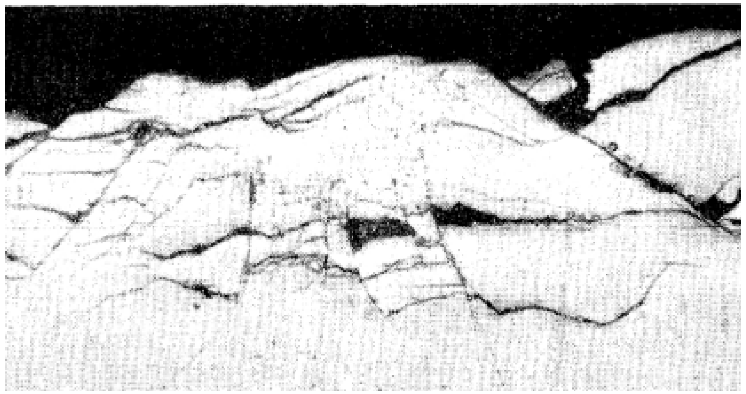
A Neural Network-Based Enrichment of Reproducing Kernel Approximation for Modeling Localization and Brittle Fracture

J. S. Chen, J. Baek, Y. Wang
Structural Engineering Department
Mechanical & Aerospace Engineering Department
Center for Extreme Events Research
University of California San Diego

Motivation

Discrete Representation of Localization and Fractures
Interface element insertion, embedded weak/strong discontinuities, enrichment

- Ineffective to determine curvilinear crack paths, crack kinking and branching
- Tedious and even impossible to track complex crack topologies.

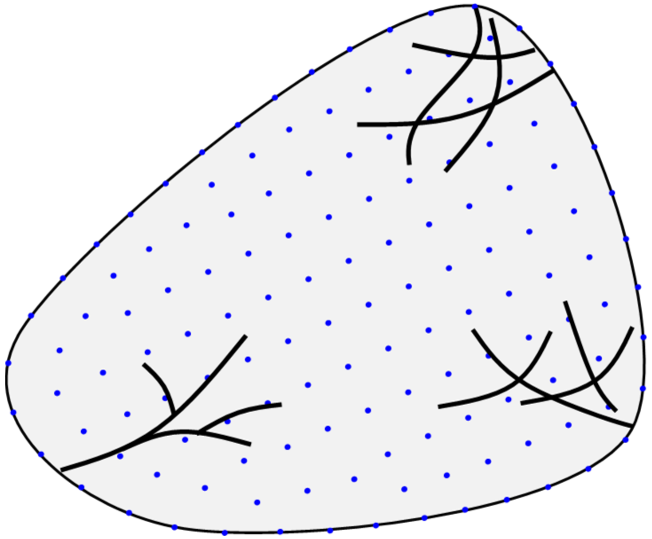


Diffuse Representation of Localization and Fractures
Quantity averaging, gradient methods, phase field

- Very fine discretization required.
- Adaptive model refinement is cumbersome for traditional mesh-based methods.

Proposed Neural Network Partition of Unity (NN-PU) Ritz Method

- PU (RKPM) + machine learned enrichment functions + energy minimization
- Loss function minimization: enrichment functions for localization and fracture
- Problems with local features: feature encoded transfer learning



NN-PU for Strain Localizations and Fractures

Solution decomposition

$$\mathbf{u}^h(\mathbf{x}) = \tilde{\mathbf{u}}^h(\mathbf{x}) + \hat{\mathbf{u}}^h(\mathbf{x})$$

- $\tilde{\mathbf{u}}^h$: smooth solution (**background**: fixed discretization)
- $\hat{\mathbf{u}}^h$: rough solution (**foreground**: evolving NN enrichment)

Smooth solution approximation by partition of unity (PU)

$$\tilde{\mathbf{u}}^h(\mathbf{x}) \approx \mathbf{u}^{RK}(\mathbf{x}) = \sum_{I=1}^{NP} \Psi_I(\mathbf{x})\mathbf{d}_I$$

Rough solution approximation (for strong/weak discontinuities)

$$\hat{\mathbf{u}}^h(\mathbf{x}) \approx \mathbf{u}^{NN}(\mathbf{x}) = \sum_{J=1}^{NB} \mathbf{u}_J^B(\mathbf{x})$$

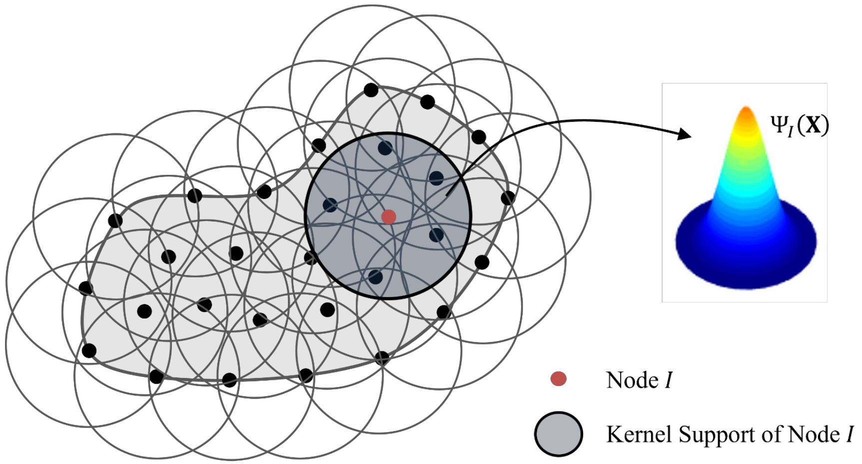
- $\mathbf{u}_J^B$: block-level NN approximation

$$\mathbf{u}_J^B(\mathbf{x}) = \sum_{K=1}^{n_{NK}} \hat{\phi}_{JK}(\mathbf{x})\hat{\boldsymbol{v}}_{JK}(\mathbf{x})$$

$$\hat{\boldsymbol{v}}_{JK}(\mathbf{X}) = \sum_{I\in\bar{\mathcal{S}}} \Psi_I(\mathbf{x})\hat{w}_{IJK}^C$$

$$\mathbf{u}^h = \tilde{\mathbf{u}}^h + \hat{\mathbf{u}}^h = \sum_{I\in\mathcal{S}} \Psi_I(\mathbf{x})\left(\mathbf{d}_I + \sum_{K=1}^{n_{NK}} \zeta_K(\mathbf{x})w_{IK}^C\right)$$

$$w_{IK}^C = 0 \;\forall I \notin \bar{\mathcal{S}} = \{J \mid \exists\mathbf{x} \in supp(\Psi_J), \kappa(\mathbf{x}) > \kappa_c\}$$



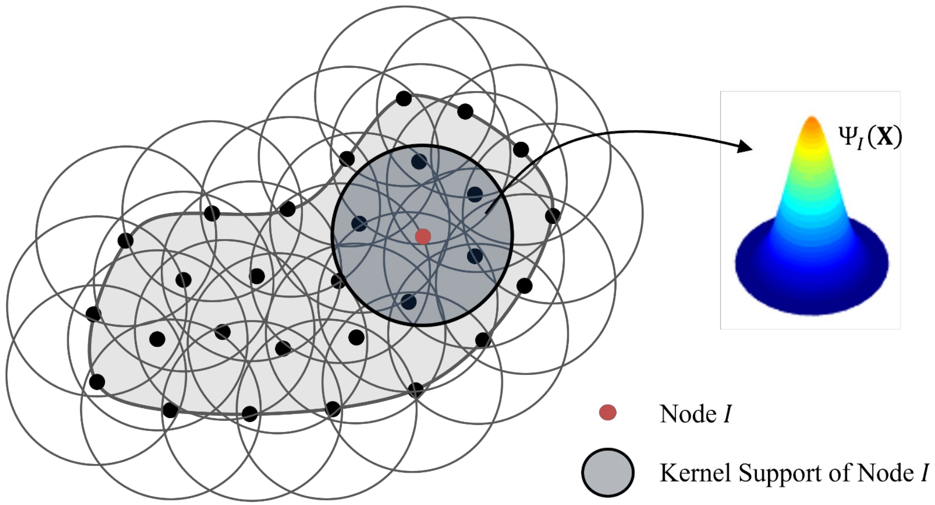
- $\mathbf{u}^{RK}$: Reproducing Kernel (RK) approximation
- $\mathbf{u}^{NN}$: Neural Network (NN) approximation
- $\mathbf{u}_J^B$: block-level NN approximation

Baek, J., Chen, J. S., Susuki, K, International Journal for Numerical Methods in Engineering. Vol. 123, pp 4422-4454, 2022.
Baek, J., Chen, J. S., Computer Methods in Applied Mechanics and Engineering, arXiv:2307.01937, 2023

Reproducing Kernel Particle Method

Reproducing kernel (RK) approximation

- The order of continuity and the order of completeness are independently defined in the RK approximation.



RK approximation

$$u^h(\mathbf{X}) = \sum_{I=1}^{NP} \Psi_I(\mathbf{X}) d_I$$

RK shape function

Order of completeness — Order of continuity

$$\Psi_I(\mathbf{X}) = \left\{ \sum_{|\boldsymbol{\alpha}| \le n} (\mathbf{X} - \mathbf{X}_I)^{\boldsymbol{\alpha}} b_{\boldsymbol{\alpha}}(\mathbf{x}) \right\} \Phi_a(\mathbf{X} - \mathbf{X}_I)$$

Reproducing condition

$$\sum_{J=1}^{NP} \Psi_I(\mathbf{X}) \mathbf{X}_I^{\boldsymbol{\alpha}} = \mathbf{X}^{\boldsymbol{\alpha}}, \qquad |\boldsymbol{\alpha}| \le n$$

$$\Longrightarrow \quad \Psi_I(\mathbf{x}) = \mathbf{H}^T(0)\mathbf{M}^{-1}(\mathbf{x})\mathbf{H}(\mathbf{x} - \mathbf{x}_I)\phi_a(\mathbf{x} - \mathbf{x}_I)$$

$$\mathbf{M}(\mathbf{x}) = \sum_{I=1}^{NP} \mathbf{H}(\mathbf{x} - \mathbf{x}_I)\mathbf{H}^T(\mathbf{x} - \mathbf{x}_I)\phi_a(\mathbf{x} - \mathbf{x}_I)$$

$$\mathbf{H}(\mathbf{x} - \mathbf{x}_I) = \left[1, (x - x_I), \cdots, (x - x_I)^n\right]$$

- Straightforward adaptive refinement
- Arbitrary order continuities
- Enrichment with special functions

Liu WK, Jun S, Zhang YF. *Int J Numer Methods Fluids* 1995;20:1081–106.
Chen JS, Pan C, Wu C-T, Liu WK. *Comput Methods Appl Mech Engrg* 1996;139:195–227.

Block-level NN Approximation

Neural network (NN) approximation

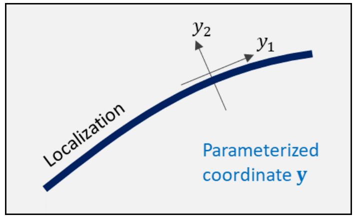
$$u^{NN}(\mathbf{x}) = \sum_{B=1}^{N_B} b_B^{NN}(\mathbf{x};\mathbf{W}_B)$$

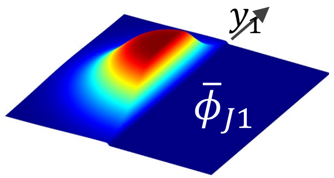
- b_B^{NN}: block-level NN approximation

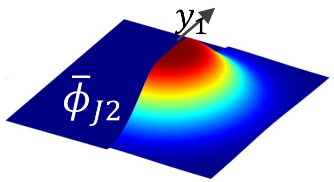
Block-level NN approximation

$$b_B^{NN}(\mathbf{x};\mathbf{W}) = \sum_{K=1}^{N_K} \hat{\phi}_{KB}\left(\mathbf{y}(\mathbf{x};\mathbf{W}_B^L),\mathbf{W}_{KB}^S\right) p(\mathbf{x};\mathbf{W}_{KB}^P)$$

- NK: the number of NN kernels per block







NN Kernel function captures
- Location and orientation of localization
- Shape of solution transition

NN Polynomial introduces
- Monomial completeness for further accuracy

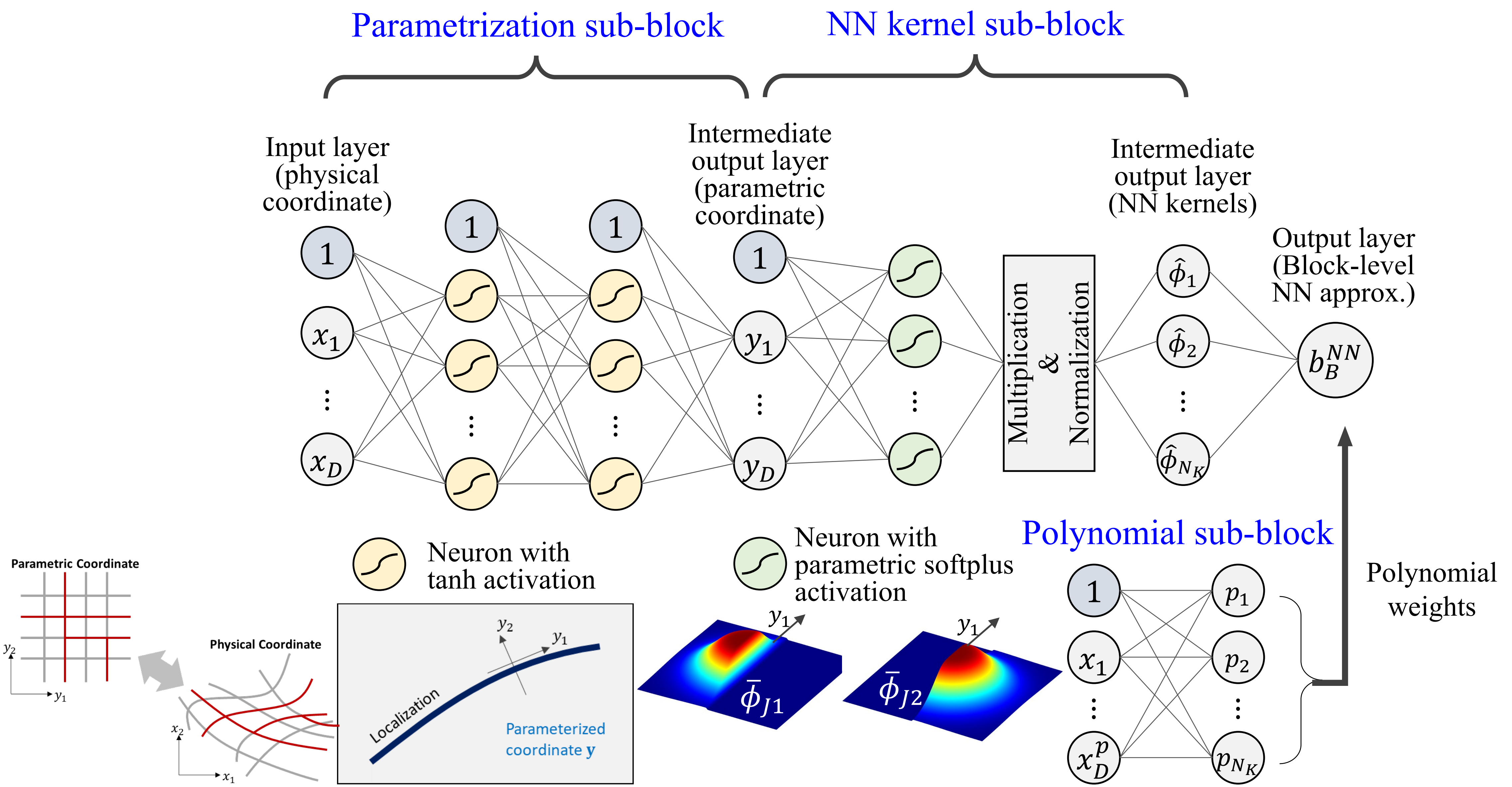
- $\mathbf{W}^L$: NN weight set controlling the location and orientation of the kernel.
- $\mathbf{W}^S$: NN weight set controlling the shape of transition.

- $\mathbf{W}^P$: NN monomial coefficient set

* The NN control parameters $\mathbf{W}^L$, $\mathbf{W}^S$, and $\mathbf{W}^P$ are determined via loss function minimization.

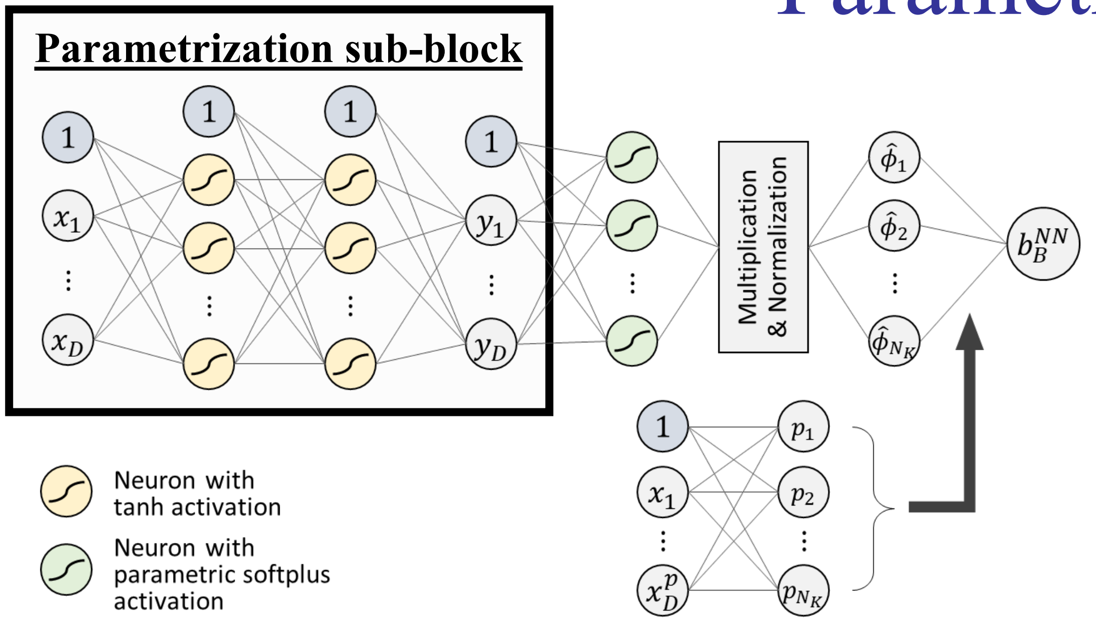
Baek, J., Chen, J. S., & Susuki, K. (2022). A Neural Network-enhanced Reproducing Kernel Particle Method for Modeling Strain Localization. *International Journal for Numerical Methods in Engineering*, Vol. 123, pp 4422-4454.

A Neural Network-enhanced RKPM for Localization Modeling

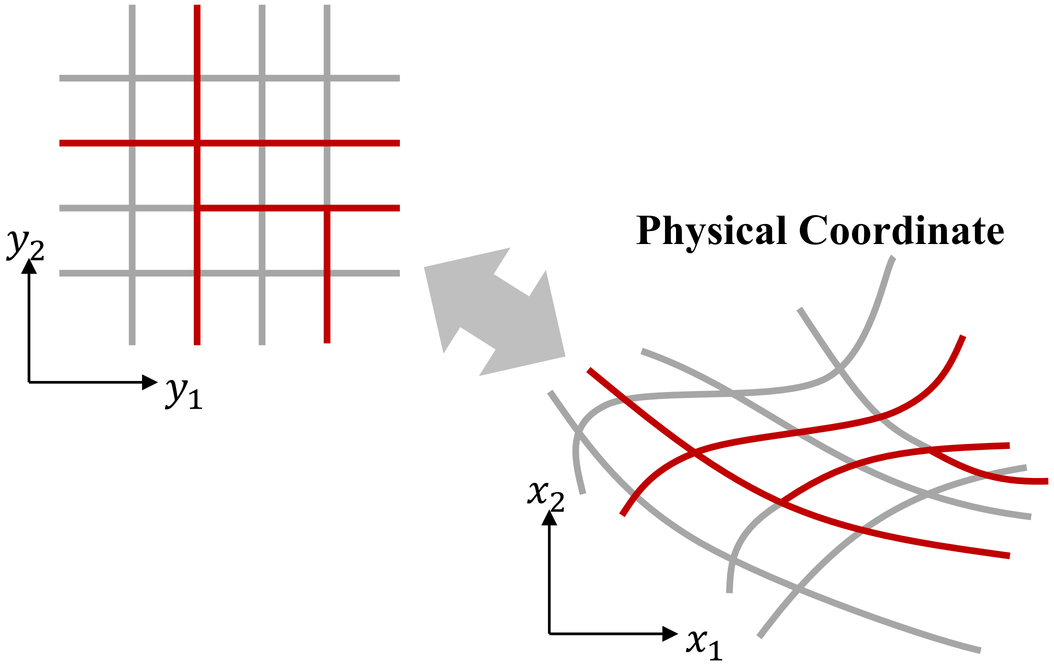


Baek, J., Chen, J. S., Susuki, K, International Journal for Numerical Methods in Engineering. Vol. 123, pp 4422-4454, 2022.
Baek, J., Chen, J. S., Computer Methods in Applied Mechanics and Engineering, arXiv:2307.01937, 2023

Parametrization by $\mathbf{W}^L$



Parametric Coordinate



- Complicated localization patterns are ***projected onto a parametric space***.
- ***NN kernel functions*** defined in the parametric space can capture complex localizations in the physical space.
- **NN Parametric Coordinate**

For NN block B, $\mathcal{P}^B: \mathbf{x} \to \mathbf{y}(\mathbf{x}; \mathbf{W}_B^L)$

$$\mathbf{y}(\mathbf{X}; \mathbf{W}_B^L) = \mathbf{h}_{n_L}(\cdot; \mathbf{W}_{Bn_L}^L) \circ \mathbf{h}_{n_L-1}(\cdot; \mathbf{W}_{Bn_L-1}^L) \circ \cdots \circ \mathbf{h}_1(\mathbf{x}; \mathbf{W}_{B1}^L)$$

where $\mathbf{h}_i(\boldsymbol{\xi}; \mathbf{W}_{Bi}^L) = \tanh\left(\mathbf{z}_i(\boldsymbol{\xi}; \mathbf{W}_{Bi}^L)\right)$, for $i < n_L$

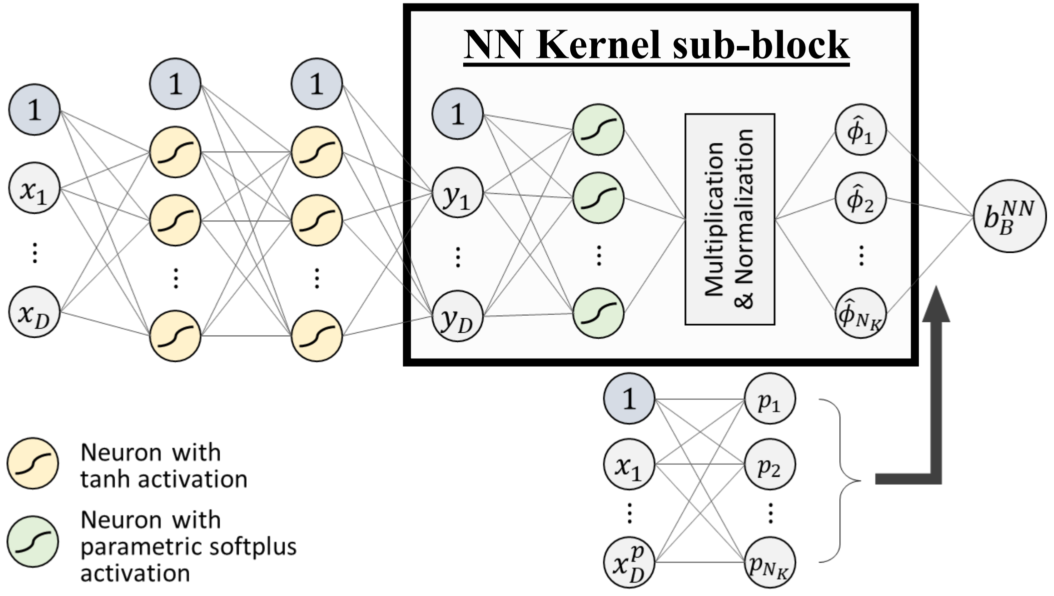
$$\mathbf{h}_i(\boldsymbol{\xi}; \mathbf{W}_{Bi}^L) = \mathbf{z}_i(\boldsymbol{\xi}; \mathbf{W}_{Bi}^L), \text{ for } i = n_L$$

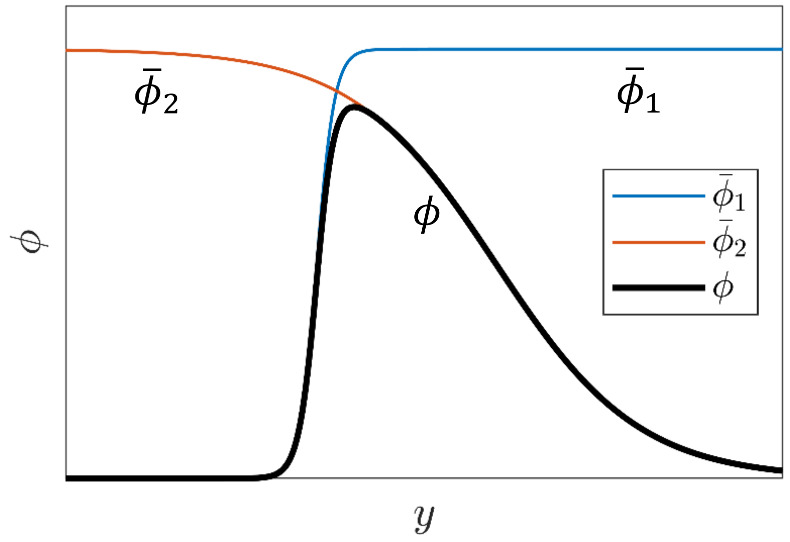
$$\mathbf{z}_i = \boldsymbol{\Theta}^{Bi}\boldsymbol{\xi} + \boldsymbol{\beta}^{Bi}$$

$$\mathbf{W}_{Bi}^L = \{\boldsymbol{\Theta}^{Bi}, \boldsymbol{\beta}^{Ii}\} \subset \mathbf{W}_B^L$$

with the weight matrix $\boldsymbol{\Theta}^{Bi}$ and the bias vector $\boldsymbol{\beta}^{Bi}$

NN Kernel Function Controlled by $\mathbf{W}^S$





NN Kernel Function

$$\phi(y; \mathbf{W}_{KB}^S) = \prod_{i=1}^{2} \underbrace{\bar{\phi}\left(z_i\left(y, \bar{y}_i^{KB}, c_i^{KB}\right); \beta_i^{KB}\right)}_{\text{Regularized step functions}}$$

Regularized Step Functions

$$\bar{\phi}(z_i; \beta_i) \equiv S\left(z_i + \frac{1}{2}; \beta_i\right) - S\left(z_i - \frac{1}{2}; \beta_i\right)$$

where $z_i = (-1)^i (y - \bar{y}_i)/c_i$, $i = 1, 2$

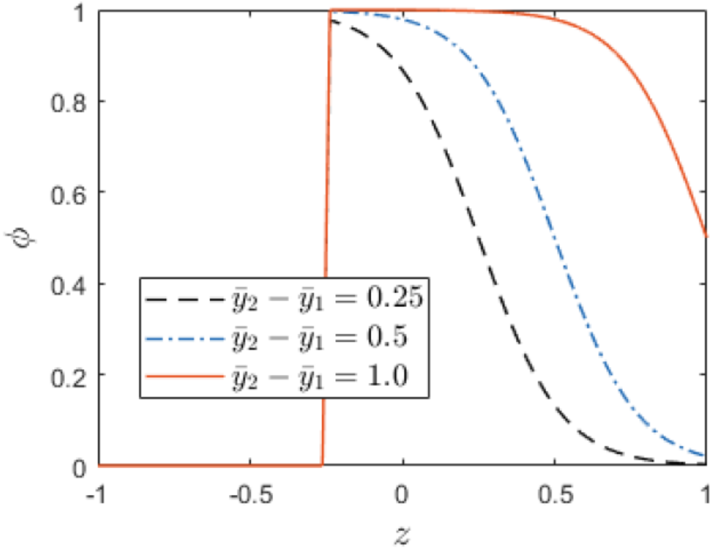
$$S(z; \beta) = \frac{1}{\beta}\log\left(1 + e^{\beta z}\right)$$

(parametric softplus function)

Neural Network Kernel Function Controlled by $\mathbf{W}^S$

NN Control Parameter $\bar{y}$

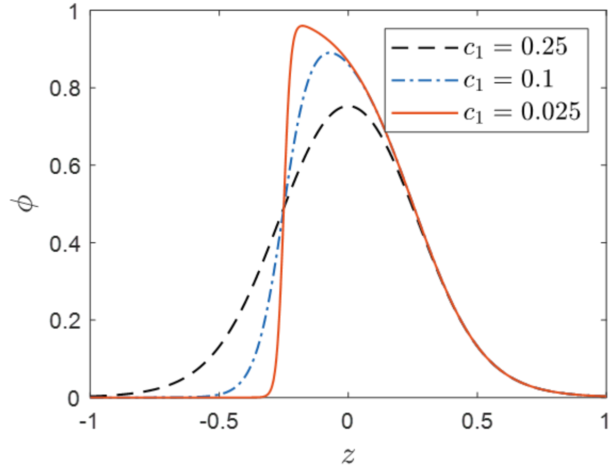
Domain of influence



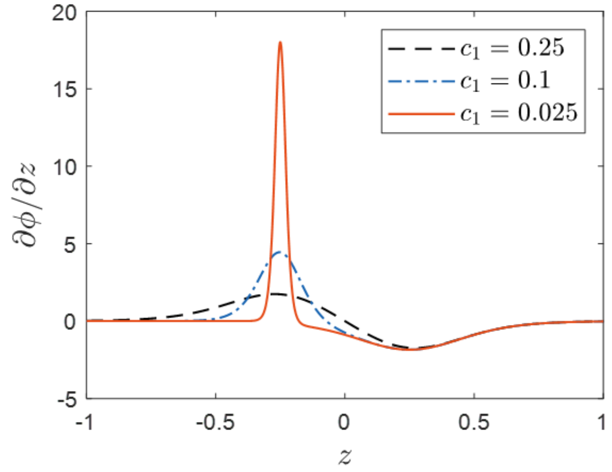
NN Control Parameter c

Transition of NN kernel function

NN kernel function

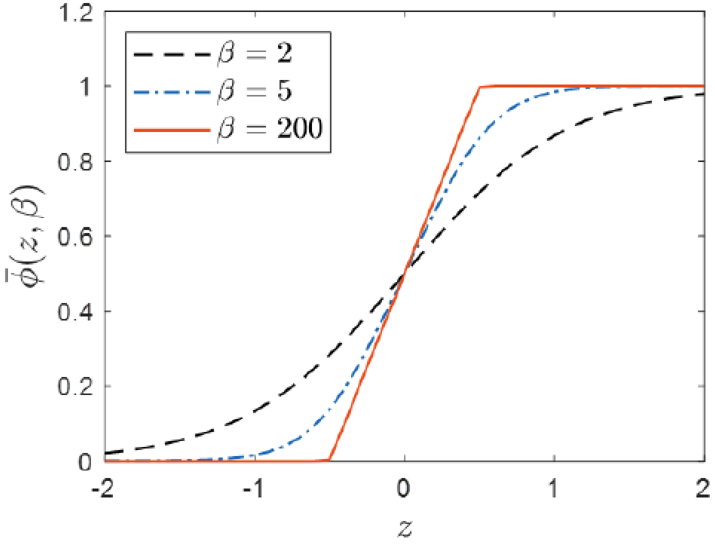


NN kernel function derivatives

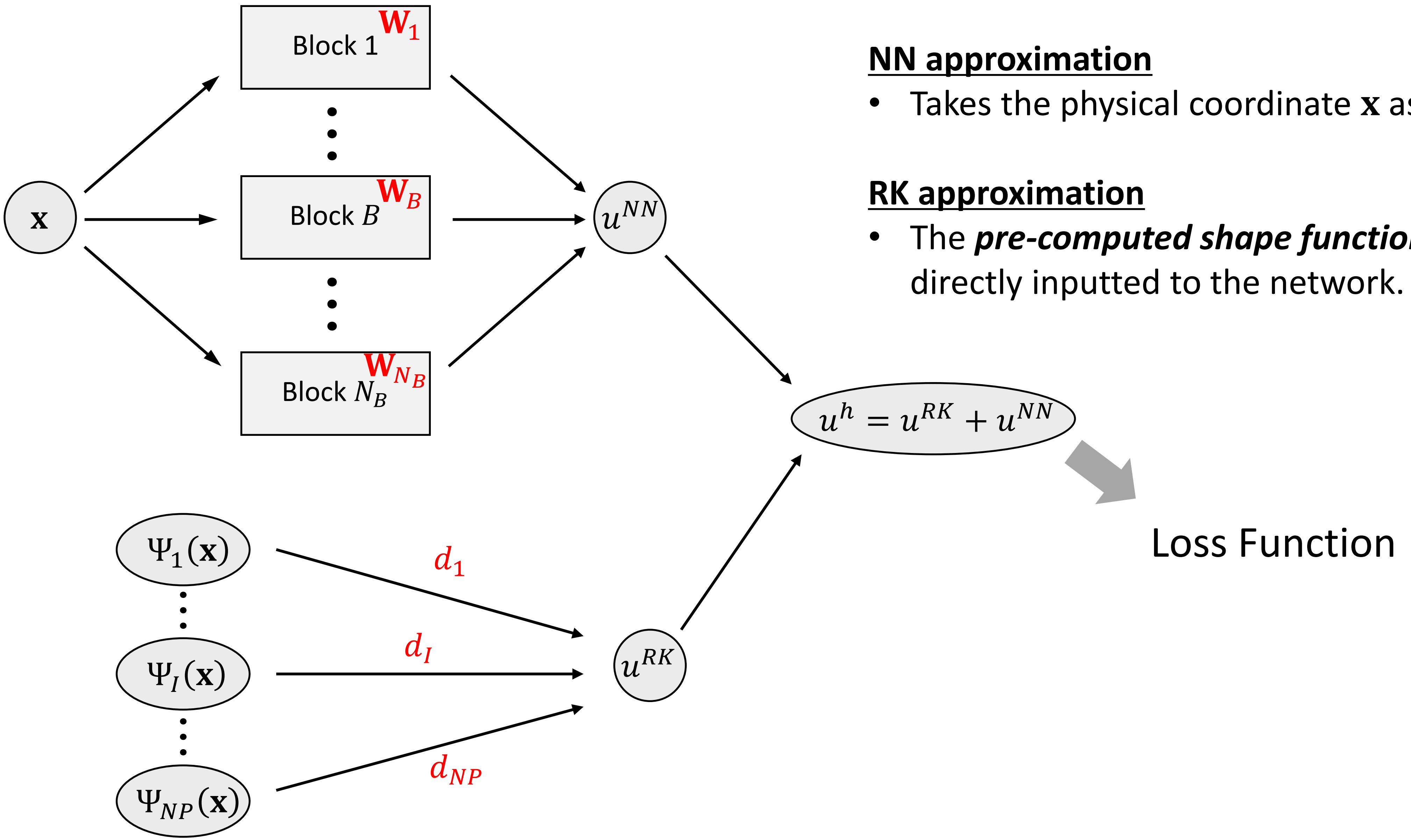


NN Control Parameter β

Transition of NN kernel function derivative



NN-enhanced RK Network Structure



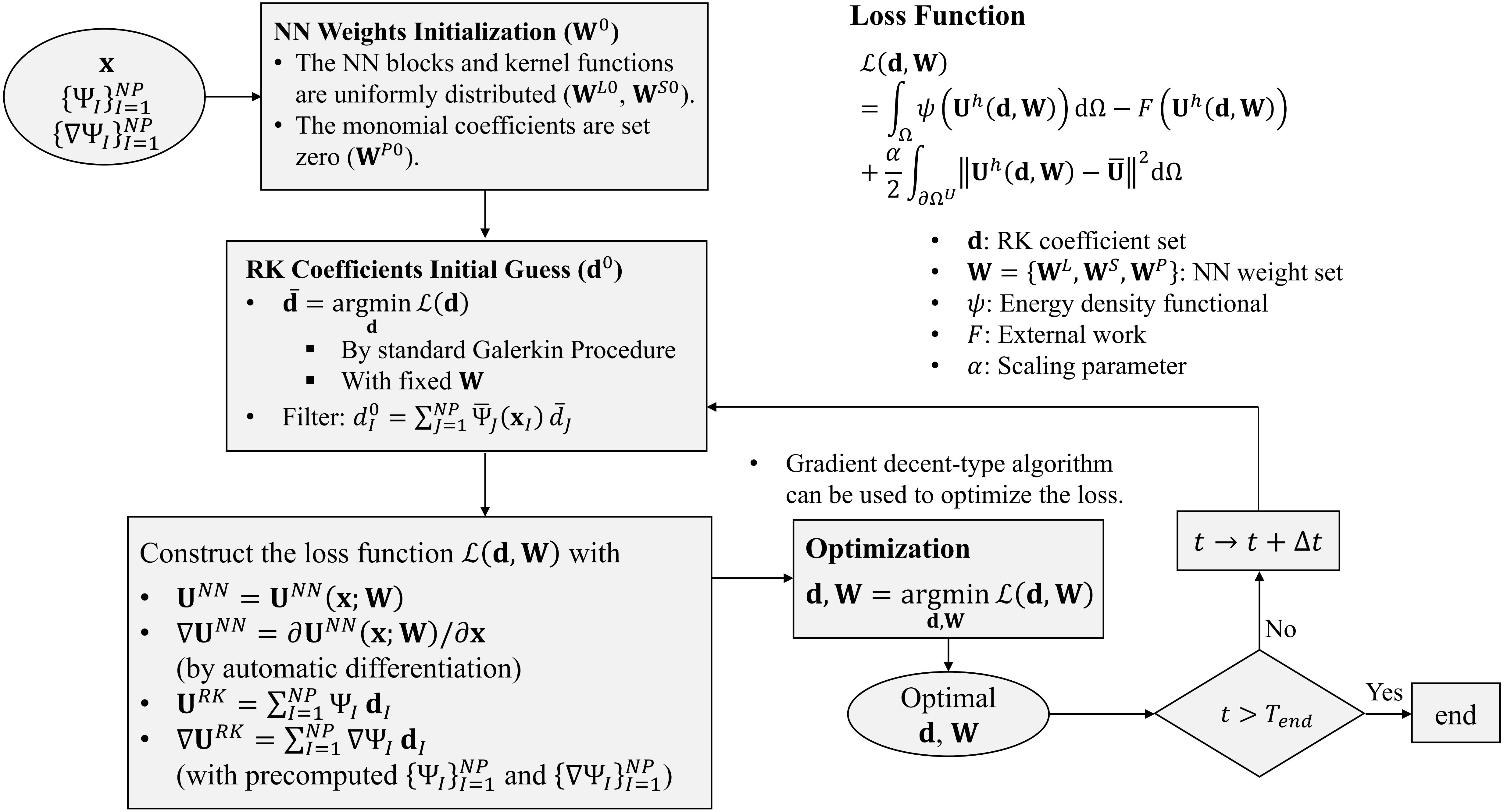
NN approximation

- Takes the physical coordinate $\mathbf{x}$ as its input.

RK approximation

- The ***pre-computed shape functions*** are directly inputted to the network.

Neural Network Enriched Partition of Unity by Ritz Method



Loss Function for Localization

Loss Function

$$\mathcal{L}(\mathbf{d},\mathbf{W}) = \int_{\Omega} \psi^{D}\left(\mathbf{U}^{h}(\mathbf{d},\mathbf{W})\right) \mathrm{d}\Omega - F\left(\mathbf{U}^{h}(\mathbf{d},\mathbf{W})\right) + \frac{\alpha}{2}\int_{\partial\Omega^{U}} \left\|\mathbf{U}^{h}(\mathbf{d},\mathbf{W}) - \bar{\mathbf{U}}\right\|^{2} \mathrm{d}\Omega$$

$$F(\mathbf{u}) = \int_{\Omega} \mathbf{u}\cdot\mathbf{b}\, d\Omega + \int_{\partial\Omega_{\mathrm{h}}} \mathbf{u}\cdot\mathbf{h}\, d\Gamma$$

$$\psi^{D}(\mathbf{u}) = g\left(\eta\left(\boldsymbol{\varepsilon}(\mathbf{u})\right)\right)\psi_0^{+}(\mathbf{u}) + \psi_0^{-}(\mathbf{u}) + \bar{\psi}\left(\eta\left(\boldsymbol{\varepsilon}(\mathbf{u})\right)\right)$$ [1]

$$\psi_0^{+} = \mu\langle\bar{\varepsilon}_i\rangle\langle\bar{\varepsilon}_i\rangle + \frac{\lambda}{2}\langle\textstyle\sum\bar{\varepsilon}_i\rangle^{2},\ \psi_0^{-} = \psi^{el} - \psi_0^{+}$$

$$\boldsymbol{\sigma} = \left(1-\eta\left(\boldsymbol{\varepsilon}(\mathbf{u})\right)\right)\frac{\partial\psi_0^{+}(\mathbf{u})}{\partial\boldsymbol{\varepsilon}(\mathbf{u})} + \frac{\partial\psi_0^{-}(\mathbf{u})}{\partial\boldsymbol{\varepsilon}(\mathbf{u})}$$

Dissipation Energy

$$\bar{\psi}(\eta) = p\left(\frac{1}{q-\eta} - \frac{1}{q}\right)$$

$$\eta(\kappa) = \min\left(1, \max\left(0, \bar{\eta}(\kappa)\right)\right), \qquad \bar{\eta}(\kappa) = \frac{1-\kappa_0/\kappa}{1-\kappa_0/\kappa_c}, \qquad g(\eta)=1-\eta$$

$$p = \frac{E}{2}(\kappa_0 q)^2, \qquad q = \frac{\kappa_c}{\kappa_c - \kappa_0}.$$

[1] Miehe C, Hofacker M, Welschinger F. Comput Methods Appl Mech Eng. 2010;199(45):2765-2778

Regularization of NN Approximation

In order to have ***discretization-insensitive*** NN-RK approximation, the original NN kernel function is modified.

Original NN kernel function

$$\phi = \prod_{\alpha=1}^{dim} \prod_{i=1}^{2} \bar{\phi}(z_{\alpha i}(y_\alpha, \bar{y}_{\alpha i}, c_{\alpha i}); \beta_{\alpha i})$$

$$z_{\alpha i} = (-1)^i \frac{(y_\alpha(\mathbf{x}; \mathbf{W}^L) - \bar{y}_i)}{c_i}$$

Modified NN kernel function

$$\phi = \prod_{\alpha=1}^{dim} \prod_{i=1}^{2} \bar{\phi}(\hat{z}_{\alpha i}(y_\alpha, \bar{y}_{\alpha i}, c_{\alpha i}); \beta_{\alpha i})$$

$$\hat{z}_{\alpha i} = (-1)^i \frac{\hat{H}(y_\alpha(\mathbf{x}; \mathbf{W}^L) - \bar{y}_i)}{c_i}$$

where $\hat{H} = \frac{1}{\|\partial y_\alpha / \partial \mathbf{x}\|_{y=\bar{y}_i}}$

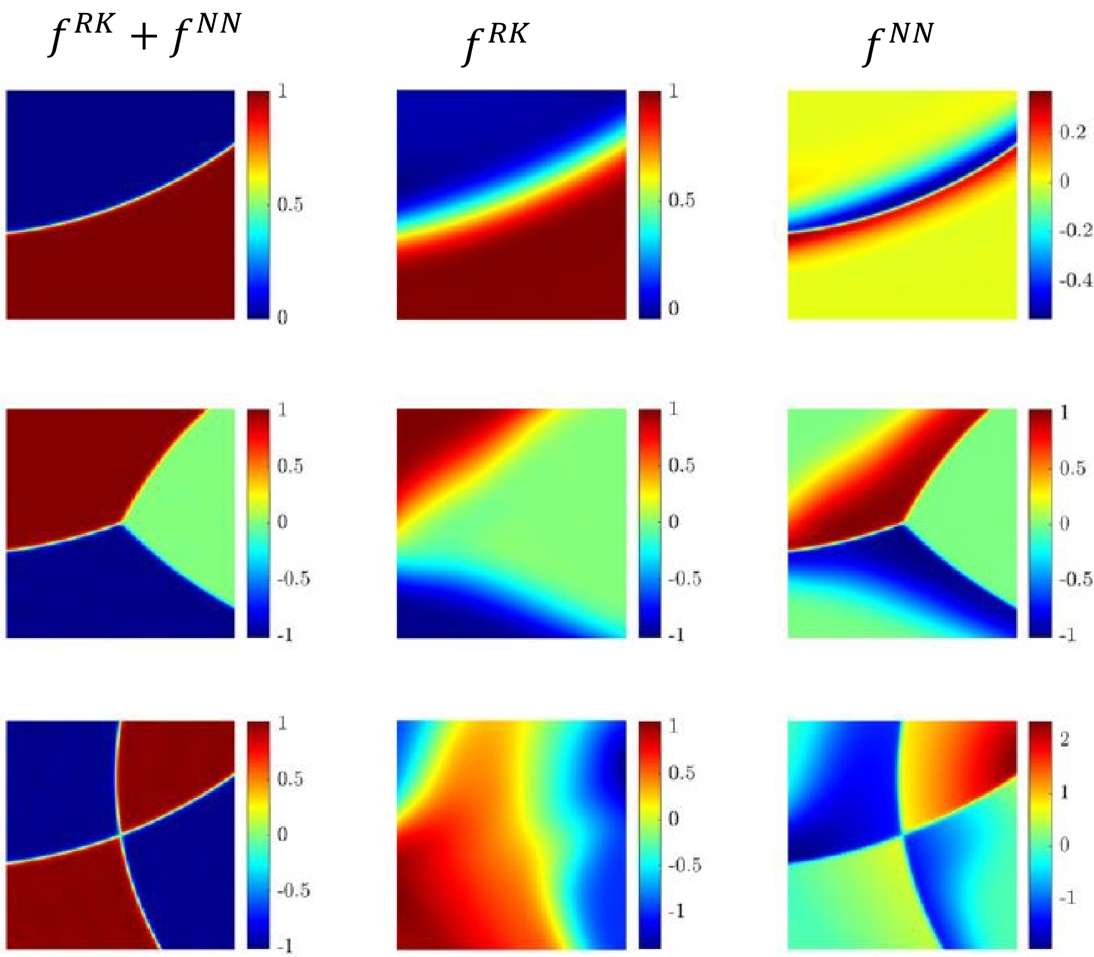
The parametric coordinate is properly scaled so that the localization width is solely determined by the NN control parameter c.

Approximation Ability of a Single NN Block

<u>A 4-kernel NN block</u> successfully captures the very high gradient in localizations for ***topological geometries*** of

- A curve without junctions
- 1 triple junction
- 1 quadruple junction

Higher order topological geometries can by captured by the <u>superposition of multiple block-level NN approximations</u>.



Error Estimate

- Ω: problem domain
- $\widehat{\Omega}$: domain within localization

Convergence rates for $\boldsymbol{\beta} \rightarrow \infty$

$$\left\|u^h - u\right\|_{0,\Omega} \leq \left\|u^h - u\right\|_{0,\Omega\backslash\widehat{\Omega}} + \hat{C}\left\|u^h - u\right\|_{0,\Omega\backslash\widehat{\Omega}}^{1/2}\left\|u^h - u\right\|_{1,\Omega\backslash\widehat{\Omega}}^{1/2} + \frac{[\![u^\Gamma]\!]}{\ell}\left\|y^h(x) - y(x)\right\|_{0,\widehat{\Omega}}$$

For RK background discretization along with a single hidden layer parametrization

$$\left\|u^h - u\right\|_{0,\Omega} \leq \underbrace{\left(C + \hat{\hat{C}}\right) a^{\hat{\gamma}} k |u|_{p+1,\Omega\backslash\widehat{\Omega}}}_{\equiv\, \tilde{e}} + \underbrace{C_y \frac{[\![u^\Gamma]\!]}{\ell} n_{NR}^{-1/2}}_{\equiv\, \hat{e}}$$

- $\hat{\gamma} = p + 0.5$
- p: order of RK basis

- When $\tilde{e}$ dominates, the convergence rate $\approx \hat{\gamma}$ on the nodal spacing. (e.g., convergence rate = 1.5 for linear basis)
- When $\hat{e}$ dominates, the convergence rate ≈ 1 on $1/\sqrt{NR}$.

Baek, J., Chen, J. S., Computer Methods in Applied Mechanics and Engineering, Vol. 410, 116590, 2024.

1D Elasticity with Pre-degraded Zones

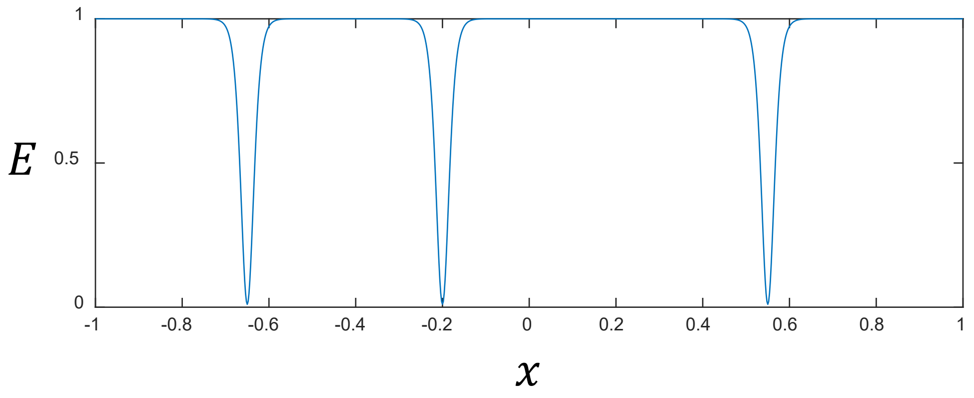
Problem

$$\min \mathcal{L} = \int_{-1}^{+1} \frac{1}{2} E u_{,x}^2 - ub \, dx + \frac{1000E}{2h}(u-g)^2$$

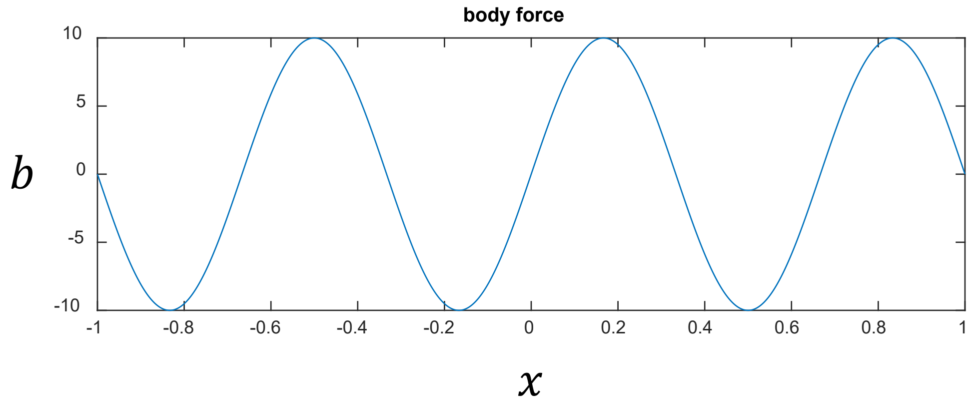
Unknowns

- Reproducing kernel approximation:
  21 uniformly distributed RK nodes
- Neural network approximation:
  36 unknowns with 4 blocks
- **<u>57 total unknowns</u>**

Young's modulus distribution

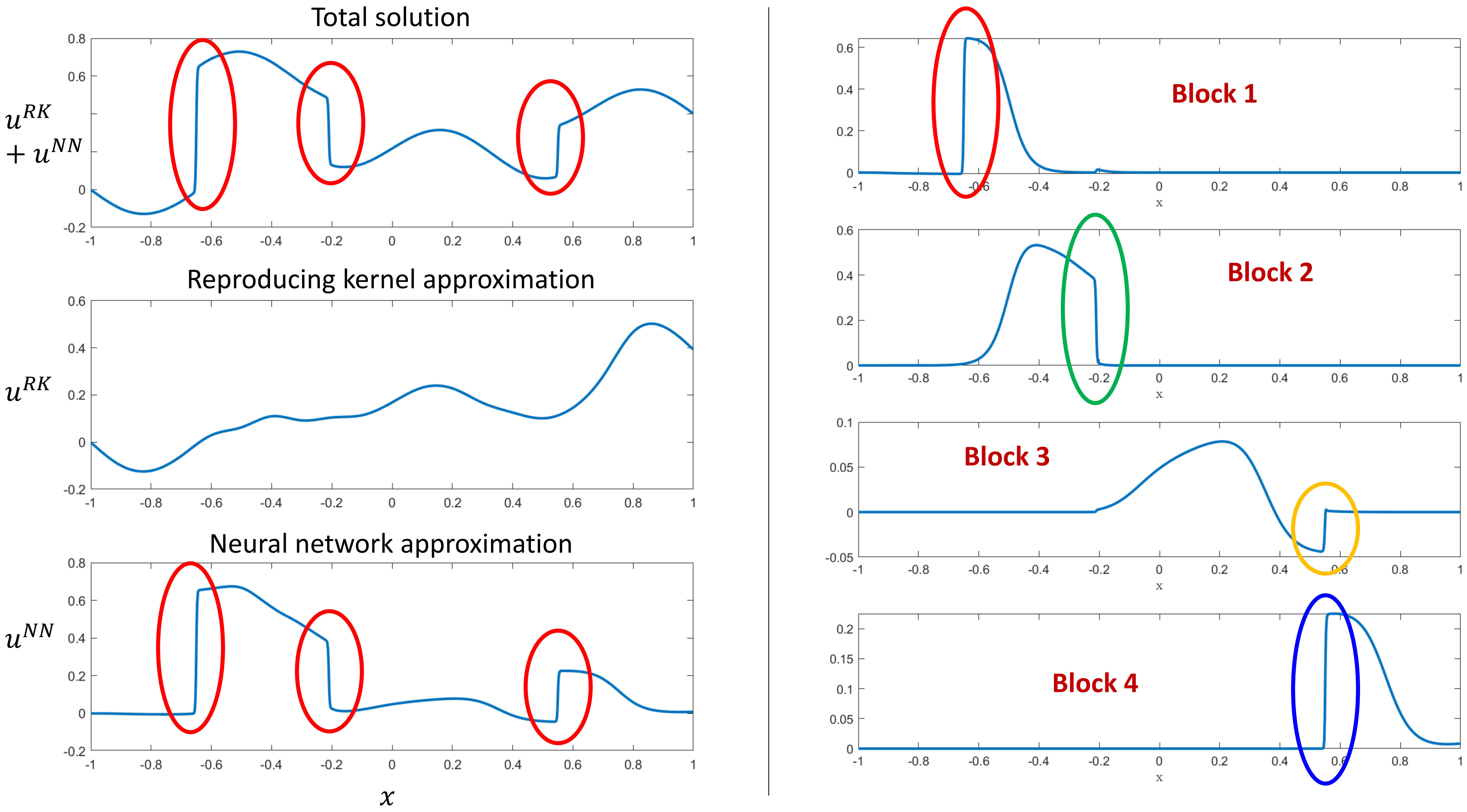


Body force distribution



1D Elasticity with Pre-degraded Zones

- Initially, kernels are uniformly distributed and the NN approximation is initialized to be zero. (i.e., the ***NN approximation initially does NOT know*** the information on the localization.)
- The ***kernels actively evolve*** during the loss function minimization and the ***sharp solution transition is captured***.

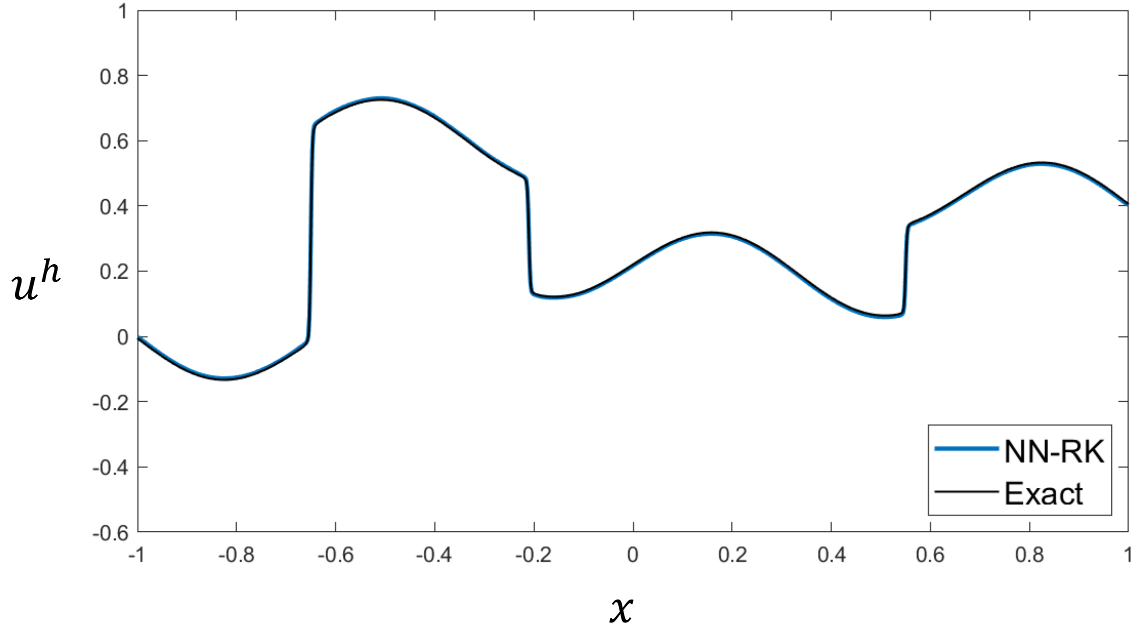


Baek, J., Chen, J. S., & Susuki, K. (2022). *International Journal for Numerical Methods in Engineering*, Vol. 123, pp 4422-4454, DOI: 10.1002/nme.7040

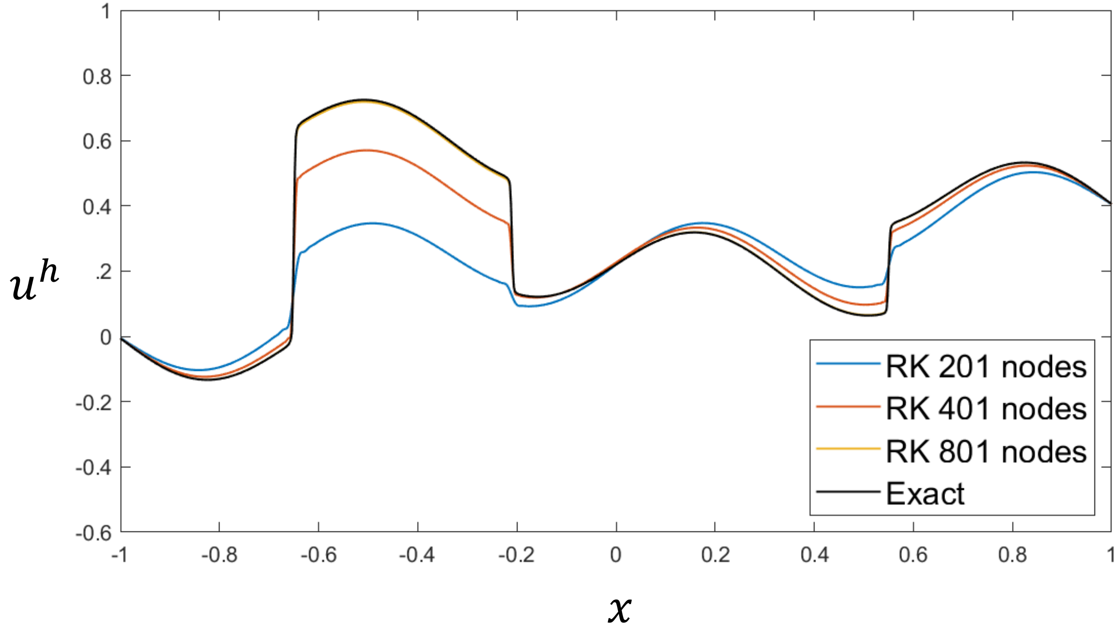
57 vs 801 unknowns

- Without NN enhancement, the transitions are <u>NOT sufficiently sharp</u> even with 201 and 401 RK nodes.

Neural network-enhanced solution (57 unknowns)

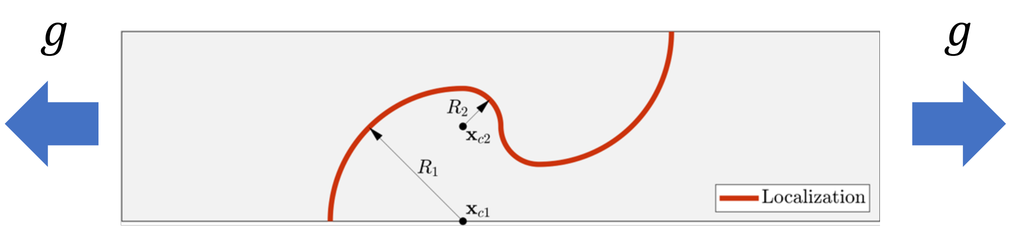


Reproducing kernel only solution



2D Elasticity with Pre-degradation

Tensile specimen with pre-degradation



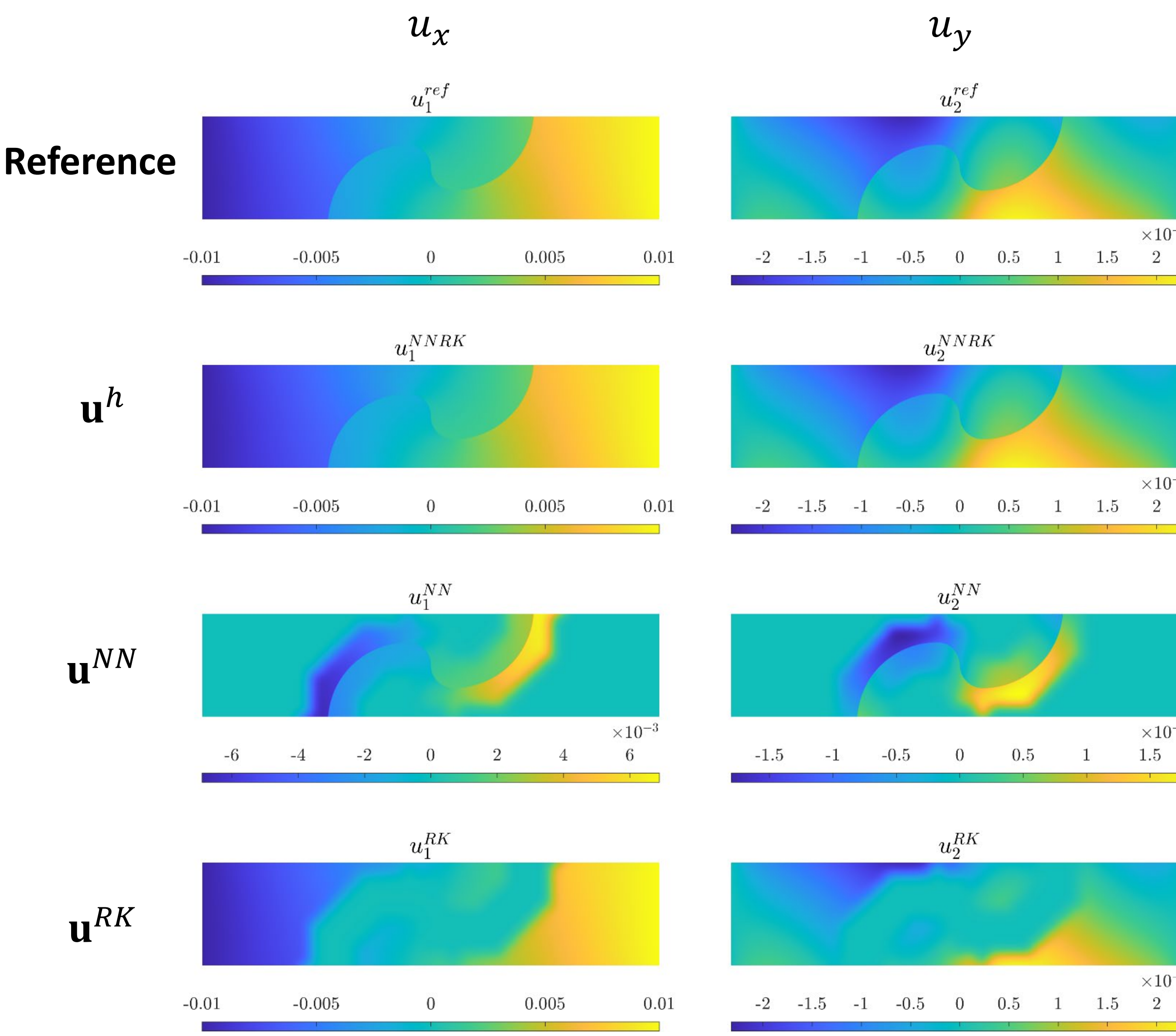
Minimization problem

$$\min \Pi = \frac{1}{2}\int_{\Omega} \boldsymbol{\varepsilon}(\mathbf{u}^h) : \mathbf{D} : \boldsymbol{\varepsilon}(\mathbf{u}^h)\, d\Omega$$

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

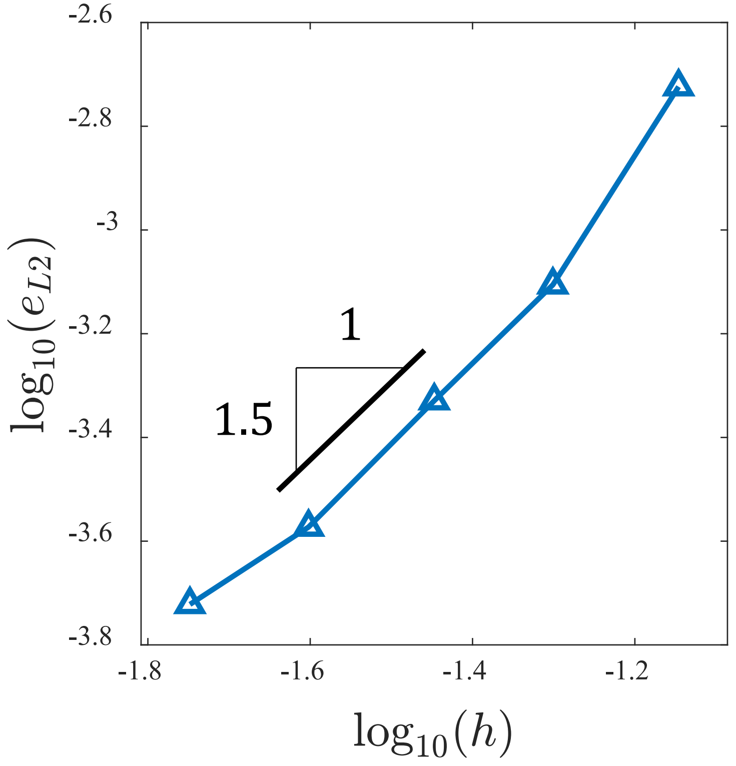
Approximation

- 451 RK nodes with linear basis (902 unknowns)
- Single hidden layer with 40 neurons for NN parametrization

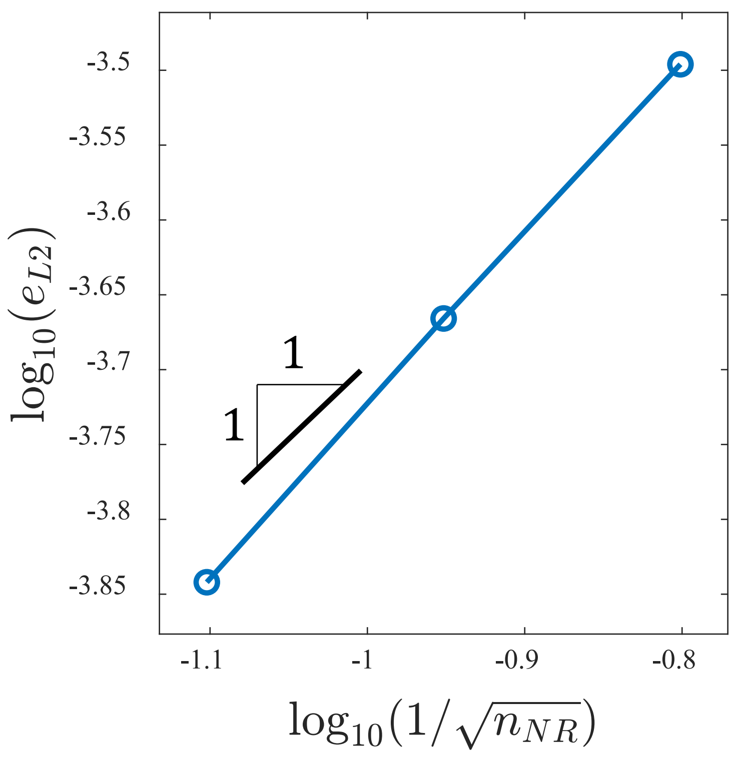


The convergence rates agree with the theoretical values.

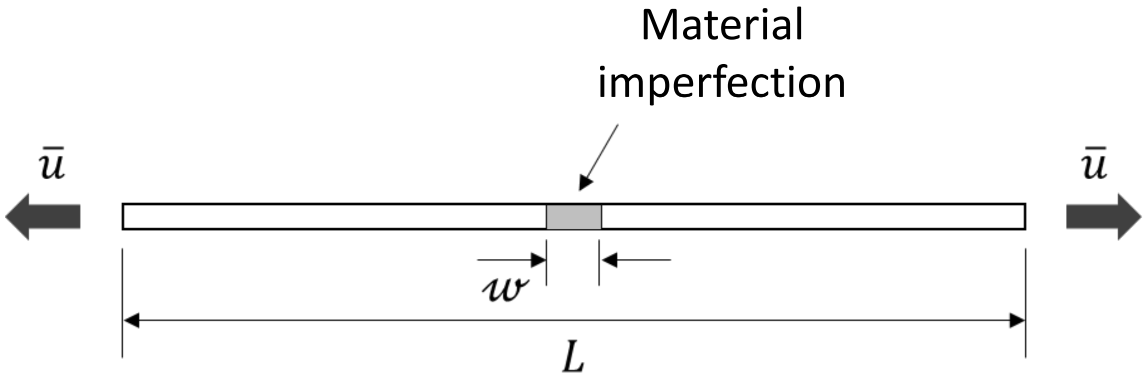
Convergence for varying h (avg. rate of 1.647)



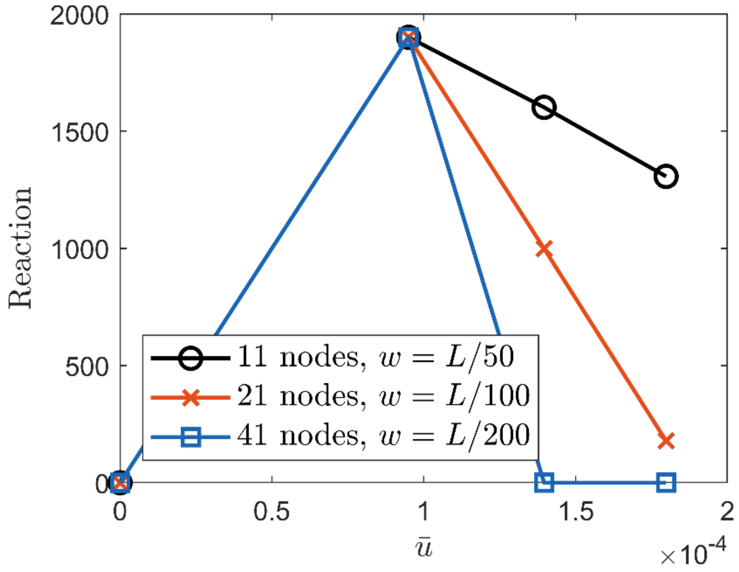
Convergence for varying n_{NR} (avg. rate of 1.150)



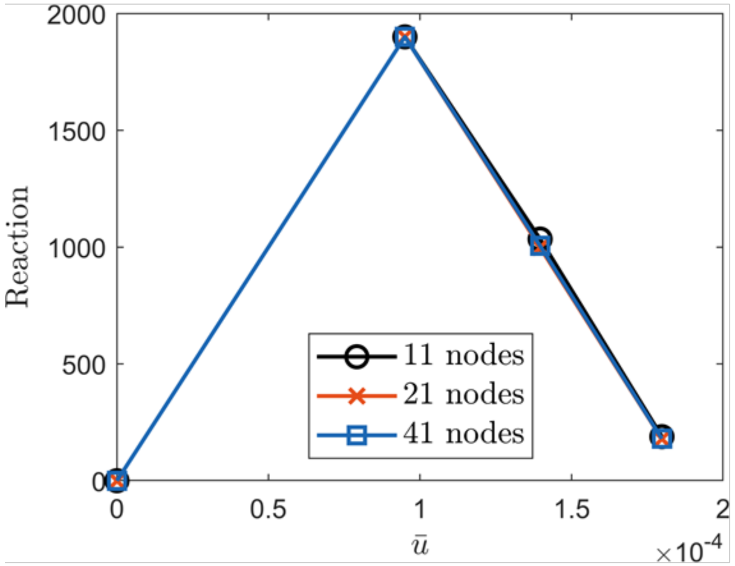
An Elastic-damage Bar Under Tension



- Compared to the un-regularized counterpart, the regularized NN-enhanced RKPM yields ***discretization-independent results***.



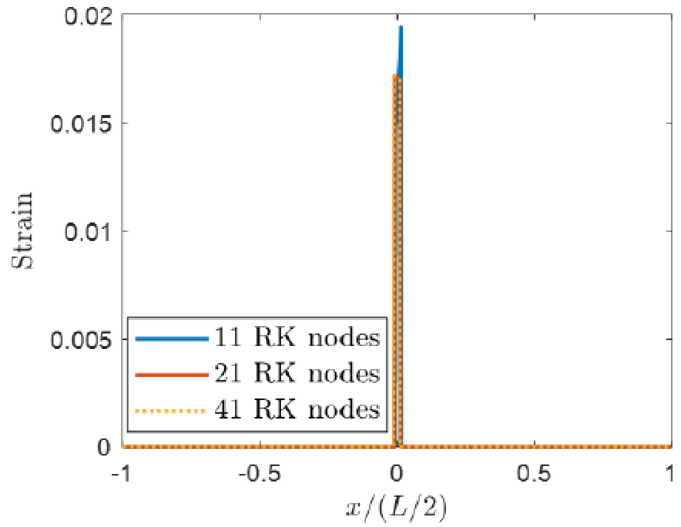
Load-displacement response of un-regularized models



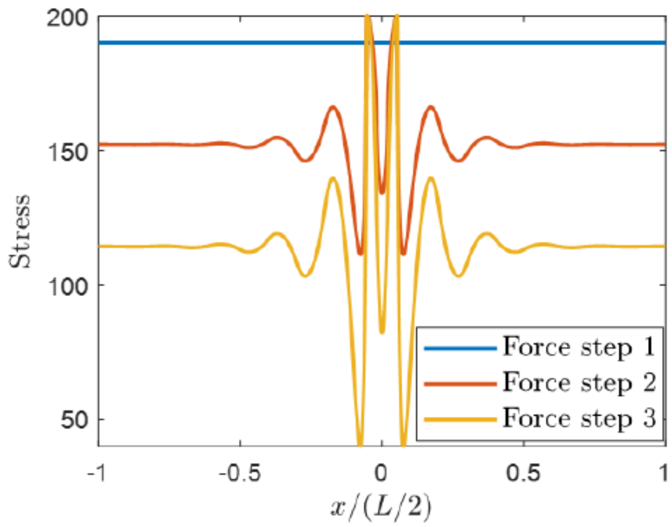
Load-displacement response of regularized NN-enhanced RKPM

Baek, J., Chen, J. S., & Susuki, K. (2022). *International Journal for Numerical Methods in Engineering*.

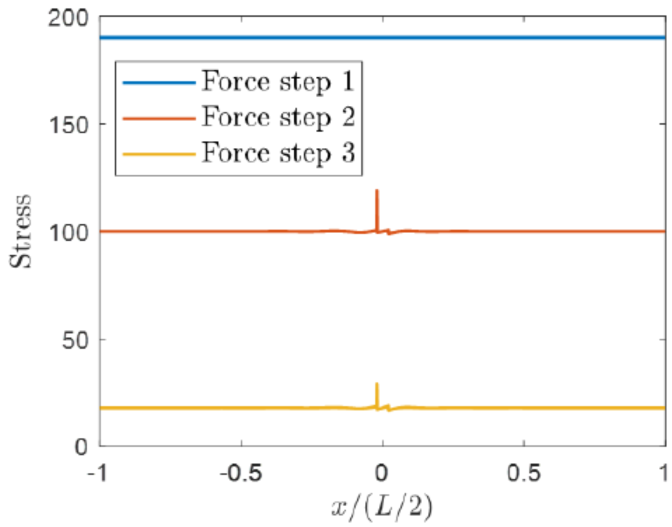
- ***Highly localized strain field*** is well captured.
- *The NN control parameter* $\beta \in \mathbf{W}^S$ ***suppresses the stress oscillation*** which is shown unless special treatment is performed.



Strain field

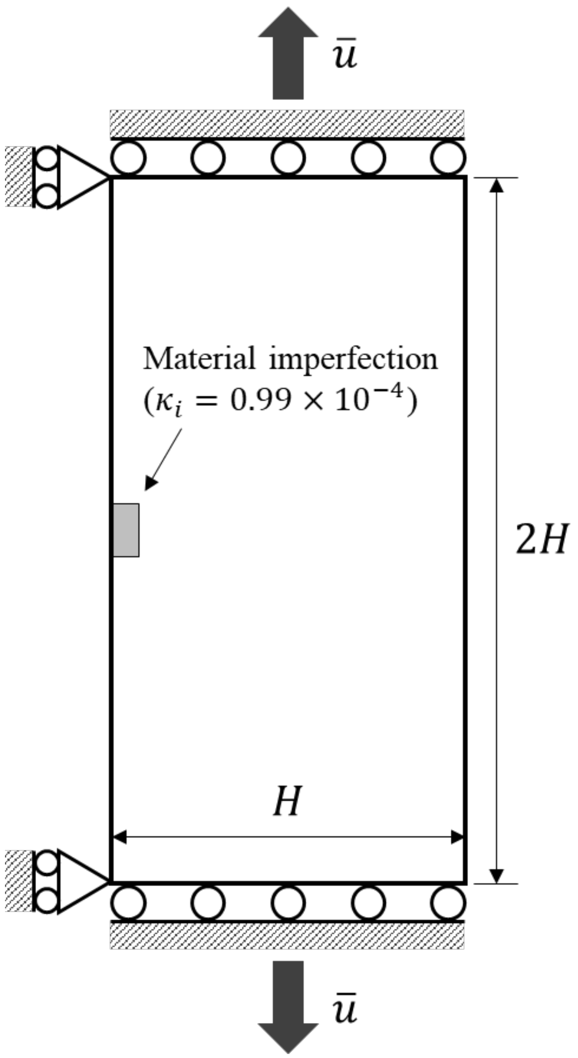


Stress obtained by smooth NN kernels

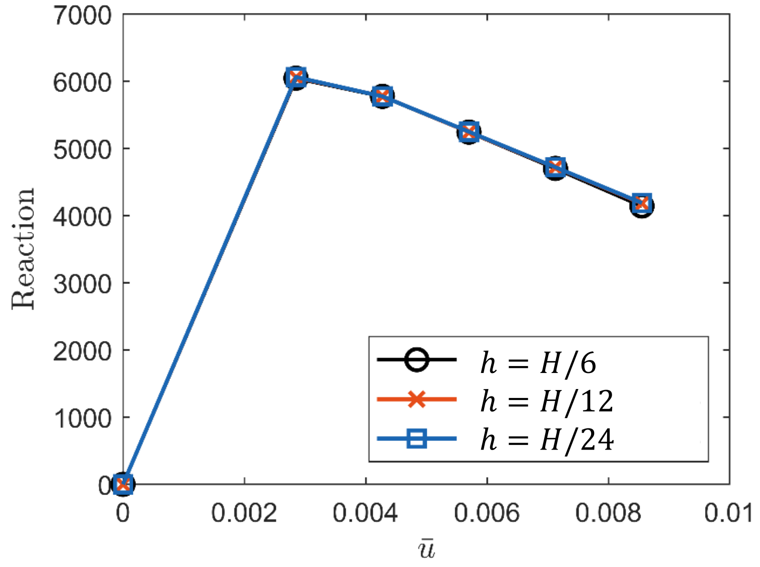


Stress obtained by adaptive NN kernels controlled by β.

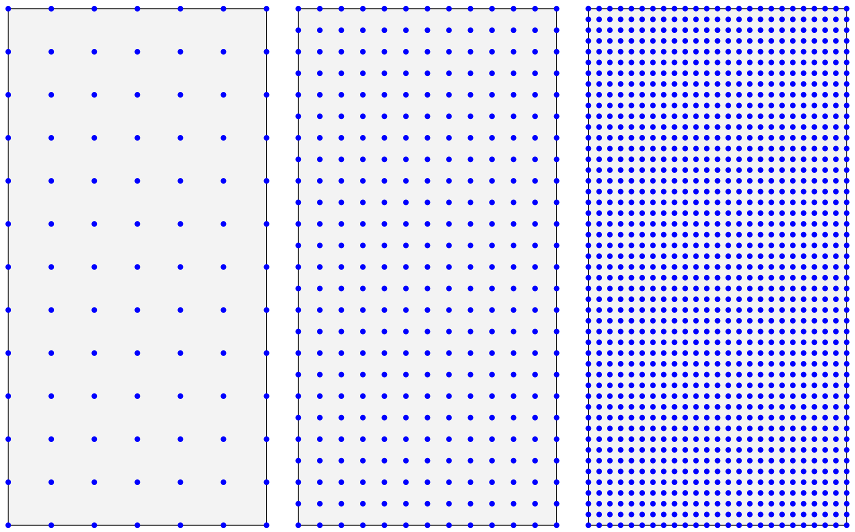
Tensile specimen with asymmetric imperfection



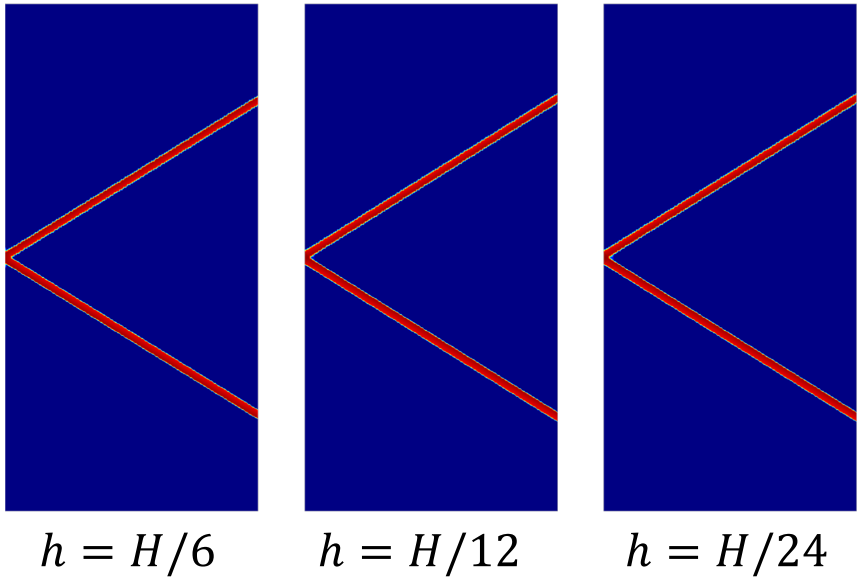
- The proposed NN-enhanced RKPM yields ***discretization-independent results***.



Load-displacement curve

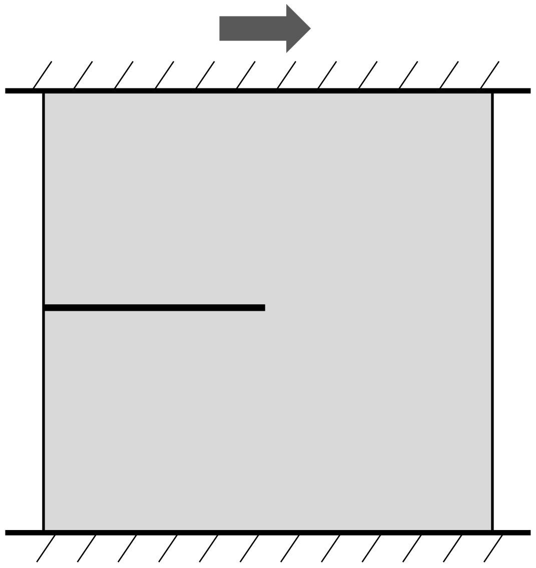


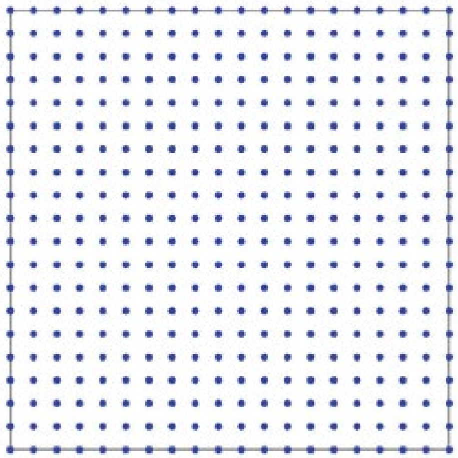
Three different discretizations



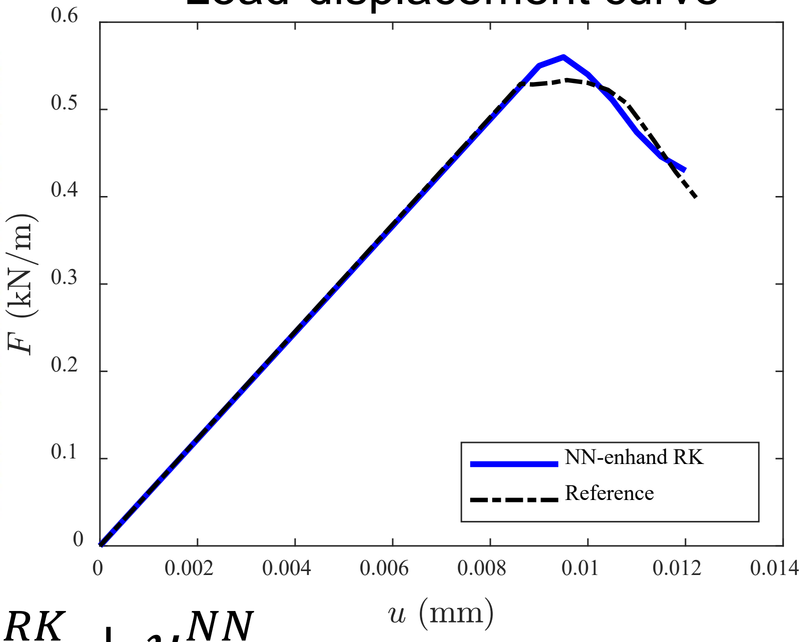
Damage pattern predicted by NN-enhanced RKPM

NN-Enhanced RKPM for Modeling Crack Propagation in Pre-notched Specimen





Load-displacement curve

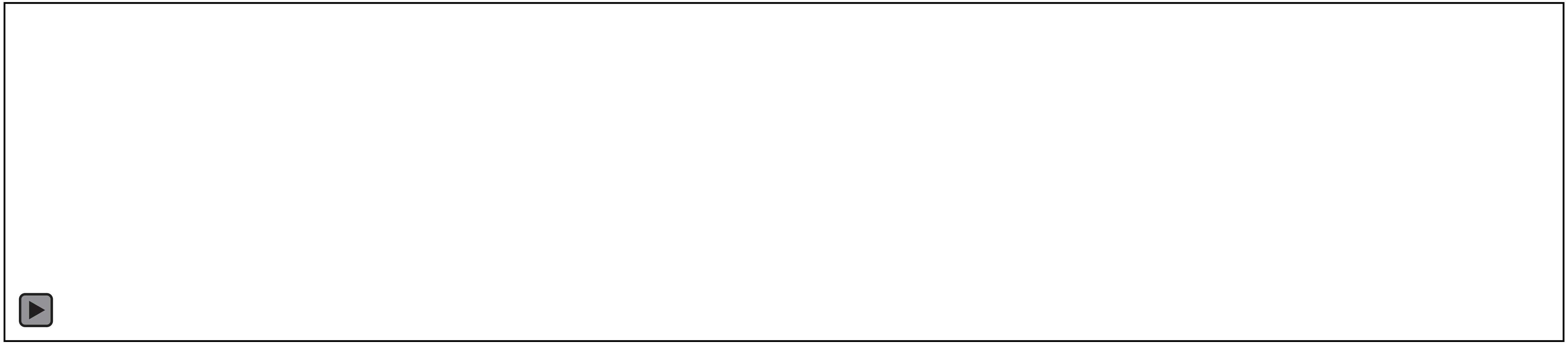


- 256 RK particles (16X16) are used with **512 RK coefficients**.
- **3 NN blocks** are used with **540 total NN unknown** weights and biases.

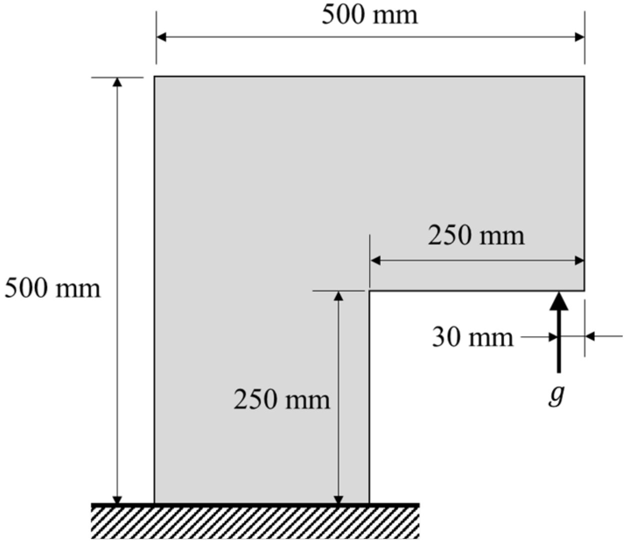
$u_1^{RK} + u_1^{NN}$

$u_2^{RK} + u_2^{NN}$

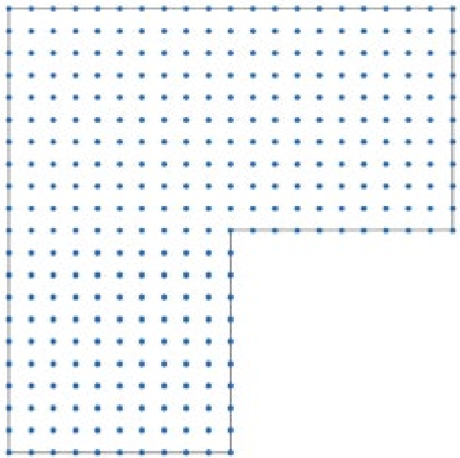
Damage

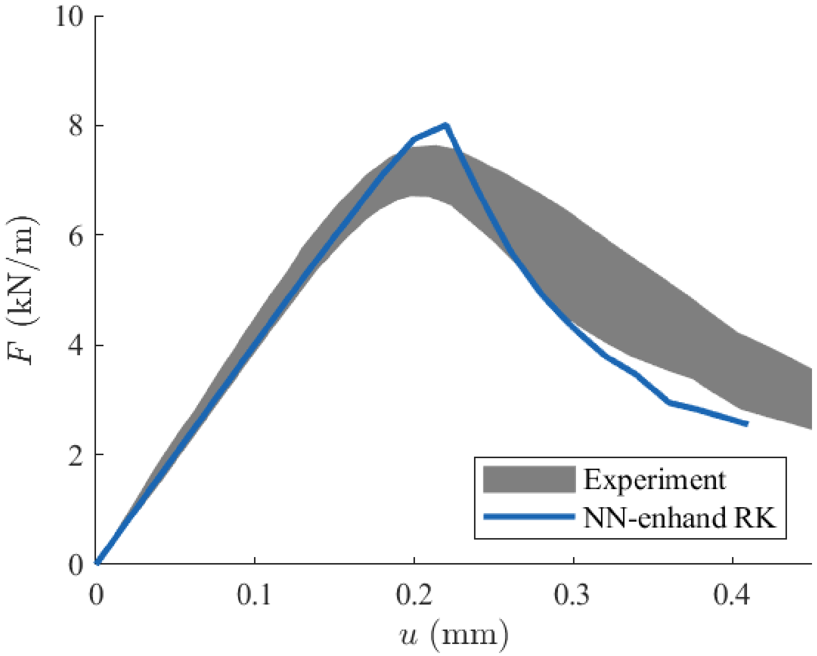


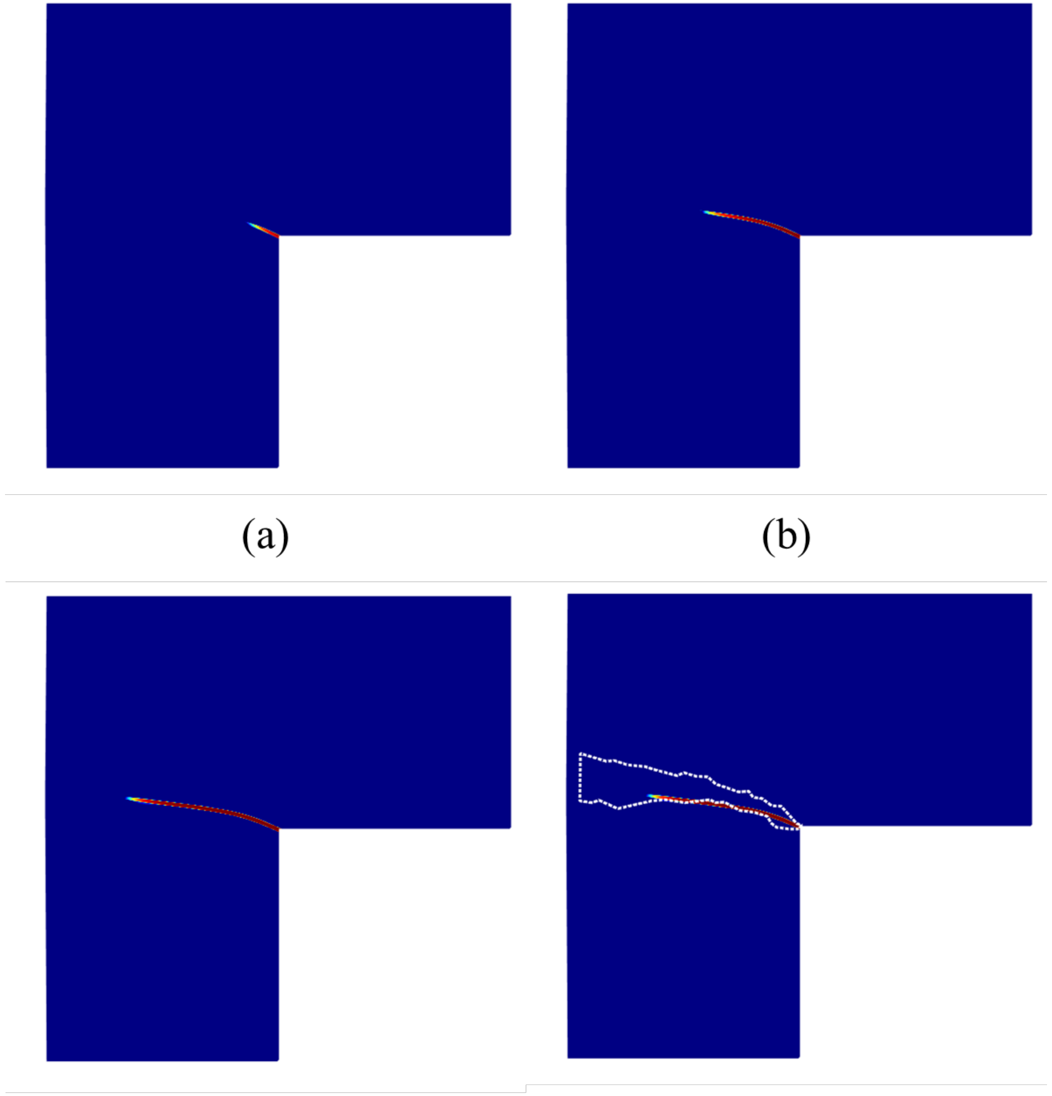
Baek, J., Chen, J. S., & Susuki, K. International Journal for Numerical Methods in Engineering. Vol. 123, pp 4422-4454, 2022.



- 256 RK particles (16X16) are used with **512 RK coefficients**.
- **1 NN block** is used with **278 total NN unknown** weights and biases.







(a) (b)

(c) (d)

Comparison with experimentally observed crack range

Baek, J., Chen, J. S., Susuki, K, International Journal for Numerical Methods in Engineering. Vol. 123, pp 4422-4454, 2022.

Loss Function for Brittle Fracture

Loss Function

$$\mathcal{L}(\mathbf{d},\mathbf{W}) = \int_\Omega \psi^D\left(\mathbf{U}^h(\mathbf{d},\mathbf{W})\right)\mathrm{d}\Omega - F\left(\mathbf{U}^h(\mathbf{d},\mathbf{W})\right) + \frac{\alpha}{2}\int_{\partial\Omega^U}\left\|\mathbf{U}^h(\mathbf{d},\mathbf{W}) - \bar{\mathbf{U}}\right\|^2 \mathrm{d}\Omega$$

$$F(\mathbf{u}) = \int_\Omega \mathbf{u}\cdot\mathbf{b}\, d\Omega + \int_{\partial\Omega_\mathrm{h}} \mathbf{u}\cdot\mathbf{h}\, d\Gamma$$

$$\psi^D(\mathbf{u}) = g\left(\eta\left(\boldsymbol{\varepsilon}(\mathbf{u})\right)\right)\psi_0^+(\mathbf{u}) + \psi_0^-(\mathbf{u}) + \bar{\psi}\left(\eta\left(\boldsymbol{\varepsilon}(\mathbf{u})\right)\right)$$ [1]

$$\psi_0^+ = \mu\langle\bar{\varepsilon}_i\rangle\langle\bar{\varepsilon}_i\rangle + \frac{\lambda}{2}\langle\textstyle\sum\bar{\varepsilon}_i\rangle^2,\ \psi_0^- = \psi^{el} - \psi_0^+$$

$$\boldsymbol{\sigma} = g\left(\eta\left(\boldsymbol{\varepsilon}(\mathbf{u})\right)\right)\frac{\partial\psi_0^+(\mathbf{u})}{\partial\boldsymbol{\varepsilon}(\mathbf{u})} + \frac{\partial\psi_0^-(\mathbf{u})}{\partial\boldsymbol{\varepsilon}(\mathbf{u})}$$

Dissipation Energy

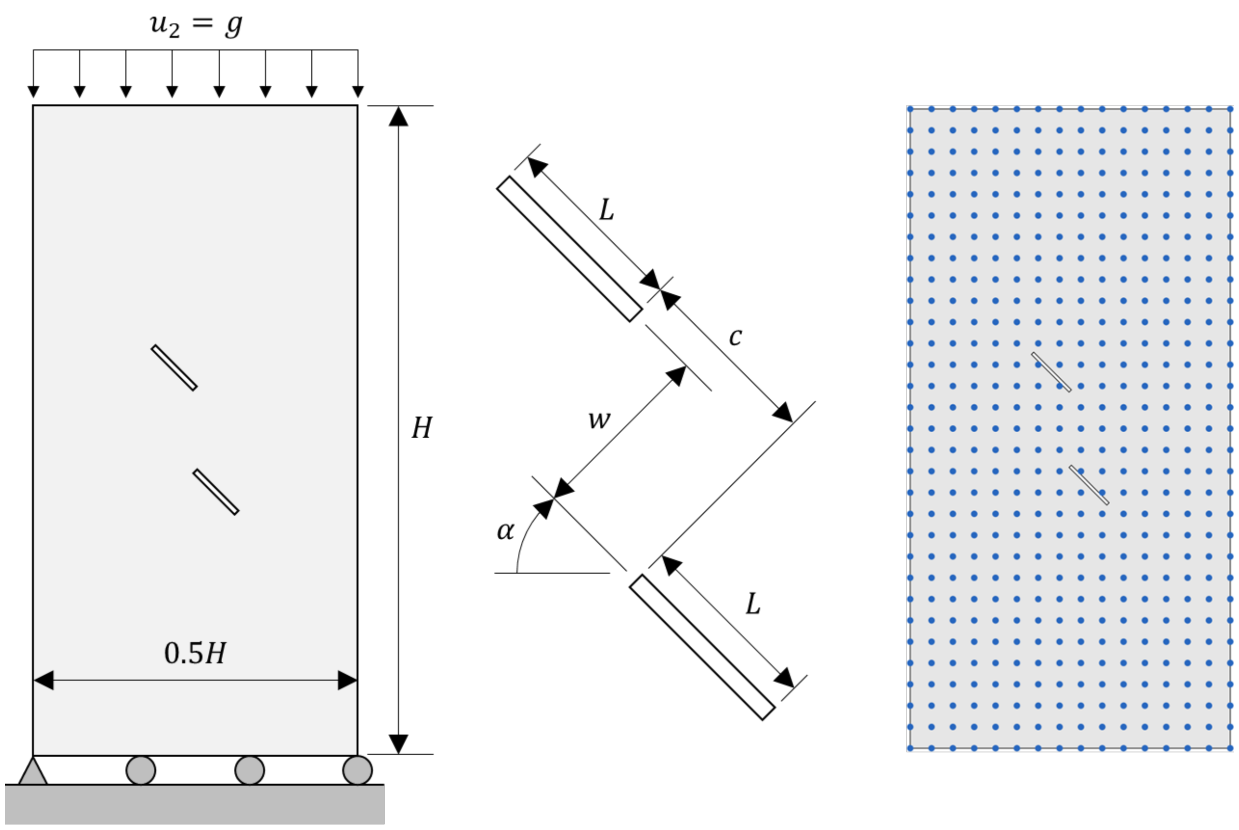
$$\bar{\psi}(\eta) = p\eta^2$$

$$\eta = \frac{\mathcal{H}}{\mathcal{H}+p},\quad \mathcal{H} = \max\left(\max_{t\in[0,T]}\{\psi_0^+(\boldsymbol{\varepsilon}) - \psi_c\}, 0\right),\quad \psi_c = \frac{f_t}{2E},\quad g = (1-\eta)^2$$

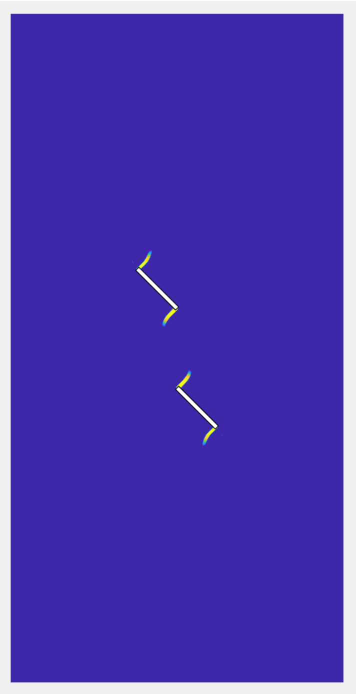
$$p = \frac{\mathcal{G}_c}{\ell},\quad \mathcal{G}_c = \frac{\psi_0^+}{\psi_I^+/\mathcal{G}_{cI} + \psi_{II}^+/\mathcal{G}_{cII}}.$$

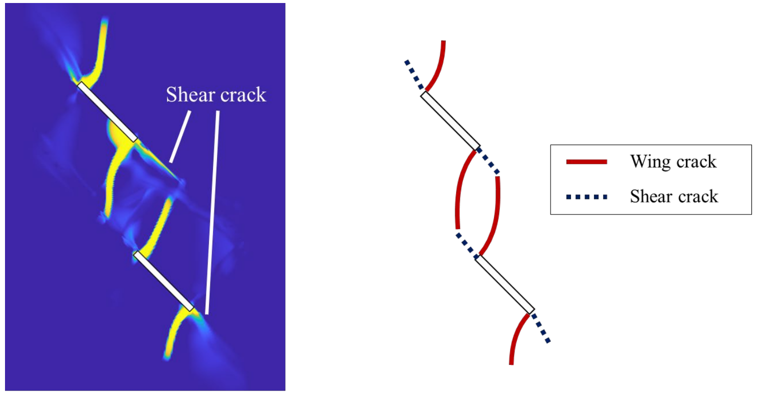
[1] C. Miehe, F. Welschinger, M. Hofacker, Int. J. Numer. Methods Eng. 83 (2010) 1273–1311.

Mixed-mode Fracture of a Doubly Notched Crack Branching



(a) (b) (c)





(a) (b)

NN-RKPM

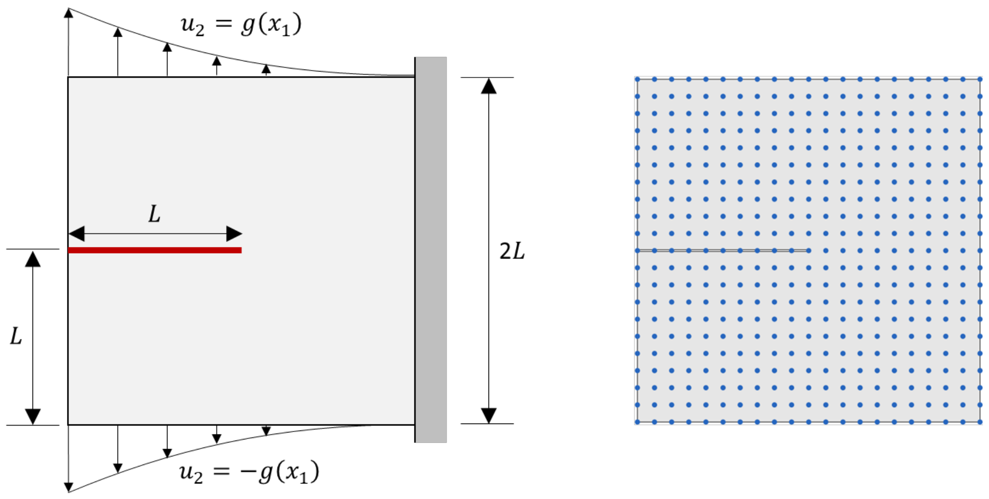
Experiment (reference)

A. Bobet, H.H. Einstein, 1998

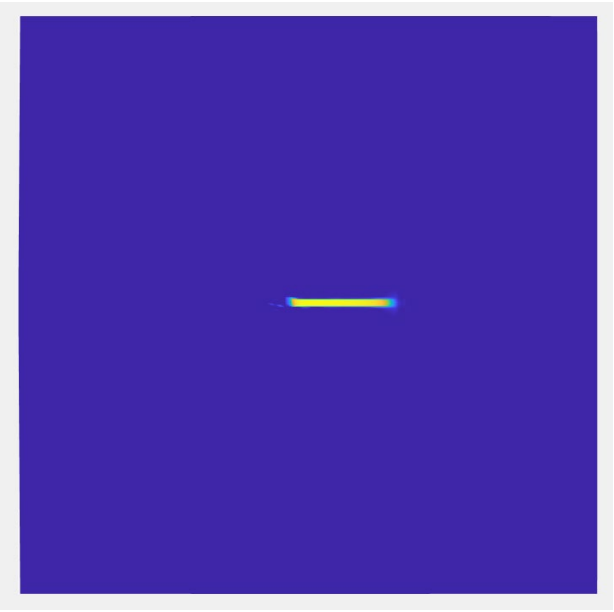
- 496 RK particles (16X31) are used with **992 RK coefficients**.
- a neural network with **two 40-neuron hidden layers, involving 1,842 unknown weights and biases**.

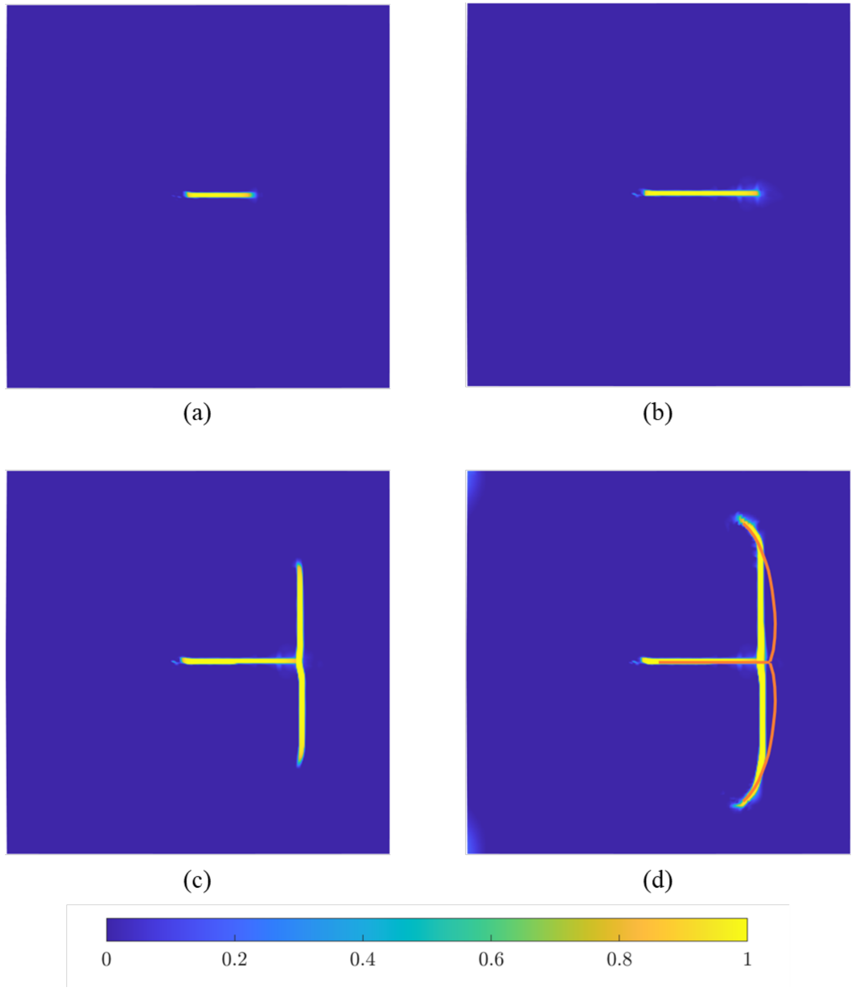
Baek, J., Chen, J. S., Computer Methods in Applied Mechanics and Engineering, Vol. 410, 116590, 2024

Fracture Branching



(a) (b)





Baek, J., Chen, J. S., Computer Methods in Applied Mechanics and EngineeringVol. 410, 116590, 2024

Phase Field Grain Growth Simulation

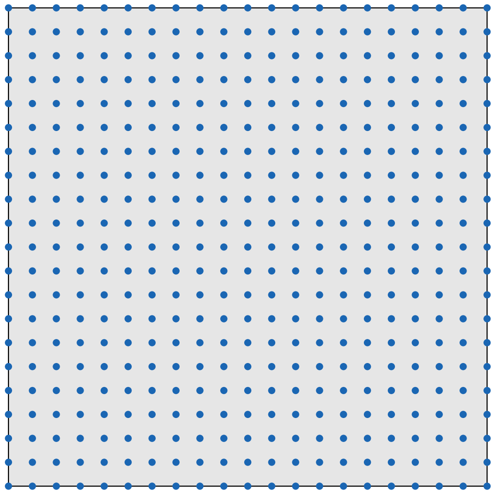
Time-discretized orientation-phase field problem

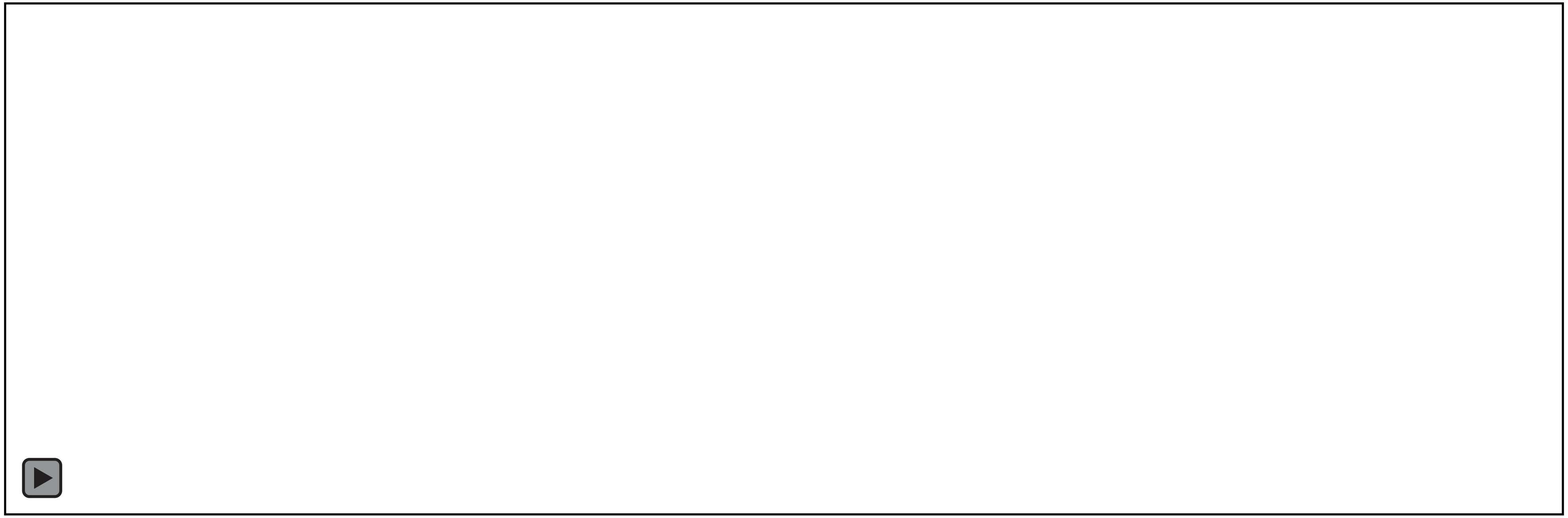
$$\min \Pi = \int_{\Omega} \left(\frac{\rho}{2\Delta t} \theta(\theta - 2\bar{\theta}) + \alpha \|\nabla\theta\| + \frac{\beta}{2} \|\nabla\theta\|^2 + \frac{p}{2}\eta^2 + \frac{q}{2}\|\nabla\eta\|^2 \right) d\Omega$$

- DOF: lattice orientation θ and phase field η
- $\rho, \alpha, \beta, p, q$: model parameters
- $\bar{\theta}$: lattice orientation from the previous time step

- **441 background RK** nodes with linear basis
- **Two 4-kernel NN blocks**
- 20-neuron 2-hidden layer parametrization network (1044 unknown NN parameters)

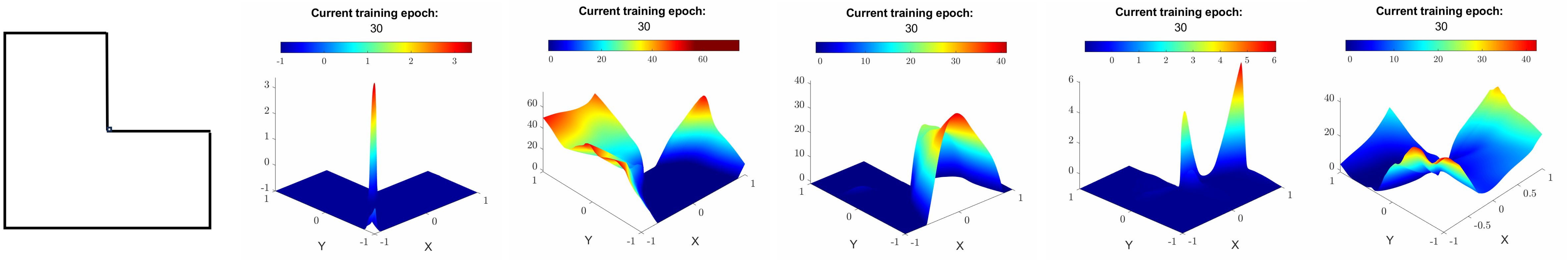
Background RK discretization



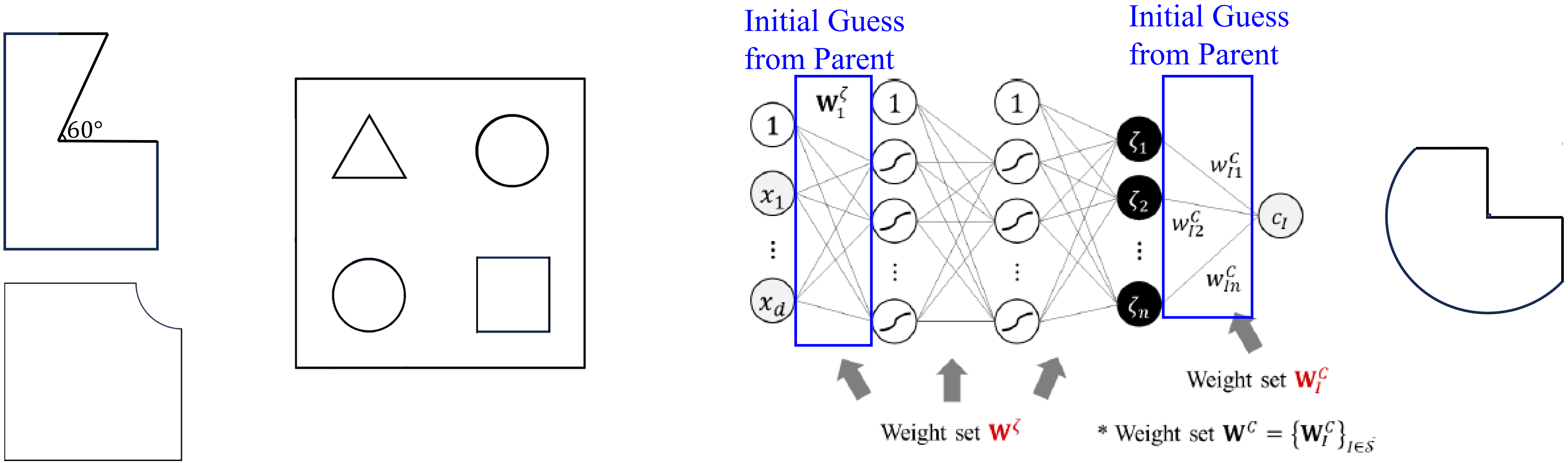


Transfer Learning of Neural Network Basis Functions

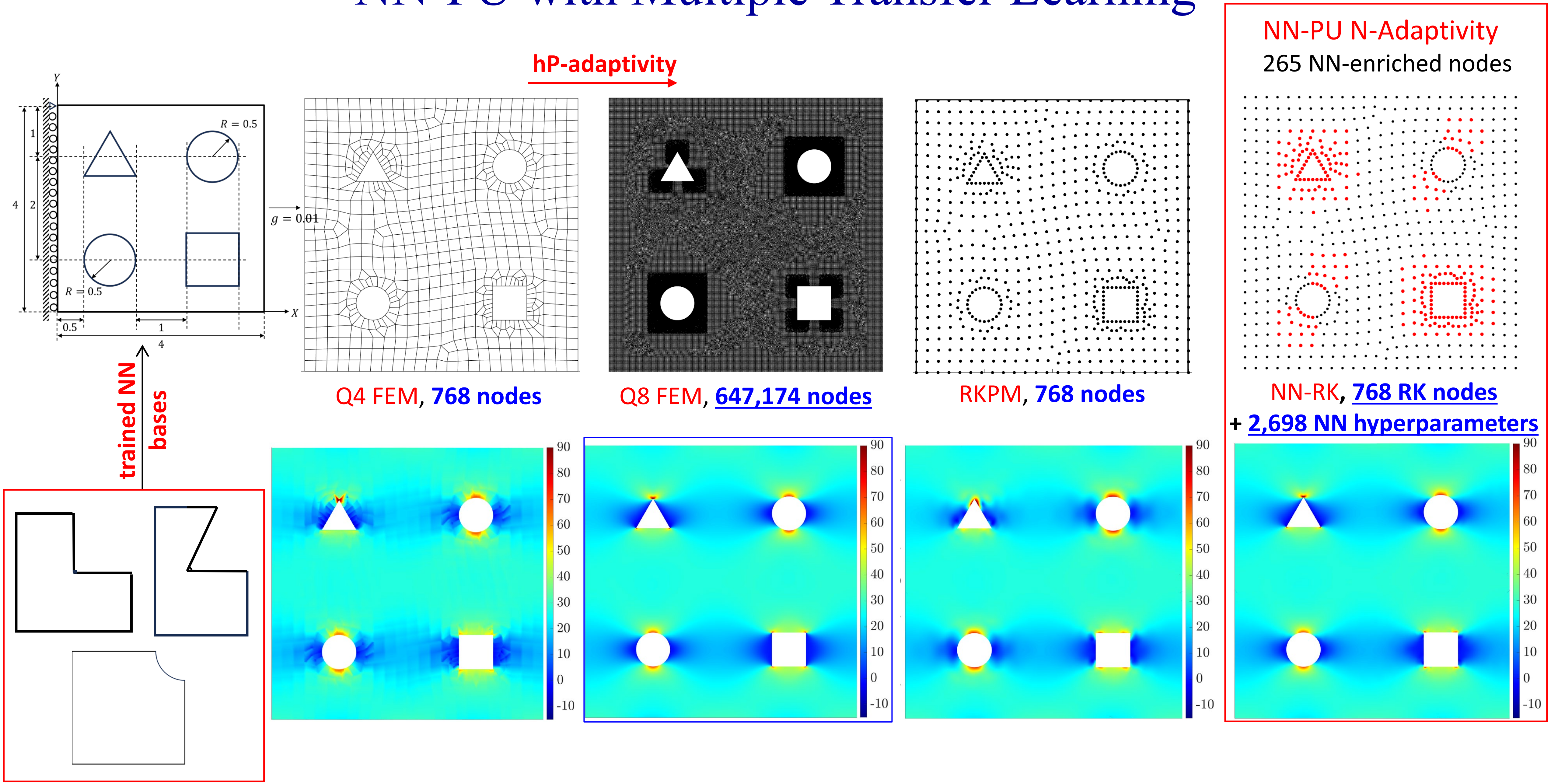
Off-Line Training of “Parent” NN Basis Functions



On-Line Transfer Learning of “Feature Encoded” NN Basis Functions



NN-PU with Multiple Transfer Learning



Conclusion

- A neural network-enhanced reproducing kernel particle method was developed.
  - The NN control parameters that determines the location, orientation, and transition shape are automatically found during the loss optimization.
  - Block-level neural network approximation allows a sparse neural network with significantly small number of unknown NN parameters.
  - Complex localizations can be captured by the superposition of multiple NN blocks, each of which can capture low order topological geometries such as a triple or a quadruple junctions.

- Energy based Neural Network Enriched NN-PU Ritz method:
  - Localization and fracture process modeling with a fixed discretization
  - Neural Network enrichment function and transfer learning for local features
  - N-adaptivity for PDE solver without re-meshing demonstrated superior computational efficiency and human effort reduction compared to FEM