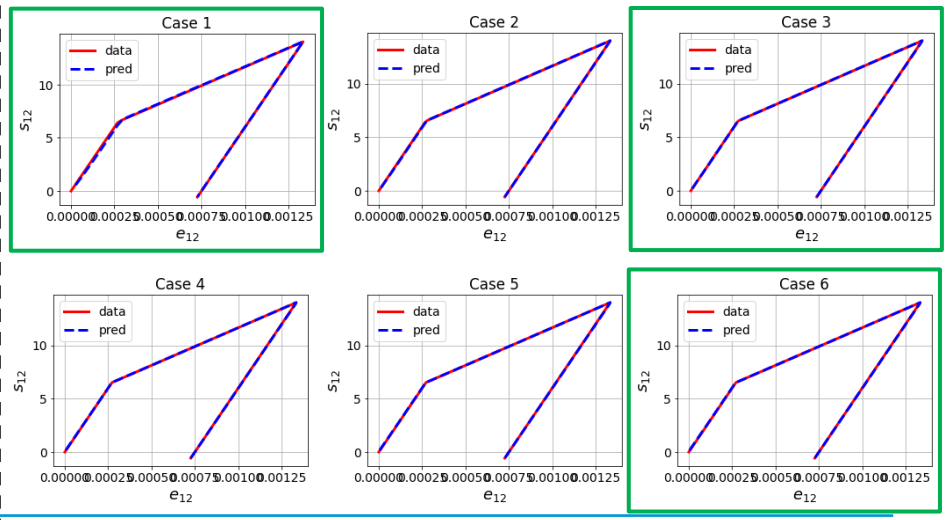
Incremental-Form RNN



Total-Form RNN







Recurrent Neural Networks (RNNs): Effects of Strain Increments

RNN	Train Case	Test Case	Total Loss
Incremental-Form	3	1,2,4,5,6	123.4
Incremental-Form	1,3,6	2,4,5	14.0
Total-Form	3	1,2,4,5,6	14.2
Total-Form	1,3,6	2,4,5	1.3

- Total-Form RNN is more robust against variations in strain increment sizes
- Training RNN with multiple increment sizes improves performance

He, X., Chen, J. S., Physics-informed data-driven constitutive modeling for path-dependent materials, to be submitted

Physics-Informed RNNs

- Strain energy convexity $\frac{1}{2}\dot{\boldsymbol{\sigma}}^T\dot{\boldsymbol{\varepsilon}} \geq 0$:

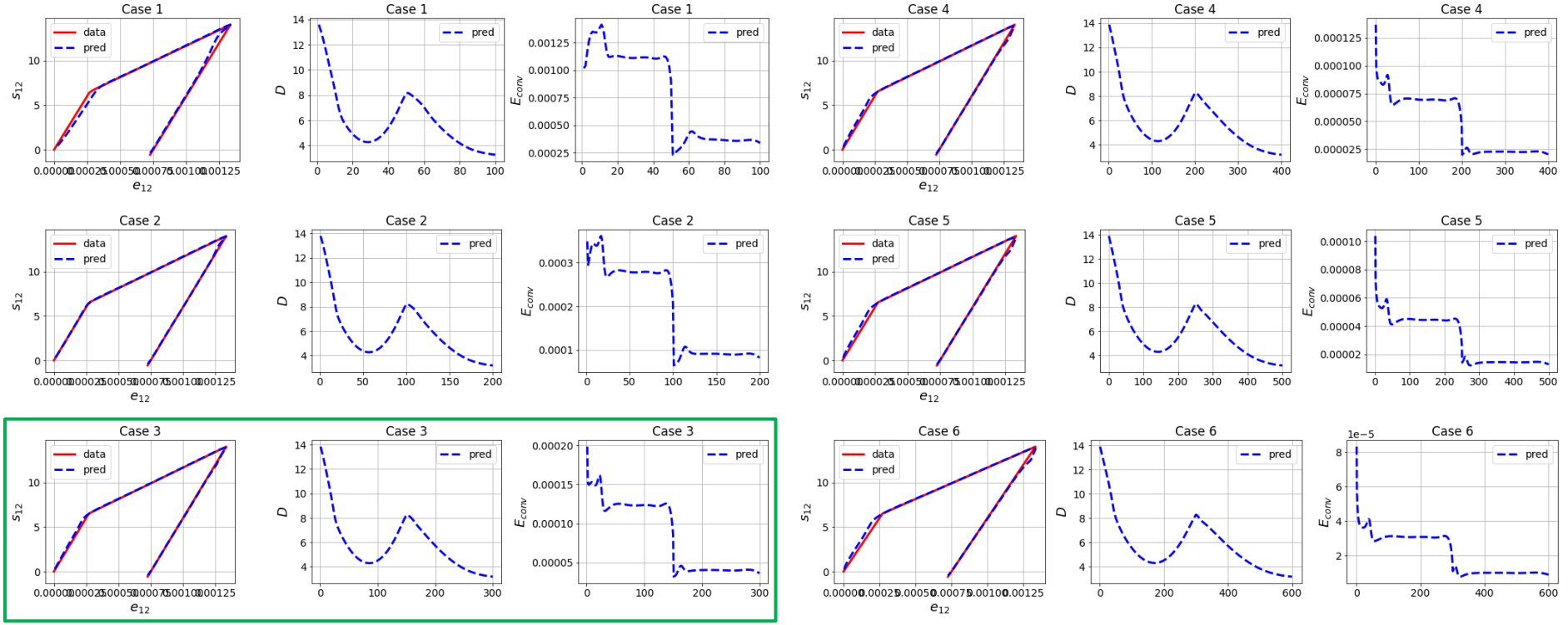
- Thermodynamics:
  - Include rate of mechanical dissipation to distinguish between reversible and irreversible processes

- Frame invariance (objectivity):

- Time consistency $\lim_{\Delta\boldsymbol{\varepsilon}\to\mathbf{0}} \Delta\boldsymbol{\sigma} = \mathbf{0}$:
  - Data augmented by pairs of zero strain-stress increments

He, X., Chen, J. S., Physics-informed data-driven constitutive modeling for path-dependent materials, to be submitted

Physics-Informed RNNs

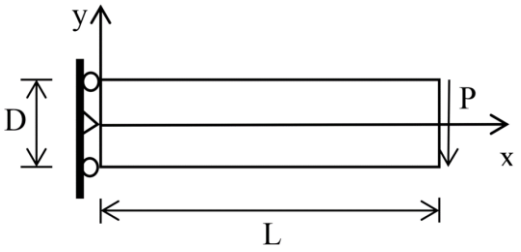


RNN	Train Case	Test Case	Test Loss
Incremental-Form	3	1,2,4,5,6	123.4
Total-Form	3	1,2,4,5,6	14.2
Physics-Informed	3	1,2,4,5,6	7.4

- Physics-Informed RNN more robust against variations in strain increment sizes

He, X., Chen, J. S., Physics-informed data-driven constitutive modeling for path-dependent materials, to be submitted

Inelastic Beam under Cyclic Shear Loadings



$L = 50$

$D = 10$

$P = 100$

$E = 3 \times 10^4$

$E^{pl} = 5 \times 10^3$

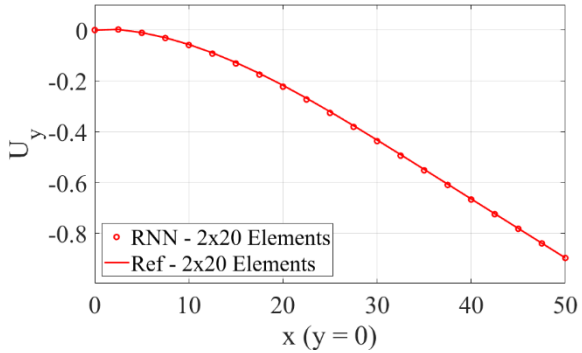
$\upsilon = 0.25$

$yield_0 = 50$

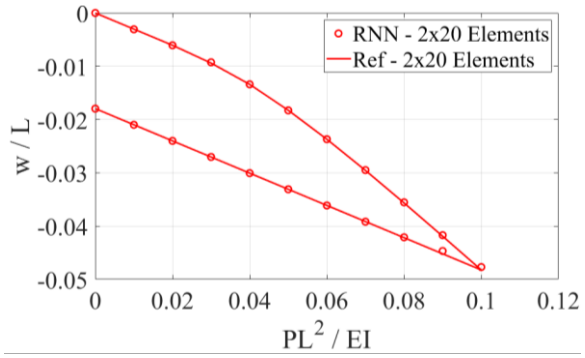
Xiaolong He, "Physics-Informed Convexity-Preserving Data-Driven Constitutive Modeling for Path-Dependent Materials"., MS 6-1, Session 2, 3:00 PM, Wednesday

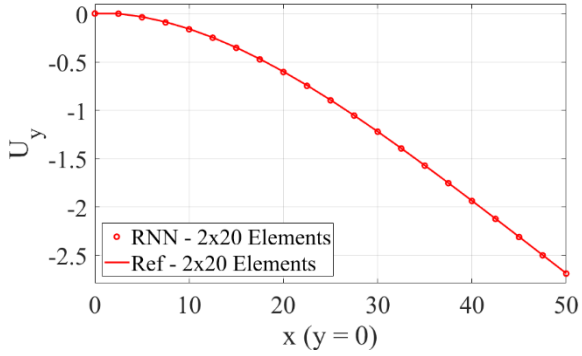
- Training data from J2 plastic material with linear isotropic hardening

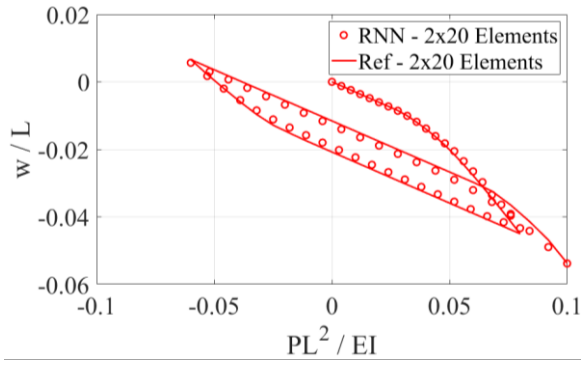
Deflection along y = 0



Normalized Tip Deflection







He, X., Chen, J. S., Physics-informed data-driven constitutive modeling for path-dependent materials, to be submitted

Conclusions

- In the last decade, we have witnessed the advancement of human-machine intelligence in unlocking many scientific similitude laws.
- Data science, powered by Machine Learning and Digital Twins, is at the heart of human-machine intelligence.
- More progress is direly needed for tackling scientific and engineering problems that are often complicated by the scarcity of data, the lack of understanding of the underlying physics, or a combination of both.
- Mechanistic Machine Learning and Digital Twins integrates available data with the fundamental principles of mathematical science through machine learning algorithms.
- It allows extraction of mechanistic features, and knowledge-driven dimension reduction to streamline the predictions.