



Cong Tien Nguyen · Selda Oterkus  · Erkan Oterkus

A peridynamic-based machine learning model for one-dimensional and two-dimensional structures

Received: 27 January 2020 / Accepted: 23 July 2020
© The Author(s) 2020

Abstract With the rapid growth of available data and computing resources, using data-driven models is a potential approach in many scientific disciplines and engineering. However, for complex physical phenomena that have limited data, the data-driven models are lacking robustness and fail to provide good predictions. Theory-guided data science is the recent technology that can take advantage of both physics-driven and data-driven models. This study presents a novel peridynamics-based machine learning model for one- and two-dimensional structures. The linear relationships between the displacement of a material point and displacements of its family members and applied forces are obtained for the machine learning model by using linear regression. The numerical procedure for coupling the peridynamic model and the machine learning model is also provided. The numerical procedure for coupling the peridynamic model and the machine learning model is also provided. The accuracy of the coupled model is verified by considering various examples of a one-dimensional bar and two-dimensional plate. To further demonstrate the capabilities of the coupled model, damage prediction for a plate with a preexisting crack, a two-dimensional representation of a three-point bending test and a plate subjected to dynamic load are simulated.

Keywords Machine learning · Peridynamics · Fracture · Peridynamic-based machine learning · Linear regression

1 Introduction

Predicting progressive failures in structures is a challenging task in engineering. The classical continuum mechanics faces conceptual and mathematical difficulties in terms of predicting crack nucleation and growth, especially for multiple crack paths because it uses differential equations. In contrast, peridynamics (PD) is a nonlocal theory representing material behavior by using integro-differential equations that are valid in both continuous and discontinuous models [1–5]. Therefore, PD is suitable for predicting progressive damages.

Peridynamics can be applicable for both elastic and plastic materials [6–10], composite and polycrystalline materials [11–16], multiphysics [17–19], large deformation problems [20], topology optimization [21] and multiscale modeling [22, 23]. Peridynamics can also be suitable for structural idealization to analyze slender structures by using PD beam models [24–27] or thin wall structures by using PD plate and shell models [27–32]. Moreover, peridynamic differential operator is used to approximate field variables and their temporal and spatial derivatives for the conversion of the local form of differentiation to its nonlocal integral form [33–36].

Communicated by Luca Placidi.

C. Tien · S. Oterkus (✉) · E. Oterkus
Department of Naval Architecture, Ocean and Marine Engineering, PeriDynamics Research Centre (PDRC),
University of Strathclyde, Glasgow, UK
E-mail: selda.oterkus@strath.ac.uk

Peridynamics can also be combined with other well-known numerical methods such as finite element analysis (FEA) [12,37–39] or smooth particle method [40,41], or computational fluid dynamics (CFD) [42]. Extensive literature surveys on peridynamics are given in [4,5,43].

However, similar to other physics-based models, solving PD equations of motion could be time-consuming, especially when real-time predictions on live data are required. In contrast, with the support of computer resources as well as the rapid growth of available data, artificial intelligence (AI), machine learning (ML) and data analytics are providing an alternative solution for physics-based models. These data-driven models can be applicable for many scientific disciplines such as image recognition [44], natural language processing [45], cognitive science [46] and genomics [47]. In engineering, machine learning and artificial intelligence also show potential applications in many areas, including material science [48], fluid dynamics [49,50], structural health monitoring [51], additive manufacturing [52] and fracture mechanics and failure analysis [53–55].

The study in [53] used time-series algorithms named “long short-term memory” and “multilayer neural network” to predict fatigue damages in structures subjected to cyclic loading. The time series algorithms are trained by using a dataset obtained from numerical predictions. The dataset includes crack growth direction and the crack growth data (a–N curve). The data for the crack growth direction are determined by the maximum circumferential stress criterion. Meanwhile, the data for the crack growth under cyclic loading are predicted by using Paris’ law.

The study in [54] used neural networks as material models within a multiscale approach. Specifically, the neural networks are used to approximate the mesoscale response for homogenization procedure to couple mesoscale and macroscale. Meanwhile, the damages in structures are predicted by using the cohesive model.

The study in [55] proposed convolutional neural networks for predicting the damage patterns on a disk hit by an indenter. First, the PD simulations to predict crack patterns on a disk hit by an indenter are conducted to generate the data set. Later, the obtained data set is used to train the neural networks. Therefore, the trained neural networks can be used to predict damage patterns on a disk subjected to different impact loadings. Moreover, as the inverse problem, the trained neural networks can also identify the collision location, angle, velocity and size of the indenter from a given damage pattern on the disk.

As can be observed from [53,55], the neural networks are trained by using the crack propagation data predicted by trial numerical predictions, which can be very computationally expensive and limited. On the other hand, when available data are limited, the vast majority of recent machine learning techniques are lacking robustness and accuracy [56]. Therefore, a hybrid approach of combining machine learning and physics-based modeling becomes highly beneficial. Therefore, in this study, a hybrid approach of coupling machine learning and peridynamic models for fracture prediction of structures is presented. Specifically, the PD model is applied for special regions in structures such as near crack surfaces or near boundary areas. Meanwhile, the ML model is used for the remaining regions to reduce the computational cost.

The machine learning models to find displacements of a material point based on displacements of its family members and its external body forces in one-dimensional (1D) and two-dimensional (2D) structures are presented. The numerical procedure for the coupling of the ML model and PD model is also provided. The capability of the hybrid approach is verified by considering various examples for 1D and 2D structures. The results predicted by the coupled models are compared with FEA and conventional PD results. For further verifying the capability of the coupling model, progressive damages in a plate with a preexisting crack subjected to tension, on a 2D representation of a three-point bending test, on a plate subjected to dynamic loads are presented.

This paper is organized as follows. Section 2 presents multiple linear regression. Sections 3 and 4 present PD theory and PD-based machine learning models for 1D and 2D structures, respectively. Section 5 presents the numerical implementation. Section 6 presents the verification for the developed PD-based machine learning models. Finally, the conclusion is presented in Sect. 7.

2 Multiple linear regression

In this section, the basic concept of multiple linear regression is presented. According to Montgomery et al. [57] and Alpaydin [58], in the multiple linear regression, the numeric output y is assumed to be written as a function of several input variables, x_1, x_2, \dots, x_N and noise as

$$y = w_0 + w_1x_1 + w_2x_2 + \dots + w_Nx_N + \varepsilon \quad (1)$$

where x_1, x_2, \dots, x_N represent the input variables that can be called the regressors. The parameters w_0, w_1, \dots, w_N represent regression coefficients that need to be determined. The parameter ε represents the noise of the model.

The coefficients given in Eq. (1) can be determined based on the least square criterion [57,58]. The sum of the squares of the differences between the predicted values and the correct outputs is minimized. The minimization of the squared errors can be expressed as follows;

Let's assume that we have training data as $(x_{1(i)}, x_{2(i)}, \dots, x_{N(i)}, y(i))$, $i = 1, \dots, M$. Therefore, Eq. (1) can be rewritten for every single data as

$$y_i = w_0 + w_1 x_{1(i)} + w_2 x_{2(i)} + \dots + w_N x_{N(i)} + \varepsilon(i), \quad i = 1, 2, \dots, M \quad (2)$$

The sum of squared errors can be calculated as

$$S(w_0, w_1, \dots, w_N) = \sum_{i=1}^M (y_i - (w_0 + w_1 x_{1(i)} + w_2 x_{2(i)} + \dots + w_N x_{N(i)}))^2 \quad (3)$$

In order to minimize the squared errors given in Eq. (3), the coefficients w_j , $j = 1, 2, \dots, N$ must satisfy the following conditions

$$\frac{\partial S}{\partial w_j} = 0, \quad j = 0, 1, 2, \dots, N \quad (4)$$

By substituting Eq. (3) into Eq. (4), the following relations are obtained

$$\begin{aligned} \sum_{i=1}^M y_i &= M w_0 + w_1 \sum_{i=1}^M x_{1(i)} + w_2 \sum_{i=1}^M x_{2(i)} + \dots + w_N \sum_{i=1}^M x_{N(i)} \\ \sum_{i=1}^M x_{1(i)} y_i &= w_0 \sum_{i=1}^M x_{1(i)} + w_1 \sum_{i=1}^M (x_{1(i)})^2 + w_2 \sum_{i=1}^M x_{1(i)} x_{2(i)} + \dots + w_N \sum_{i=1}^M x_{1(i)} x_{N(i)} \\ &\vdots \\ \sum_{i=1}^M x_{N(i)} y_i &= w_0 \sum_{i=1}^M x_{N(i)} + w_1 \sum_{i=1}^M x_{N(i)} x_{1(i)} + w_2 \sum_{i=1}^M x_{N(i)} x_{2(i)} + \dots + w_N \sum_{i=1}^M (x_{N(i)})^2 \end{aligned} \quad (5)$$

The relations given in Eq. (5) can be rewritten in a matrix form as

$$\mathbf{X}^T \mathbf{X} \mathbf{w} = \mathbf{X}^T \mathbf{y} \quad (6a)$$

with

$$\mathbf{X} = \begin{bmatrix} 1 & x_{1(1)} & x_{2(1)} & \dots & x_{N(1)} \\ 1 & x_{1(2)} & x_{2(2)} & \dots & x_{N(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{1(M)} & x_{2(M)} & \dots & x_{N(M)} \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_N \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(M) \end{bmatrix} \quad (6b)$$

Therefore, the vector of coefficients, \mathbf{w} , can be obtained as

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}. \quad (7)$$

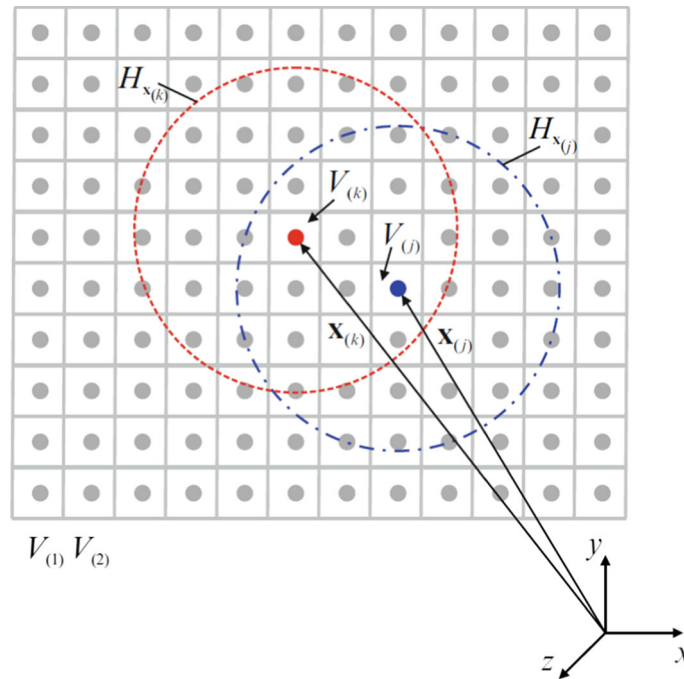


Fig. 1 PD material points and interaction of material points at $\mathbf{x}^{(k)}$ and $\mathbf{x}^{(j)}$ [5]

3 Peridynamic theory

Peridynamics (PD) is a nonlocal theory representing material behavior by using integro-differential equations. In peridynamics, a material point k located at coordinates $\mathbf{x}^{(k)}$ has interactions with its surround points within a distance δ as shown in Fig. 1. The circle, $H_{x^{(k)}}$ surrounding material point k with the radius of δ is called the horizon of this material point. The radius δ is called horizon size and material points inside the horizon $H_{x^{(k)}}$, except material point k , are family members of the material point k . The bond-based peridynamic equation of motion introduced by Silling [1,3] is an integro-differential equation in time and space as

$$\rho(\mathbf{x}) \ddot{\mathbf{u}}(\mathbf{x}, t) = \int_{H_x} (\mathbf{f}(\mathbf{u}' - \mathbf{u}, \mathbf{x}' - \mathbf{x}, t)) dV' + \mathbf{b}(\mathbf{x}, t) \quad (8a)$$

which can be rewritten in the discretized form as

$$\rho_{(k)} \ddot{\mathbf{u}}_{(k)} = \sum_{j=1}^N (\mathbf{f}_{(k)(j)}(\mathbf{u}_{(j)} - \mathbf{u}_{(k)}, \mathbf{x}_{(j)} - \mathbf{x}_{(k)}, t)) V_{(j)} + \mathbf{b}_{(k)} \quad (8b)$$

where N represents the number of family members of the material point k and j represents a family member of material point k . Each material point is identified by its coordinates, $\mathbf{x}_{(k)}$, volume $V_{(k)}$ and mass density $\rho_{(k)}$. The displacement and acceleration are denoted by $\mathbf{u}_{(k)}$ and $\ddot{\mathbf{u}}_{(k)}$, respectively. The term $V_{(j)}$ represents the volume of the material point j , and $\mathbf{f}_{(k)(j)}$ represents the vector of interaction forces. The terms $\mathbf{b}(\mathbf{x}, t)$ or $\mathbf{b}_{(k)}$ represent the body forces applied to the material point k .

3.1 PD equation of motion for 1D structures

The PD equation of motion given in Eq. (8b) can be rewritten for a linear elastic bar as [1,4,24,59]

$$\rho \ddot{u}_{(k)} = C_{ax} \sum_{j=1}^N \frac{u_{(j)} - u_{(k)}}{\xi} V_{(j)} + b_{x(k)} \quad (9a)$$

with

$$C_{ax} = \frac{2E}{A\delta^2} \quad (9b)$$

$$b_{x(k)} = \frac{F_{x(k)}}{A\Delta x} \quad (9c)$$

$$\xi = |x_{(j)} - x_{(k)}| \quad (9d)$$

where $b_{x(k)}$ represents the axial body force applied on the material point k , ξ represents the distance between material points k and j . The term C_{ax} represents the PD bond constant.

For a static state with $\ddot{u}_{(k)} = 0$, Eq. (9a) can be rewritten as

$$b_{x(k)} = -C_{ax} \sum_{j=1}^N \left(\frac{u_{(j)} - u_{(k)}}{\xi} \right) V_{(j)}. \quad (10)$$

3.2 PD equations of motion for 2D structures

According to Silling [1], Silling and Lehoucq [4], Diyaroglu [59], Nguyen and Oterkus [31], the PD equation of motion given in Eq. (8b) can be rewritten in the linearized form for 2D structures as

$$\rho h \ddot{u}_{(k)} = \sum_{j=1}^N C \mu_{(k)(j)} s_{(k)(j)} \cos \varphi V_{(j)} + \bar{b}_{x(k)} \quad (11a)$$

$$\rho h \ddot{v}_{(k)} = \sum_{j=1}^N C \mu_{(k)(j)} s_{(k)(j)} \sin \varphi V_{(j)} + \bar{b}_{y(k)} \quad (11b)$$

with

$$\bar{b}_{x(k)} = \frac{F_{x(k)}}{\Delta x^2} \quad (11c)$$

$$\bar{b}_{y(k)} = \frac{F_{y(k)}}{\Delta x^2} \quad (11d)$$

$$s_{(k)(j)} = \frac{(u_{(j)} - u_{(k)}) \cos \varphi + (v_{(j)} - v_{(k)}) \sin \varphi}{\xi} \quad (11e)$$

$$\xi = \sqrt{(x_{(j)} - x_{(k)})^2 + (y_{(j)} - y_{(k)})^2} \quad (11f)$$

$$C = \frac{12E}{(1 + \nu) \pi \delta^3} \quad (11g)$$

$$\cos \varphi = \frac{x_{(j)} - x_{(k)}}{\xi} \quad (11h)$$

$$\sin \varphi = \frac{y_{(j)} - y_{(k)}}{\xi} \quad (11i)$$

where $\bar{b}_{x(k)}$ and $\bar{b}_{y(k)}$ represent applied force per unit area [31], Δx represents uniform mesh size in the PD discretized model. The term $s_{(k)(j)}$ represents linearized bond stretch, h represents the thickness of the plate, C represents bond constant, and $(x_{(k)}, y_{(k)})$ and $(x_{(j)}, y_{(j)})$ represent coordinates of material points k and j , respectively.

In Eqs. (11a–b), the parameter $\mu_{(k)(j)}$ represents the interaction state between material points k and j which can be defined as [3]

$$\mu_{(k)(j)} = \begin{cases} 1 & \text{interaction exists} \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

The state of interaction, intact or broken, can be defined based on the critical stretch criteria as [1,5]

$$\mu_{(k)(j)} = \begin{cases} 1 & \text{if } s_{(k)(j)} < s_c \\ 0 & \text{otherwise} \end{cases} \quad (13)$$

where s_c represents the critical stretch. This critical stretch can be evaluated for 2D bond-based PD model as [5]

$$s_c = \sqrt{\frac{\pi G_c (1 + \nu)}{3E\delta}} \quad (14)$$

where G_c represent the critical energy release rate of the material, E represents the elastic modulus and ν represents Poisson's ratio.

If a static condition with $\ddot{u}_{(k)} = \ddot{v}_{(k)} = 0$ is considered, Eqs. (11a–b) can be rewritten as

$$\bar{b}_{x(k)} = - \left(\sum_{j=1}^N C \mu_{(k)(j)} s_{(k)(j)} \cos \varphi V_{(j)} \right) \quad (15a)$$

$$\bar{b}_{y(k)} = - \left(\sum_{j=1}^N C \mu_{(k)(j)} s_{(k)(j)} \sin \varphi V_{(j)} \right). \quad (15b)$$

4 PD-based machine learning model

In this section, the PD-based machine learning models for 1D and 2D structures are presented. The displacement of one material point is expressed as a linear function of displacements of its family members in the horizon size of $\delta = 3.015\Delta x$ and the external forces applied to it. The training data are generated by using modal analysis in ANSYS. The PD-based machine learning model is obtained from the data set by using linear regression.

4.1 PD-based machine learning model for one-dimensional structures

As presented in Eq. (10), within the linear elasticity, the relation between applied body force and displacements of material points is linear. Moreover, the relationship between the displacement of a material point and its family members can also be linear. Therefore, the displacement value for a material point can be expressed as a linear function of displacements of its family members and the external force applied to it as

$$u_{(k)} = m_1 u_{(k-3)} + m_2 u_{(k-2)} + m_3 u_{(k-1)} + m_4 u_{(k+1)} + m_5 u_{(k+2)} + m_6 u_{(k+3)} + m_7 \left(\frac{F_{x(k)} \Delta x}{AE} \right) \quad (16a)$$

or

$$u_{(k)} = \mathbf{m}_{(j)} \cdot \mathbf{u}_{(j)} + m_7 \left(\frac{F_{x(k)} \Delta x}{AE} \right) \quad (16b)$$

with

$$\mathbf{m}_{(j)} = [m_1 \ m_2 \ m_3 \ m_4 \ m_5 \ m_6] \quad (16c)$$

$$\mathbf{u}_{(j)} = [u_{(k-3)} \ u_{(k-2)} \ u_{(k-1)} \ u_{(k+1)} \ u_{(k+2)} \ u_{(k+3)}] \quad (16d)$$

where $m_i, i = 1, \dots, 7$ represent coefficients that need to be determined for the ML model. The term $F_{x(k)}$ represents the axial force applied on the material point k , Δx represents mesh size in the discretized model, A represents the cross-sectional area of the 1D bar, and E represents material elastic modulus. In Eq. (16), $(k-3), \dots, (k+3)$ represent family members of the material point k as shown in Fig. 2.

Note that with the mesh size, elastic modulus and cross-sectional area presented in the last term of Eq. (16a), the coefficient m_7 is expected to be independent of these geometrical and material properties. As a result, the expected ML model can be applied for any cross-sectional area, mesh size and elastic modulus.

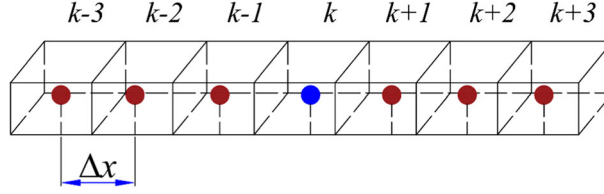


Fig. 2 A material point with its family members, $\delta = 3\Delta x$

Table 1 Arrangement for the dataset for 1D machine learning model

Dataset	Input variables							Output variable
	$u_{(k-3)}$	$u_{(k-2)}$	$u_{(k-1)}$	$u_{(k+1)}$	$u_{(k+2)}$	$u_{(k+3)}$	$\frac{F_{x(k)}\Delta x}{AE}$	
1	0	0.0817	0.1414	0.1414	0.0817	0	0.0399	0.16331
2	0	-0.1514	-0.1514	0.1514	0.1514	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
49	0.2957	-0.2812	0.2392	0.0914	0	0	-0.1808	-0.1738
50	0.2719	-0.2626	0.2355	0.1359	-0.0704	0	-0.1968	-0.1922

4.1.1 Generating the training and testing data set

In this section, the training and testing data for Eq. (16a) are obtained from the modal analysis. The data set includes seven input variables ($u_{(k-3)}$, $u_{(k-2)}$, $u_{(k-1)}$, $u_{(k+1)}$, $u_{(k+2)}$, $u_{(k+3)}$, $F_{x(k)}\Delta x/AE$) and one output variable ($u_{(k)}$).

By using Eq. (9c), the last input variable in Eq. (16a) can be represented as

$$\frac{F_{x(k)}\Delta x}{AE} = \frac{b_{x(k)}(\Delta x)^2}{E} \quad (17)$$

By utilizing the PD relation given in Eq. (10), the input variable given in Eq. (17) can be calculated as

$$\frac{F_{x(k)}\Delta x}{AE} = \frac{\left(-C_{ax} \sum_{j=1}^N \frac{u_{(j)} - u_{(k)}}{\xi} V_{(j)}\right) (\Delta x)^2}{E} \quad (18)$$

In order to generate the data set for 1D structures, a bar with a length of $L = 6$ m a cross-sectional area of $A = 0.01$ m² and the elastic modulus of $E = 2 \times 10^{11}$ N/m² is chosen. The bar is discretized with 6 elements (7 material points) with a uniform mesh size of $\Delta x = 1$ m and the LINK180 element is used by allowing only axial deformations. The bar is considered for 10 basic boundary conditions as shown in Fig. 3.

By considering all possible vibration modes of the bar for 10 boundary conditions as shown in Fig. 3, 50 data sets are obtained. The data sets are arranged as shown in Table 1. Specifically, for each vibration mode of the bar, the displacements of all nodes are obtained. The displacement of node k is added into the output variable column in Table 1. The displacements of the remaining nodes ($u_{(k-3)}$, $u_{(k-2)}$, $u_{(k-1)}$, $u_{(k+1)}$, $u_{(k+2)}$, $u_{(k+3)}$) are added into the first 6 columns in Table 1. The input variable in 7th column is calculated by using Eq. (18).

4.1.2 Finding the coefficients for the ML model

The data are split into two parts for training and testing purposes. The first 45 data sets are used for training and the remainders are used for testing the accuracy of the ML model. By using linear regression, the coefficients for the relation given in Eq. (16a) are obtained as

$$m_1 = m_6 = 0.05134596 \quad (19a)$$

$$m_2 = m_5 = 0.14955135 \quad (19b)$$

$$m_3 = m_4 = 0.29910269 \quad (19c)$$

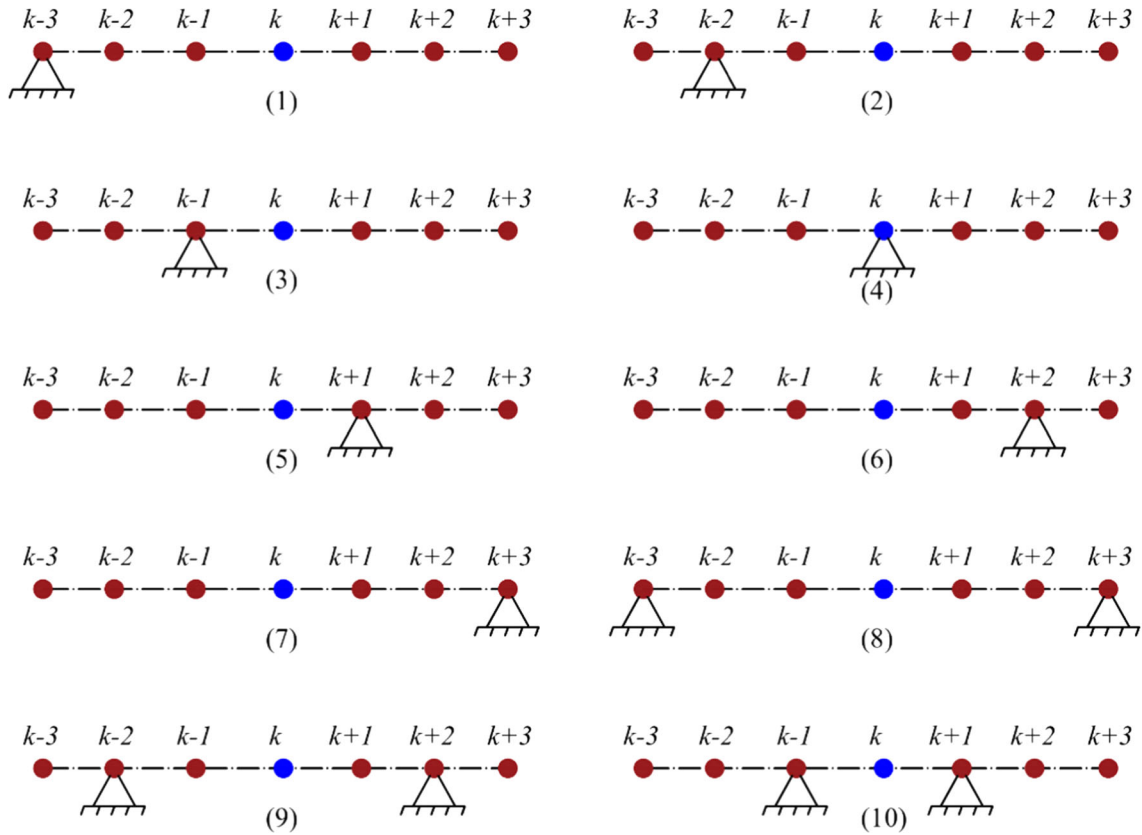


Fig. 3 Boundary conditions in modal analyses for 1D structure

$$m_7 = 1.35942174 \quad (19d)$$

In order to evaluate the accuracy of the regression model, the obtained regression model is used to predict output for the test data. By using Eq. (3), the mean-squared error between the predicted values and the original output values in the testing data is calculated as 9.09×10^{-20} .

4.1.3 Extending the PD-based ML model for dynamic problems

The machine learning model provided in Eq. (16a) can be used for static problems. The model can also be further developed by adding the inertia term for dynamic problems.

The relationship provided in Eq. (16a) can be rewritten as

$$m_1 u_{(k-3)} + m_2 u_{(k-2)} + m_3 u_{(k-1)} + m_4 u_{(k+1)} + m_5 u_{(k+2)} + m_6 u_{(k+3)} - u_{(k)} + m_7 \left(\frac{(b_x^{(k)} A \Delta x) \Delta x}{AE} \right) = 0 \quad (20a)$$

or

$$\frac{E}{\Delta x^2} \frac{1}{m_7} (m_1 u_{(k-3)} + m_2 u_{(k-2)} + m_3 u_{(k-1)} + m_4 u_{(k+1)} + m_5 u_{(k+2)} + m_6 u_{(k+3)} - u_{(k)}) + b_x^{(k)} = 0 \quad (20b)$$

Next, by adding the inertia term, Eq. (20b) can be extended for dynamic problems as

$$\rho \ddot{u}_{(k)} = \frac{E}{\Delta x^2} \frac{1}{m_7} (m_1 u_{(k-3)} + m_2 u_{(k-2)} + m_3 u_{(k-1)} + m_4 u_{(k+1)} + m_5 u_{(k+2)} + m_6 u_{(k+3)} - u_{(k)}) + b_x^{(k)} \quad (21a)$$

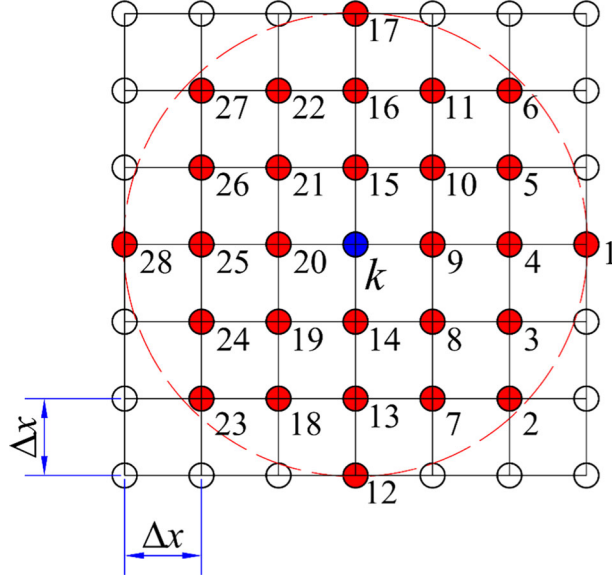


Fig. 4 PD horizon in the 2D model

or

$$\rho \ddot{u}_{(k)} = \frac{E}{\Delta x^2} \frac{1}{m_j} (\mathbf{m}_j \cdot \mathbf{u}_{(j)} - u_{(k)}) + b_{x(k)} \quad (21b)$$

where the linear regression coefficients are presented in Eq. (19).

4.2 PD-based machine learning model for two-dimensional structures

In this section, first, the machine learning model for 2D is presented. Next, training and testing data set are obtained from modal analyses by using ANSYS. The data set is then used for linear regression analysis to obtain the regression coefficients for the machine learning model.

The displacement of the material point k can be determined based on the displacements of its family members and the applied forces. In the discretized model, the material point k with a horizon size of $\delta = 3.015\Delta x$ has 28 family members that can be numbered in the order as shown in Fig. 4.

Similar to the 1D structure, the displacement of a material point k can be assumed as a linear function of the displacements of its family members and the applied forces as

$$u_{(k)} = \sum_{j=1}^{28} (a_j u_j + b_j v_j) + c \left(\frac{2(1+\nu)}{Eh} F_{x(k)} \right) + d \left(\frac{2(1+\nu)}{Eh} F_{y(k)} \right) \quad (22a)$$

$$v_{(k)} = \sum_{j=1}^{28} (m_j u_j + n_j v_j) + p \left(\frac{2(1+\nu)}{Eh} F_{x(k)} \right) + q \left(\frac{2(1+\nu)}{Eh} F_{y(k)} \right) \quad (22b)$$

or

$$u_{(k)} = \mathbf{a}_j \cdot \mathbf{u}_j + \mathbf{b}_j \cdot \mathbf{v}_j + c \left(\frac{2(1+\nu)}{Eh} F_{x(k)} \right) + d \left(\frac{2(1+\nu)}{Eh} F_{y(k)} \right) \quad (22c)$$

$$v_{(k)} = \mathbf{m}_j \cdot \mathbf{u}_j + \mathbf{n}_j \cdot \mathbf{v}_j + p \left(\frac{2(1+\nu)}{Eh} F_{x(k)} \right) + q \left(\frac{2(1+\nu)}{Eh} F_{y(k)} \right) \quad (22d)$$

with

$$\mathbf{a}_j = [a_1, a_2, \dots, a_{28}] \quad (22e)$$

$$\mathbf{b}_j = [b_1, b_2, \dots, b_{28}] \quad (22f)$$

$$\mathbf{m}_j = [m_1, m_2, \dots, m_{28}] \quad (22g)$$

$$\mathbf{n}_j = [n_1, n_2, \dots, n_{28}] \quad (22h)$$

$$\mathbf{u}_j = [u_1, u_2, \dots, u_{28}] \quad (22i)$$

$$\mathbf{v}_j = [v_1, v_2, \dots, v_{28}] \quad (22j)$$

where $a_j, b_j, m_j, n_j, c, d, p, q$ are coefficients that need to be determined for the ML model. The terms $F_{x(k)}, F_{y(k)}$ represent external forces in x, y directions that are applied on the material point k .

Note that the vectors $\mathbf{a}_j, \mathbf{b}_j, \mathbf{m}_j, \mathbf{n}_j$ store the coefficients a_j, b_j, m_j, n_j which are associated with 1st, . . . , 28th family members of the material point k . Similarly, vectors \mathbf{u}_j and \mathbf{v}_j store displacement components of 1st, . . . , 28th family members of the material point k as shown in Fig. 4.

4.2.1 Generating the training and testing data set

In this section, the training and testing data for Eq. (22) are obtained from the modal analysis. The data set includes 58 input variables and two output variables ($u_{(k)}, v_{(k)}$).

By using Eqs. (11c, d, g), the last two terms on the right-hand side of Eqs. (22c–d) can be represented as

$$\frac{2(1+\nu)}{Eh} F_{x(k)} = \frac{\bar{b}_{x(k)}}{Ch\Delta x} \frac{24}{\pi(3^3)} \quad (23a)$$

$$\frac{2(1+\nu)}{Eh} F_{y(k)} = \frac{\bar{b}_{y(k)}}{Ch\Delta x} \frac{24}{\pi(3^3)} \quad (23b)$$

where C represent the bond constant given in Eq. (11g).

By using the PD relationships given in Eq. (15), the input variables given in Eq. (23) can be calculated as

$$\frac{2(1+\nu)}{Eh} F_{x(k)} = \frac{-\left(\sum_{j=1}^N C\mu^{(k)(j)}s^{(k)(j)}\cos\varphi V_{(j)}\right)}{Ch\Delta x} \frac{24}{\pi(3^3)} \quad (24a)$$

$$\frac{2(1+\nu)}{Eh} F_{y(k)} = \frac{-\left(\sum_{j=1}^N C\mu^{(k)(j)}s^{(k)(j)}\sin\varphi V_{(j)}\right)}{Ch\Delta x} \frac{24}{\pi(3^3)} \quad (24b)$$

In order to generate data set for 2D structures, a 6×6 m² square plate is chosen as shown in Fig. 4. The plate has a thickness of 0.1 m, the elastic modulus of $E = 2 \times 10^{11}$ N/m² and Poisson's ratio of $\nu = 1/3$. The plate is discretized with a mesh size of $\Delta x = 1$ m. The material point k located at $(x = 3$ m, $y = 3$ m), shown in blue, and its 28 family members, shown in red, are considered for the data set.

Note that the dimensions of the plate, as well as the elastic modulus, are chosen arbitrarily for obtaining the data set. The ML model is expected to be applicable for any geometry and linear elastic material.

The data set is obtained from modal analyses for the plate by using the PLANE182 element in ANSYS. The plate is considered in 16 basic boundary conditions as shown in Fig. 5. In each boundary condition, the 20 possible vibration modes of the plate are considered. The displacements of the material point k and its 28 family members are obtained and added into the data set. Therefore, the data set includes 320 deformation states and it is arranged as shown in Table 2. Specifically, in each vibration mode of the plate, the displacements of all nodes are obtained. The displacements of node k (Fig. 4), ($u_{(k)}, v_{(k)}$), are added into the two last columns in Table 2 for the output variables. The displacements of the remaining nodes ($u_1, u_2, \dots, u_{28}, v_1, v_2, \dots, v_{28}$) are added for the first 56 columns in Table 2. The input variables in 57th and 58th columns are calculated by using Eq. (24).

4.2.2 Finding the coefficients for the ML model

In order to obtain the coefficients for the ML model, the obtained data set is split into two parts in which the 310 deformation states are used for training purpose and the remainders are used for testing the accuracy of

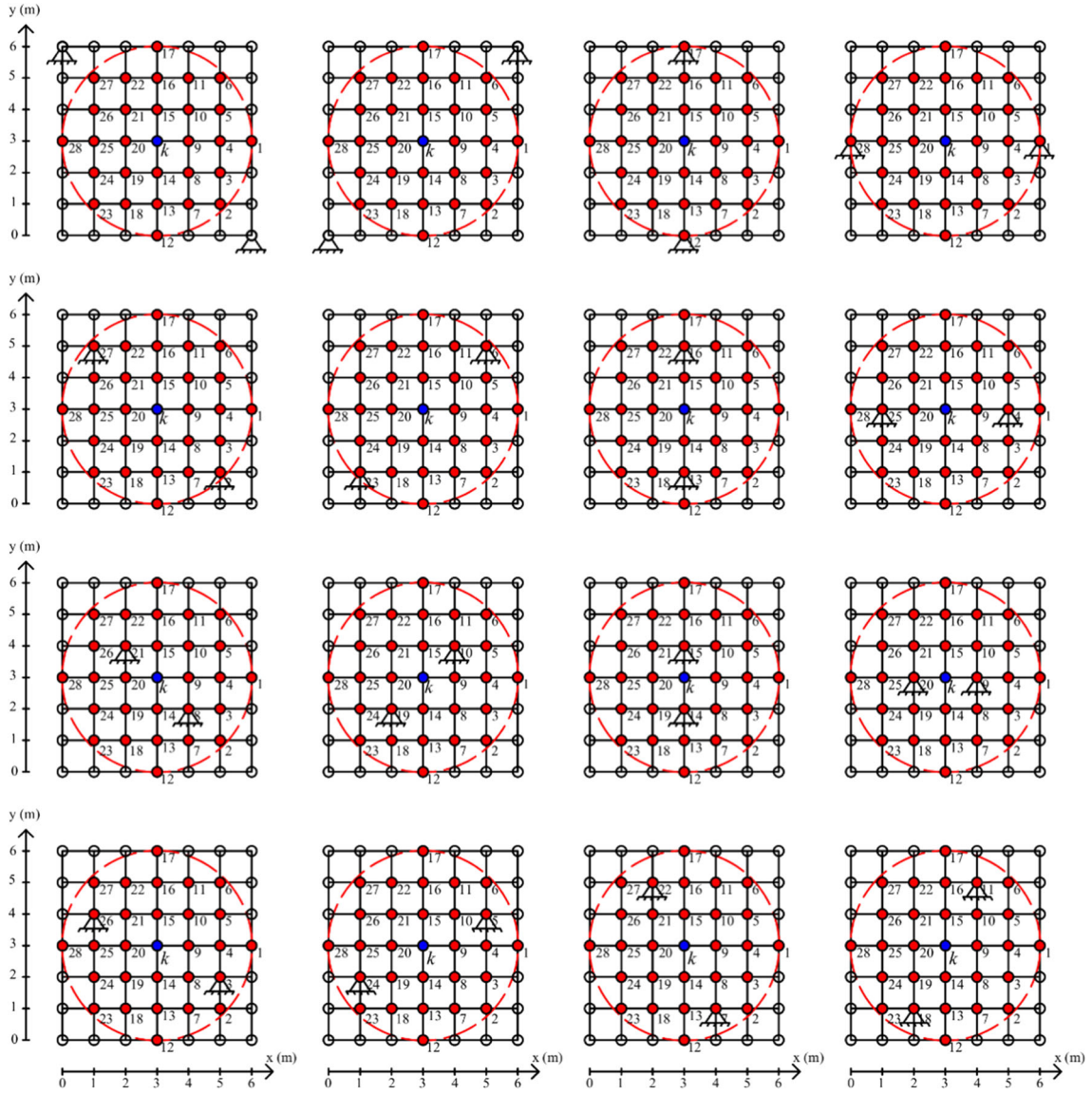


Fig. 5 Boundary conditions in modal analyses for the training model for 2D structures

Table 2 Arrangement for the dataset for 2D machine learning model

Dataset	Input variables						Output variables			
	u_1	\dots	u_{28}	v_1	\dots	v_{28}	$\frac{2(1+\nu)}{Eh} F_{x(k)}$	$\frac{2(1+\nu)}{Eh} F_{y(k)}$	$u(k)$	$v(k)$
1	-0.0014	\dots	-0.0014	0.0012	\dots	0.0012	-3.4E-05	3.4E-05	-0.0015	0.0015
2	0.0014	\dots	0.0014	0.0012	\dots	0.0012	3.4E-05	3.4E-05	0.0015	0.0015
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
319	-0.0023	\dots	0.0023	0.0011	\dots	-0.0011	0	0	0	0
320	-0.0024	\dots	-0.0024	0.0004	\dots	0.0004	-0.0006	-0.0008	-0.0005	-0.0007

the ML model. By using the training data, the coefficients of the linear relation given in Eq. (22) are obtained by using linear regression as

$$\begin{aligned} \mathbf{a}_j &= [a_1, a_2, \dots, a_{28}] \\ &= [2.44126487 \times 10^{-2}, 1.72599524 \times 10^{-2}, 5.08783725 \times 10^{-2}, 7.11047181 \times 10^{-2}, \\ &\quad 5.08783714 \times 10^{-2}, 1.72599441 \times 10^{-2}, 1.27196452 \times 10^{-2}, 5.02786632 \times 10^{-2}, \\ &\quad 1.42209416 \times 10^{-1}, 5.02786703 \times 10^{-2}, 1.27196600 \times 10^{-2}, -1.46844059 \times 10^{-8}, \\ &\quad -4.0248502 \times 10^{-8}, 3.72502411 \times 10^{-10}, -3.46583395 \times 10^{-9}, -4.29021655 \times 10^{-8}, \\ &\quad -1.4904684 \times 10^{-8}, 1.27196564 \times 10^{-2}, 5.02786706 \times 10^{-2}, 1.42209409 \times 10^{-1}, \\ &\quad 5.02786670 \times 10^{-2}, 1.27196465 \times 10^{-2}, 1.72599458 \times 10^{-2}, 5.08783714 \times 10^{-2}, \\ &\quad 7.11047206 \times 10^{-2}, 5.08783737 \times 10^{-2}, 1.72599507 \times 10^{-2}, 2.44126477 \times 10^{-2}] \end{aligned} \quad (25a)$$

$$\begin{aligned} \mathbf{b}_j &= [b_1, b_2, \dots, b_{28}] \\ &= [-5.00827301 \times 10^{-9}, -1.72599315 \times 10^{-2}, -2.54392470 \times 10^{-2}, 8.08215284 \times 10^{-9}, \\ &\quad 2.54392437 \times 10^{-2}, 1.72599372 \times 10^{-2}, -2.54391797 \times 10^{-2}, -5.02786804 \times 10^{-2}, \\ &\quad -9.15180233 \times 10^{-9}, 5.02786809 \times 10^{-2}, 2.54391712 \times 10^{-2}, 7.52227191 \times 10^{-9}, \\ &\quad 1.07005713 \times 10^{-10}, 3.63822413 \times 10^{-9}, 4.78426226 \times 10^{-9}, -1.99145690 \times 10^{-9}, \\ &\quad 9.09132078 \times 10^{-9}, 2.54391711 \times 10^{-2}, 5.02786812 \times 10^{-2}, -1.08530537 \times 10^{-8}, \\ &\quad -5.02786775 \times 10^{-2}, -2.54391817 \times 10^{-2}, 1.72599362 \times 10^{-2}, 2.54392469 \times 10^{-2}, \\ &\quad 5.87137186 \times 10^{-9}, -2.54392448 \times 10^{-2}, -1.72599313 \times 10^{-2}, -6.71179578 \times 10^{-9}] \end{aligned} \quad (25b)$$

$$[c \ d] = [4.42141045 \times 10^{-1}, 3.40539888 \times 10^{-9}] \quad (25c)$$

$$\begin{aligned} \mathbf{m}_j &= [m_1, m_2, \dots, m_{28}] \\ &= [-1.01969796 \times 10^{-9}, -1.72599441 \times 10^{-2}, -2.54391664 \times 10^{-2}, -3.12205277 \times 10^{-8}, \\ &\quad 2.54391866 \times 10^{-2}, 1.72599234 \times 10^{-2}, -2.54392424 \times 10^{-2}, -5.02786808 \times 10^{-2}, \\ &\quad 3.25989664 \times 10^{-8}, 5.02786841 \times 10^{-2}, 2.54392514 \times 10^{-2}, -5.28682972 \times 10^{-10}, \\ &\quad -4.53987184 \times 10^{-9}, -9.22966016 \times 10^{-9}, -1.53501892 \times 10^{-8}, -8.77134865 \times 10^{-9}, \\ &\quad -8.79935283 \times 10^{-10}, 2.54392457 \times 10^{-2}, 5.02786845 \times 10^{-2}, 2.19821744 \times 10^{-8}, \\ &\quad -5.02786748 \times 10^{-2}, -2.54392403 \times 10^{-2}, 1.72599262 \times 10^{-2}, 2.54391866 \times 10^{-2}, \\ &\quad -2.71851724 \times 10^{-8}, -2.54391645 \times 10^{-2}, -1.72599468 \times 10^{-2}, -2.58653592 \times 10^{-9}] \end{aligned} \quad (25d)$$

$$\begin{aligned} \mathbf{n}_j &= [n_1, n_2, \dots, n_{28}] \\ &= [-1.14255206 \times 10^{-8}, 1.72599501 \times 10^{-2}, 1.27196621 \times 10^{-2}, -2.96128749 \times 10^{-8}, \\ &\quad 1.27196573 \times 10^{-2}, 1.72599359 \times 10^{-2}, 5.08783489 \times 10^{-2}, 5.02787379 \times 10^{-2}, \\ &\quad -1.67365554 \times 10^{-7}, 5.02787515 \times 10^{-2}, 5.08783560 \times 10^{-2}, 2.44126537 \times 10^{-2}, \\ &\quad 7.11046716 \times 10^{-2}, 1.42209499 \times 10^{-1}, 1.42209501 \times 10^{-1}, 7.11046683 \times 10^{-2}, \\ &\quad 2.44126562 \times 10^{-2}, 5.08783559 \times 10^{-2}, 5.02787519 \times 10^{-2}, -1.70078338 \times 10^{-7}, \\ &\quad 5.02787426 \times 10^{-2}, 5.08783456 \times 10^{-2}, 1.72599343 \times 10^{-2}, 1.27196623 \times 10^{-2}, \\ &\quad -3.31381429 \times 10^{-8}, 1.27196654 \times 10^{-2}, 1.72599505 \times 10^{-2}, -1.41419238 \times 10^{-8}] \end{aligned} \quad (25e)$$

$$[p \ q] = [-2.54221850 \times 10^{-8}, 4.42141016 \times 10^{-1}] \quad (25f)$$

Similar to the 1D model, by using the testing data, the mean-squared error between the predicted values and the original values from testing data is 6.42×10^{-23} .

4.2.3 Extending the PD-based ML model for dynamic problems

As given in Eq. (25), the coefficients $d_{(k)}$ and $p_{(k)}$ are very small which can be ignored as

$$d \approx 0; \quad p \approx 0 \quad (26)$$

Therefore, Eq. (22) can be rewritten as

$$u_{(k)} = \mathbf{a}_j \cdot \mathbf{u}_j + \mathbf{b}_j \cdot \mathbf{v}_j + c \left(\frac{2(1+\nu)}{Eh} F_{x(k)} \right) \quad (27a)$$

$$v_{(k)} = \mathbf{m}_j \cdot \mathbf{u}_j + \mathbf{n}_j \cdot \mathbf{v}_j + q \left(\frac{2(1+\nu)}{Eh} F_{y(k)} \right) \quad (27b)$$

or

$$\frac{1}{c} [\mathbf{a}_j \cdot \mathbf{u}_j + \mathbf{b}_j \cdot \mathbf{v}_j - u_{(k)}] + \bar{b}_{x(k)} \frac{2(1+\nu)}{Eh} \Delta x^2 = 0 \quad (27c)$$

$$\frac{1}{q} [\mathbf{m}_j \cdot \mathbf{u}_j + \mathbf{n}_j \cdot \mathbf{v}_j - v_{(k)}] + \bar{b}_{y(k)} \frac{2(1+\nu)}{Eh} \Delta x^2 = 0 \quad (27d)$$

or

$$\frac{Eh}{2(1+\nu)\Delta x^2} \frac{1}{c} [\mathbf{a}_j \cdot \mathbf{u}_j + \mathbf{b}_j \cdot \mathbf{v}_j - u_{(k)}] + \bar{b}_{x(k)} = 0 \quad (27e)$$

$$\frac{Eh}{2(1+\nu)\Delta x^2} \frac{1}{q} [\mathbf{m}_j \cdot \mathbf{u}_j + \mathbf{n}_j \cdot \mathbf{v}_j - v_{(k)}] + \bar{b}_{y(k)} = 0 \quad (27f)$$

Therefore, by reintroducing the inertia forces on the left-hand side of Eqs. (27e–f), the machine learning model can be extended for dynamic problems as

$$\rho h \ddot{u}_{(k)} = \frac{Eh}{2(1+\nu)\Delta x^2} \frac{1}{c} [\mathbf{a}_j \cdot \mathbf{u}_j + \mathbf{b}_j \cdot \mathbf{v}_j - u_{(k)}] + \bar{b}_{x(k)} \quad (28a)$$

$$\rho h \ddot{v}_{(k)} = \frac{Eh}{2(1+\nu)\Delta x^2} \frac{1}{q} [\mathbf{m}_j \cdot \mathbf{u}_j + \mathbf{n}_j \cdot \mathbf{v}_j - v_{(k)}] + \bar{b}_{y(k)}. \quad (28b)$$

5 Numerical implementation

As given in Sect. 4, the PD-based ML models for 1D and 2D structures are obtained for material points with full interactions with their family members. Specifically, the 1D ML model is applicable for material points with 6 interactions and the 2D ML model is applicable for material points with 28 interactions. However, for material points that have some missing family members or broken interactions such as material points near boundary surfaces or near crack surfaces, the developed ML models can produce significant errors. Moreover, generating training data for all of these special cases can be very time-consuming. Therefore, a hybrid approach that couples the ML model with the PD model is used. The behaviors of the material points with full interactions are predicted by using the ML model. Meanwhile, all other material points are predicted using the PD model.

Similar to the conventional PD solution, the deformation behavior of structures can be obtained by using a meshless scheme. The domain is divided into a uniform mesh. In this study, for static and quasi-static loading conditions, the adaptive dynamic relaxation (ADR) method [60,61] is used. For dynamic problems, the explicit time integration scheme [5] is used. The numerical procedure for the 1D structure is shown in Fig. 6 for static problems and Fig. 7 for dynamic problems. The numerical procedure for a 2D structure is shown in Fig. 8 for static problems and Fig. 9 for dynamic problems.

As shown in Fig. 6 for 1D static problems, at each ADR iteration, first, the boundary and loading conditions are applied. Next, the PD and ML regions for the discretized model are defined. Specifically, the PD region is defined by material points with less than 6 interactions (near boundary regions). Meanwhile, the ML region is defined by material points with 6 interactions (regions that are far from boundaries). After defining the regions, by using the displacement values of the PD family members, the PD forces for each PD material points are calculated. Next, by solving the PD equations of motion given in Eq. (9) as shown in Loop 2, new displacements of PD material points are calculated. Then, based on the calculated displacement values of PD regions, the displacements of material points in ML regions are obtained by using the displacement values of its ML family members according to Eq. (16) as shown in Loop 2.2.

Similar to static problems, for 1D dynamic problems for each time step, first the boundary and loading conditions are applied as shown in Fig. 7. Next, the PD and ML regions for the discretized model are defined

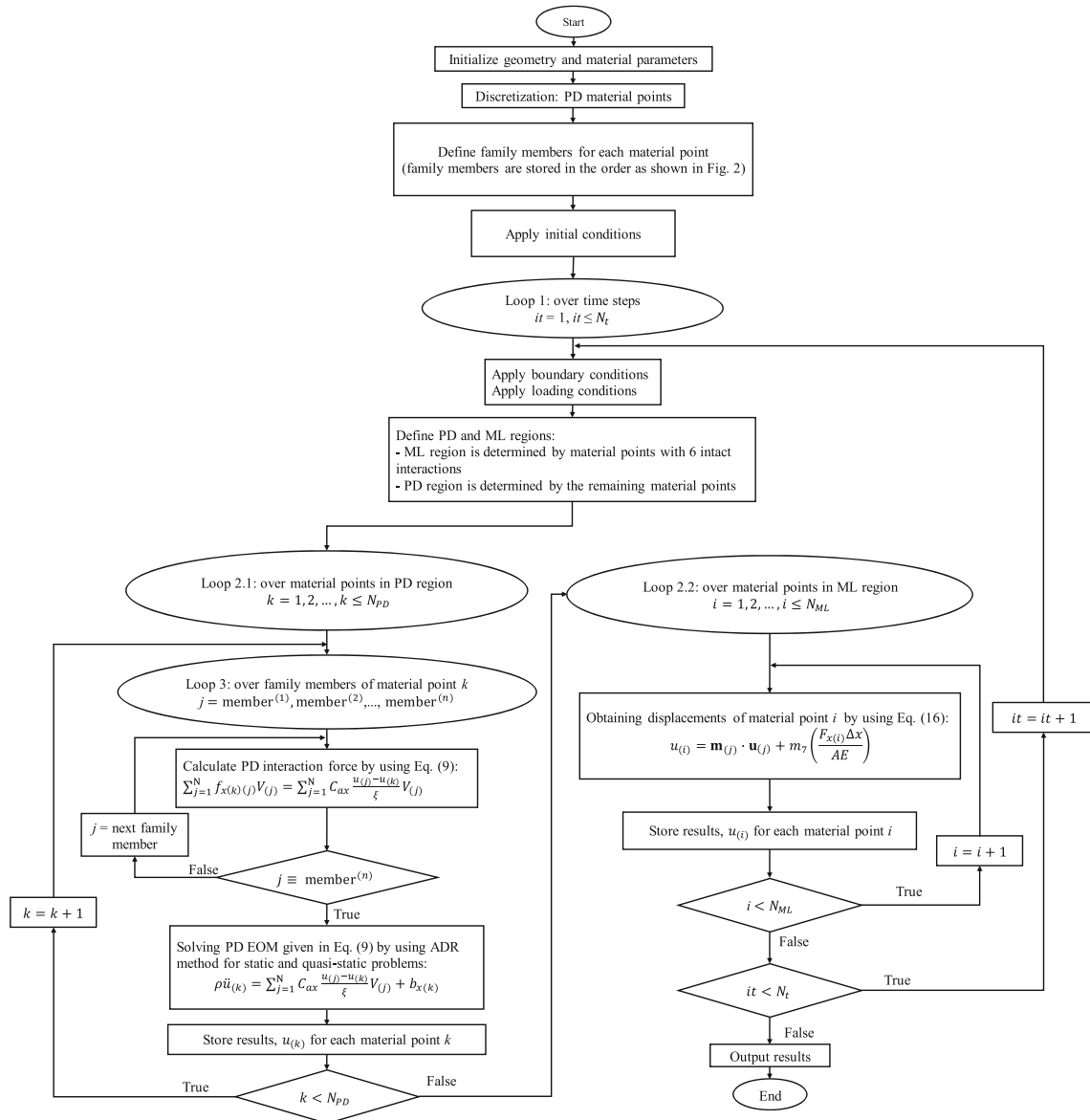


Fig. 6 Numerical procedure for the coupled approach in 1D static and quasi-static problems (N_{PD} and N_{ML} represent the number of material points in PD and ML regions, respectively. ADR, adaptive dynamic relaxation method [60,61])

by using the same method used for the 1D static problems. After defining the regions, the force densities for all material points, including PD and ML regions, are calculated. If a material point belongs to the PD regions, the PD force densities for that material point are obtained by using the displacement values of its PD family members according to Eq. (9) as

$$F_{x(k)}^{PD} = \sum_{j=1}^{N_{PD}} f_{x(k)(j)} V_{(j)} = C_{ax} \sum_{j=1}^{N_{PD}} \frac{u_{(j)} - u_{(k)}}{\xi} V_{(j)} \quad (29)$$

Otherwise, the PD force densities for that material point are approximated by using the displacement values of its ML family members according to Eq. (21b) as

$$F_{x(k)}^{ML} = \sum_{j=1}^{N_{ML}} f_{x(k)(j)} V_{(j)} = \frac{E}{\Delta x^2} \frac{1}{m_7} (\mathbf{m}_j \cdot \mathbf{u}_{(j)} - u_{(k)}) \quad (30)$$

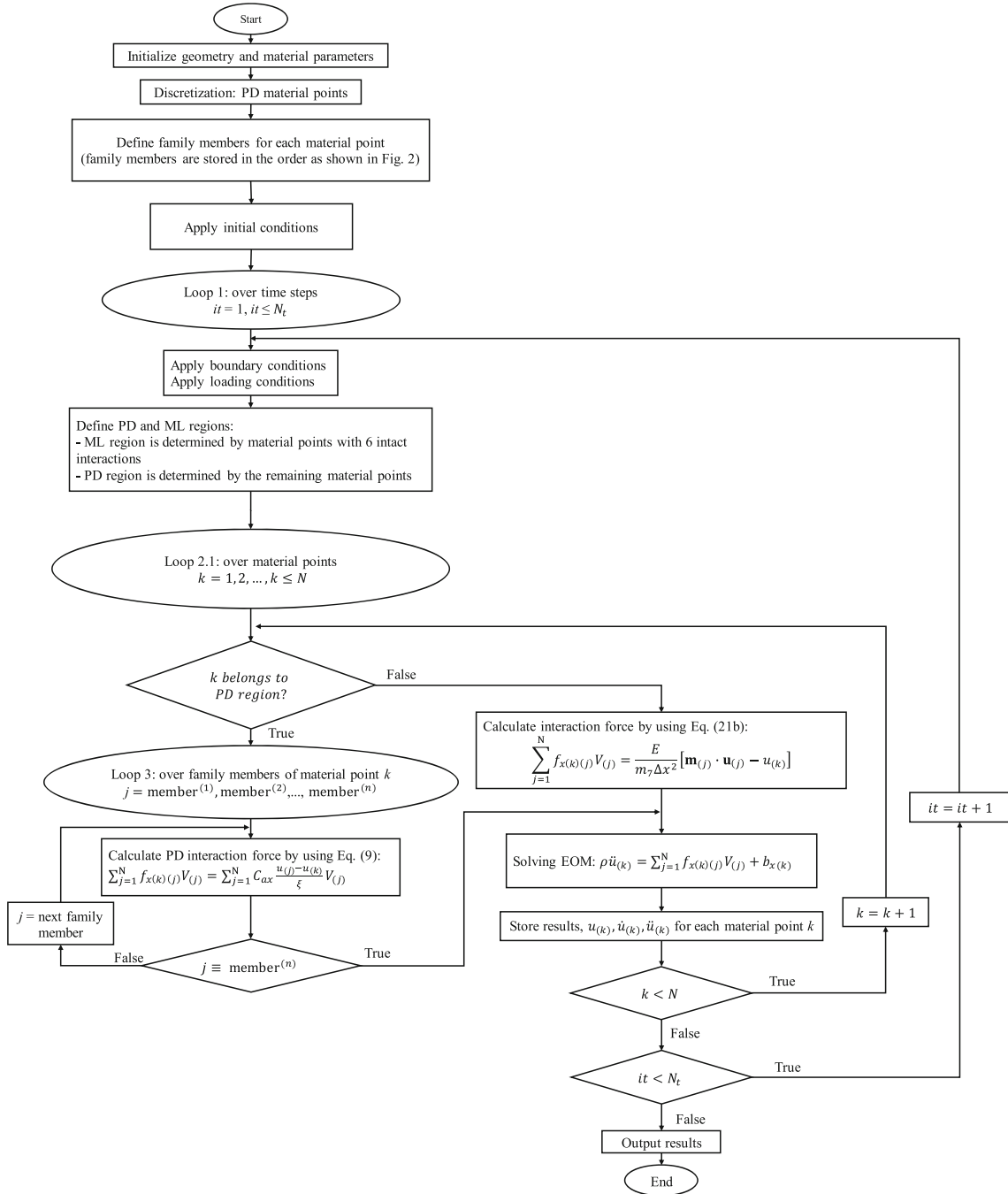


Fig. 7 Numerical procedure for the coupled approach in 1D dynamic problems

After the force densities for all nodes are obtained, the displacements, velocities and accelerations of material points are obtained by solving the 1D equation of motion as

$$\rho \ddot{u}_{(k)} = F_{x(k)}^{\text{PD}} + b_{x(k)} \quad (31a)$$

or

$$\rho \ddot{u}_{(k)} = F_{x(k)}^{\text{ML}} + b_{x(k)} \quad (31b)$$

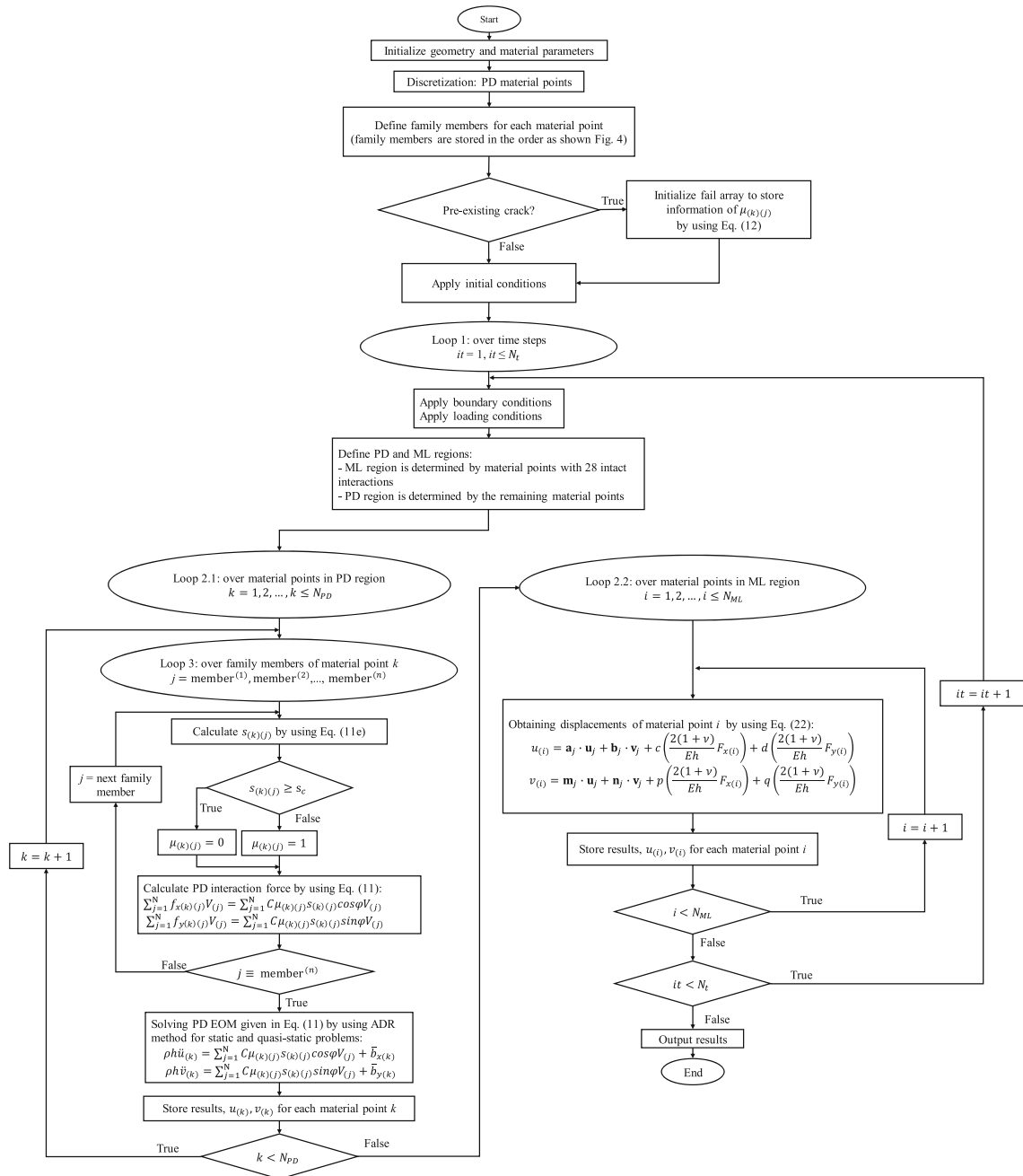


Fig. 8 Numerical procedure for the coupled approach in 2D static and quasi-static problems (N_{PD} and N_{ML} represent the number of material points in PD and ML regions, respectively. ADR, adaptive dynamic relaxation method [60,61])

Similar to 1D static case, Fig. 8 presents the algorithm for 2D static problems. In this case, PD region is defined by material points with less than 28 intact interactions (near boundaries and crack surfaces). Meanwhile, the ML region is defined by material points with 28 intact interactions (regions that are far from boundaries and crack surfaces). The displacements of material points in PD regions are obtained by solving the PD equations of motion given in Eq. (11) as shown in Loop 2.1, and the displacements of material points in ML regions are obtained by using the linear relations given in Eq. (22) as shown in Loop 2.2.

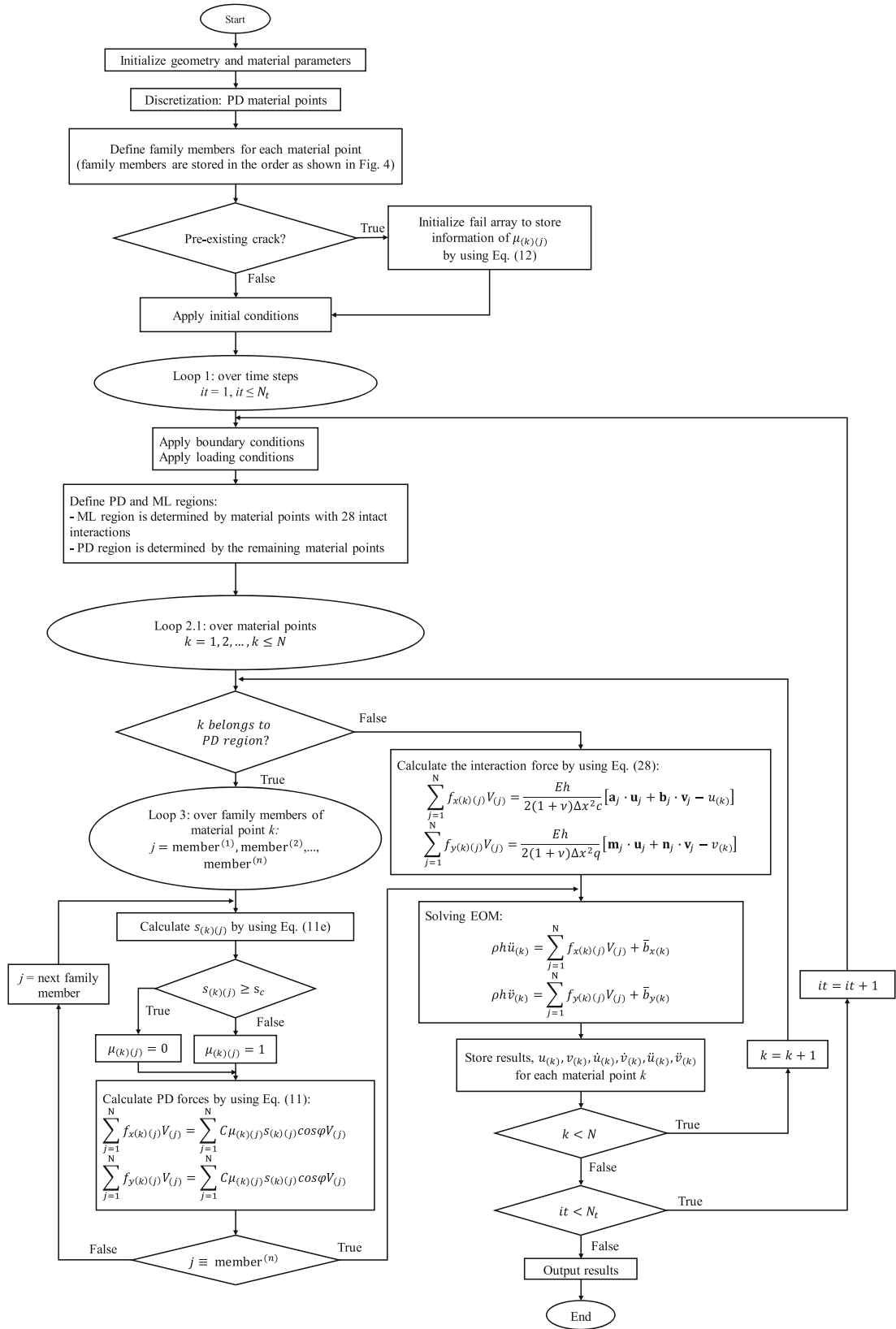


Fig. 9 Numerical procedure for coupled approach in 2D dynamic problems

Similar to 1D dynamic case, Fig. 9 presents the algorithm for 2D dynamic problems. The force densities for all material points, including PD and ML regions, are calculated according to Eq. (11) or Eq. (28) as

$$F_{x(k)}^{\text{PD}} = \sum_{j=1}^{N^{\text{PD}}} f_{x(k)(j)} V_{(j)} = \sum_{j=1}^{N^{\text{PD}}} C \mu_{(k)(j)} s_{(k)(j)} \cos \varphi V_{(j)} \quad (32a)$$

$$F_{y(k)}^{\text{PD}} = \sum_{j=1}^{N^{\text{PD}}} f_{y(k)(j)} V_{(j)} = \sum_{j=1}^{N^{\text{PD}}} C \mu_{(k)(j)} s_{(k)(j)} \sin \varphi V_{(j)} \quad (32b)$$

or

$$F_{x(k)}^{\text{ML}} = \sum_{j=1}^{N^{\text{ML}}} f_{x(k)(j)} V_{(j)} = \frac{Eh}{2(1+\nu)} \frac{1}{\Delta x^2} \frac{1}{c} [\mathbf{a}_j \cdot \mathbf{u}_j + \mathbf{b}_j \cdot \mathbf{v}_j - u_{(k)}] \quad (33a)$$

$$F_{y(k)}^{\text{ML}} = \sum_{j=1}^{N^{\text{ML}}} f_{y(k)(j)} V_{(j)} = \frac{Eh}{2(1+\nu)} \frac{1}{\Delta x^2} \frac{1}{q} [\mathbf{m}_j \cdot \mathbf{u}_j + \mathbf{n}_j \cdot \mathbf{v}_j - v_{(k)}] \quad (33b)$$

After the force densities for all nodes are obtained, the displacements, velocities and accelerations of material points are obtained by solving the 2D equation of motion as

$$\rho h \ddot{u}_{(k)} = F_{x(k)}^{\text{PD}} + \bar{b}_{x(k)} \quad (34a)$$

$$\rho h \ddot{v}_{(k)} = F_{y(k)}^{\text{PD}} + \bar{b}_{y(k)} \quad (34b)$$

or

$$\rho h \ddot{u}_{(k)} = F_{x(k)}^{\text{ML}} + \bar{b}_{x(k)} \quad (35a)$$

$$\rho h \ddot{v}_{(k)} = F_{y(k)}^{\text{ML}} + \bar{b}_{y(k)} \quad (35b)$$

6 Numerical results

In this section, first, the PD-based machine learning models are verified by considering various examples of 1D and 2D structures. As presented in Sect. 5, a hybrid approach for coupling the machine learning models and bond-based PD models is used. The results obtained by the coupled approach are compared to PD and finite element analysis (FEA) solutions. The FEA solutions are conducted by using ANSYS commercial software with the LINK180 element for the 1D bar and PLANE182 element for the 2D plate. To further verify the capabilities of the coupled approach, damage predictions on a plate with preexisting crack subjected to tension, on a 2D representation of a three-point bending test and on plate subjected to dynamic loading are performed.

6.1 Verification for 1D model

In this section, a 1D structure subjected to different loading conditions is investigated. The displacements of four material points on the left end and four material points on the right end of the bar are obtained by solving the PD equation of motion. On the other hand, the displacements of the remaining material points are obtained by using the machine learning model.

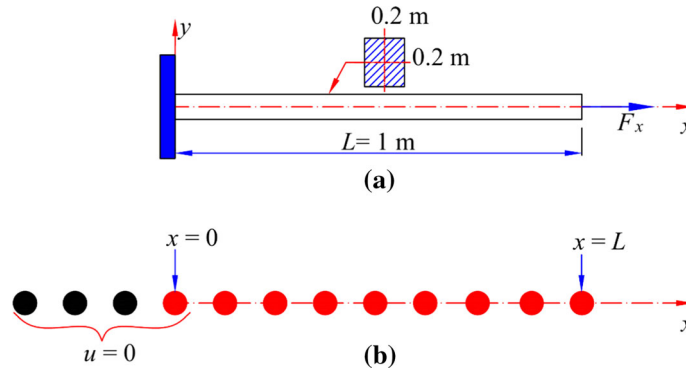


Fig. 10 Bar subjected to axial loading **a** geometry, **b** model discretization

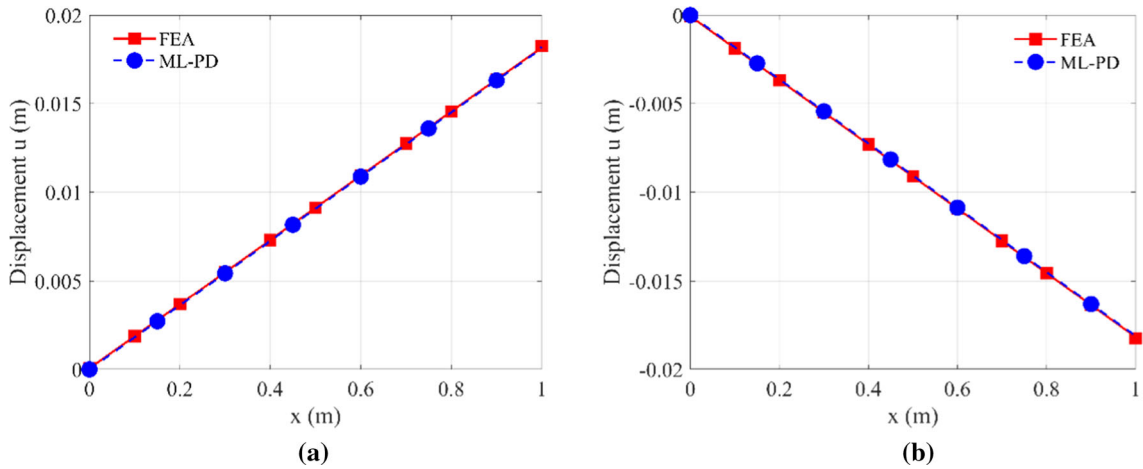


Fig. 11 Displacement variation along the bar subjected to an axial force **a** $F_x = 5 \times 10^7$ N, **b** $F_x = -5 \times 10^7$ N

6.1.1 Bar subjected to simple axial loading

A bar with a cross-sectional area $A = 0.2 \times 0.2 \text{ m}^2$ and a length of $L = 1 \text{ m}$ is investigated as shown in Fig. 10. The bar has an elastic modulus of $E = 69 \times 10^9 \text{ N/m}^2$. The bar is fixed on the left end and it is subjected to two different loading conditions which are the tensile load of $F_x = 5 \times 10^7 \text{ N}$ and the compressive load of $F_x = -5 \times 10^7 \text{ N}$.

The bar is discretized with uniform 100 integration points. To implement the boundary condition, three fictitious points [62,63] are added on the left end of the bar as shown in Fig. 10b. Displacements of three fictitious points as well as the displacement of the material point located at $x = 0$ are set equal to zero. Therefore, as shown in Fig. 10b, the first four material points on the left end of the bar are subjected to the zero displacement condition. In Fig. 10b, the red points represent the material points in the real region, and black points represent the material points in the fictitious region. In the FEA model, the bar is discretized with 100 elements by using the LINK180 element.

Figure 11 shows the displacement variations along the bar for two loading conditions. As can be seen from the figures, the results captured by the coupled machine learning and PD models match very well with the results from FEA.

6.1.2 Bar subjected multiple axial loading

For further verification, another bar with different geometrical and material properties is investigated as shown in Fig. 12. The bar has a cross-sectional area of $A = 0.02 \text{ m}^2$ and length of $L = 2 \text{ m}$ and elastic modulus of $E = 3.8 \times 10^9 \text{ N/m}^2$. The bar is subjected to forces $F_3 = 5 \times 10^6 \text{ N}$ at the right end, $F_1 = 0.4F_3$ at $x_1 = 0.6 \text{ m}$ and $F_2 = 0.5F_3$ at $x_2 = 1.4 \text{ m}$.

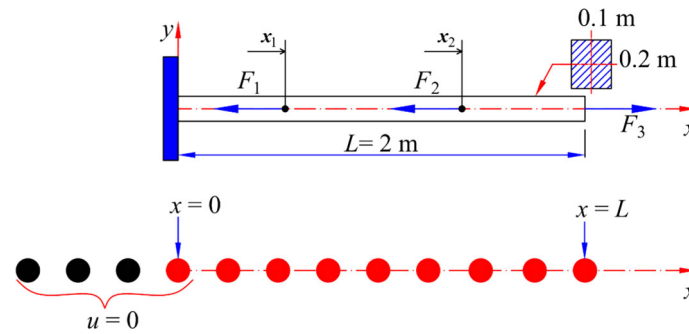


Fig. 12 Bar subjected to multiple axial loading

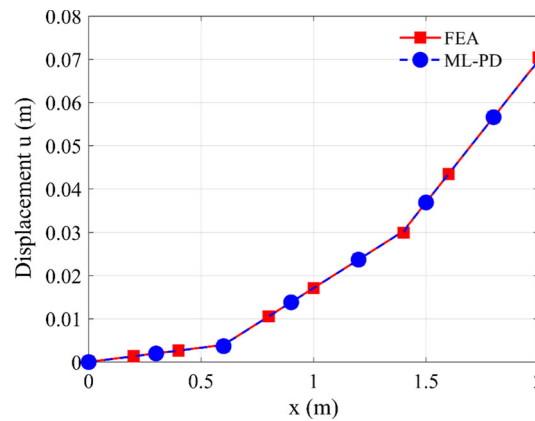


Fig. 13 Displacement variation along the bar length

The bar is discretized with uniform 200 integration points. Figure 13 shows the displacement variation along the bar. As can be seen from the figure, the results captured by the coupled model match very well with the FEA results.

6.1.3 Bar vibration

In this example, the vibration of a bar with a length of $L = 1$ m, a cross-sectional area of $A = 0.05$ m² is investigated. The bar has an elastic modulus of $E = 2 \times 10^{11}$ N/m², and it is discretized with a mesh size of $\Delta x = 0.005$ m. Initially, the bar is subjected to a displacement gradient of $\partial u / \partial x = 0.1$. Later, the bar is left to freely vibrate; meanwhile, the left end is fixed [5].

Figure 14 shows the displacement variation of a material point located at $x = L/2$. As can be seen from the figure, the results captured by the coupled model match very well with the results in both PD and FEA. Therefore, the accuracy of the coupled ML and PD models is verified.

6.2 Verification for 2D model

In this section, the coupled 2D model is verified by considering various examples of 2D structures with different geometrical and material properties.

6.2.1 Plate subjected to tension

A square plate with dimensions of $L = W = 1$ m and thickness of $h = 0.01$ m is investigated as shown in Fig. 15a. The plate has an elastic modulus of $E = 69 \times 10^9$ N/m² and Poisson's ratio of $\nu = 1/3$, and it is subjected to tensional force per unit length of $f_x = 2 \times 10^8$ N/m.

The plate is discretized uniformly with a mesh size of $\Delta x = L/100$. Similar to 1D problems, the plate is discretized with two regions as shown in Fig. 15b. The PD regions, shown in red, include three layers of

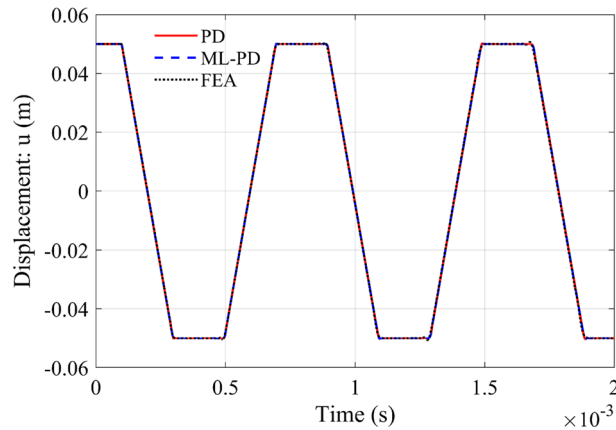


Fig. 14 Variation of displacement $u(m)$ of the material located at $x = L/2$

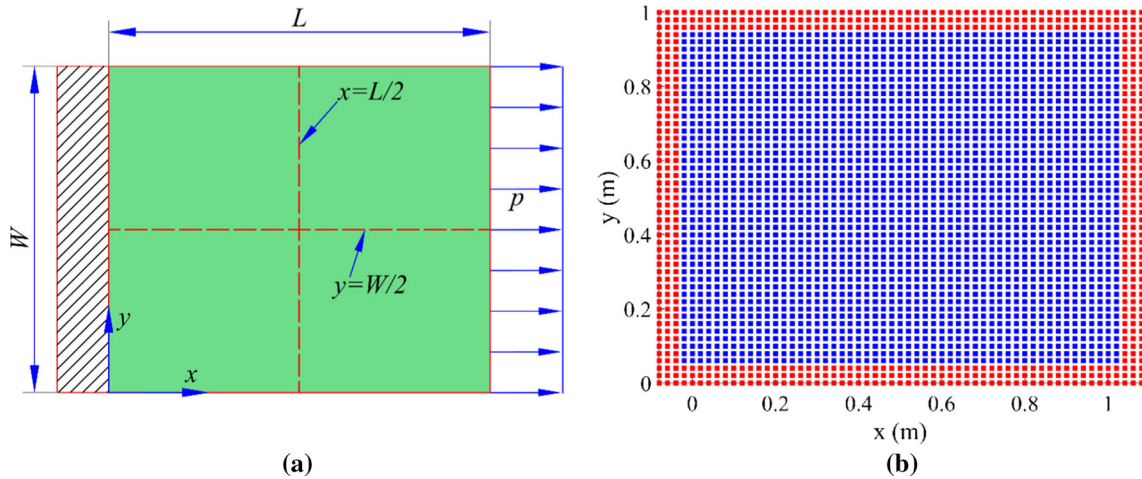


Fig. 15 Plate subjected to axial loading **a** geometry, **b** model discretization (PD regions are shown in red, and ML regions are shown in blue)

material points on four edges of the plate. The displacements of these points are obtained by solving the PD equations of motion. Since the effects of near-surface boundaries are significant in the PD models, the surface correction [5] is adopted for the PD regions. The remaining material points, shown in blue, belong to the ML region. The displacements of these material points are obtained by using the machine learning model. The numerical procedure for the coupled approach is described in Sect. 5.

Figures 16 and 17 present the variations of displacement components of the plate. Figure 18 shows variations of displacement components along two centerlines $x = L/2$ and $y = W/2$. As can be seen from the figures, the results captured by the coupled model match very well with FEA results.

Figure 19 shows the comparison of the computational time for the bond-based PD model and the coupled approach. The relationships between run-time per time step and the total number of material points in the discretized models are considered. In the PD solution, the adaptive dynamic relaxation (ADR) method is used [60,61]. In the coupled solution, the ADR method is also used for the PD region. As can be seen from the figure, the solution by using coupled ML and PD models requires less computational time for each time step than the conventional PD solution. Therefore, by using the coupled ML and PD models, the computational cost can be reduced.

6.2.2 Plate with square cutout subjected to tension

In this section, a square plate with dimensions of $L = W = 0.05$ m and thickness of $h = 0.0005$ m is investigated as shown in Fig. 20a. The plate has 0.01×0.01 m² a square cutout in the middle, and it is subjected

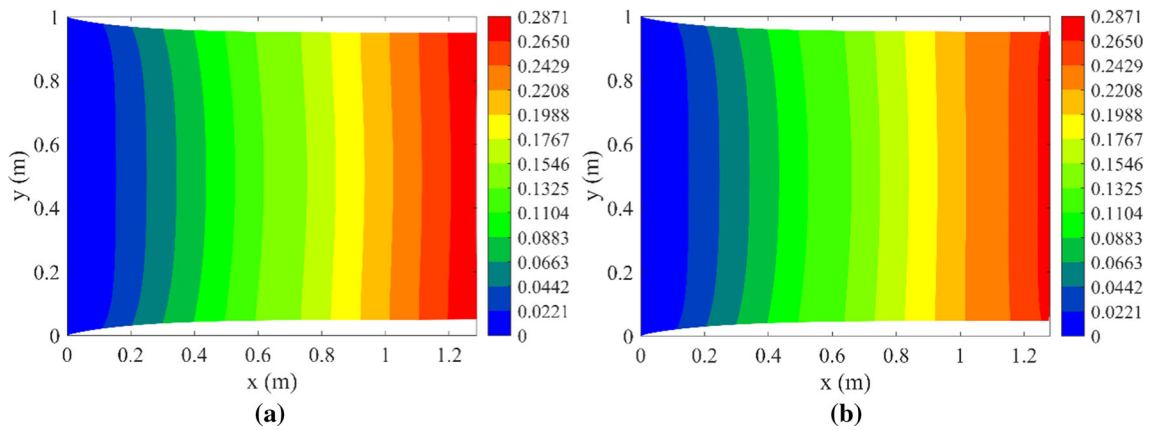


Fig. 16 Variation of displacement $u(m)$ on the plate captured by **a** FEA, **b** coupled ML and PD models

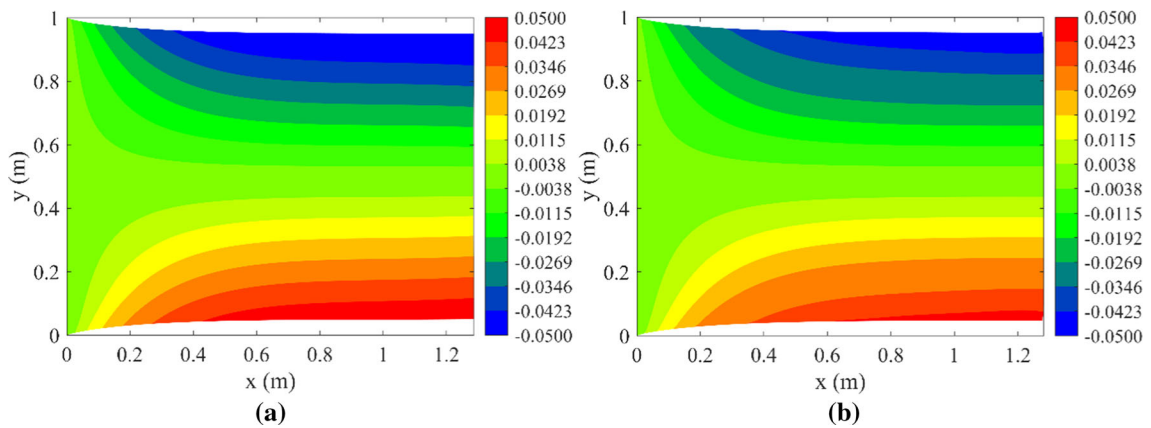


Fig. 17 Variation of displacement $v(m)$ on the plate captured by **a** FEA, **b** coupled ML and PD models

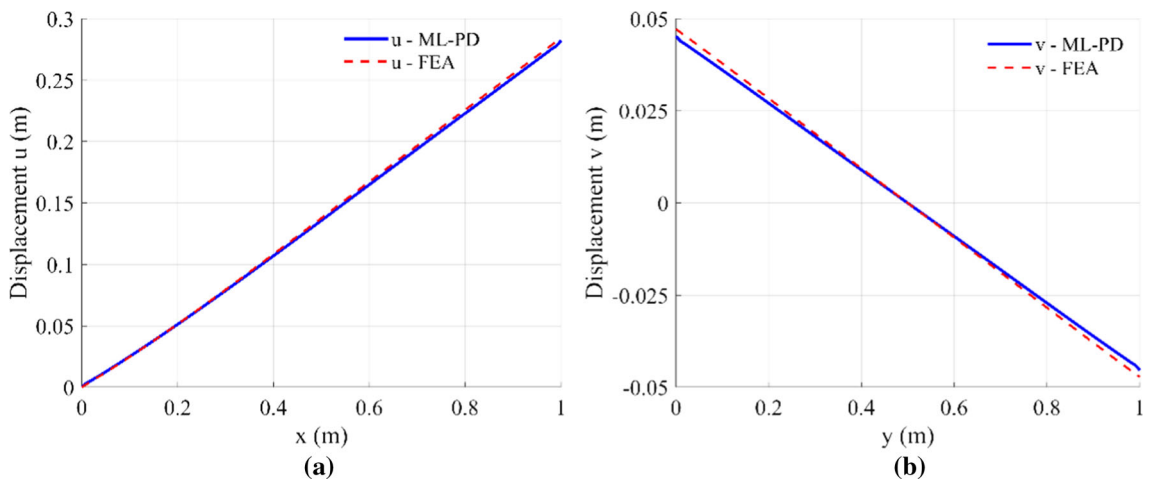


Fig. 18 Variations of displacements **a** u along $y = W/2$; **b** v along $x = L/2$ (ML-PD: coupled ML and PD models)

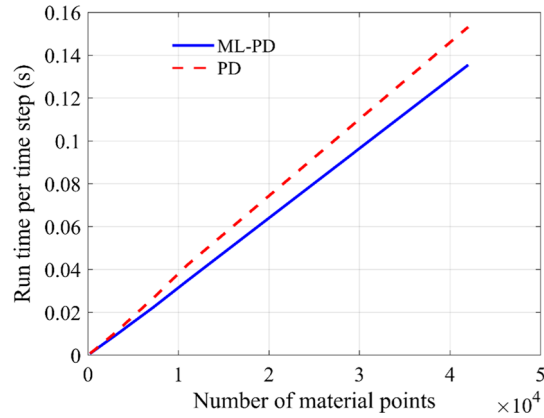


Fig. 19 Run-time (s) per time step versus the number of material points

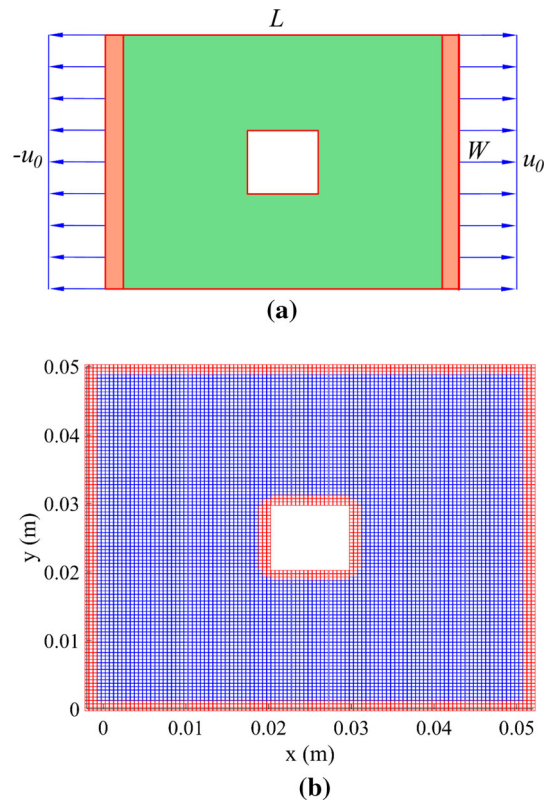


Fig. 20 Plate with square cutout subjected to axial loading **a** geometry, **b** model discretization (PD regions are shown in red, and ML regions are shown in blue)

to loading condition of $u_0 = \pm 0.005$ m at two ends. The plate has an elastic modulus of $E = 192 \times 10^9$ N/m² and Poisson's ratio of $\nu = 1/3$. The plate is discretized uniformly with a mesh size of $\Delta x = L/100$. As shown in Fig. 20b, the PD regions are shown in red and the ML regions are shown in blue. Similar to the previous example, the surface correction [5] is adopted for the PD regions.

Figures 21 and 22 show the displacement fields on the plate. Figure 23 shows variations of displacement components along two centerlines $x = L/2$ and $y = W/2$. As can be seen from the figures, the results captured by using coupled ML and PD models and the FEA have a good agreement which shows the accuracy of the coupled model for 2D.

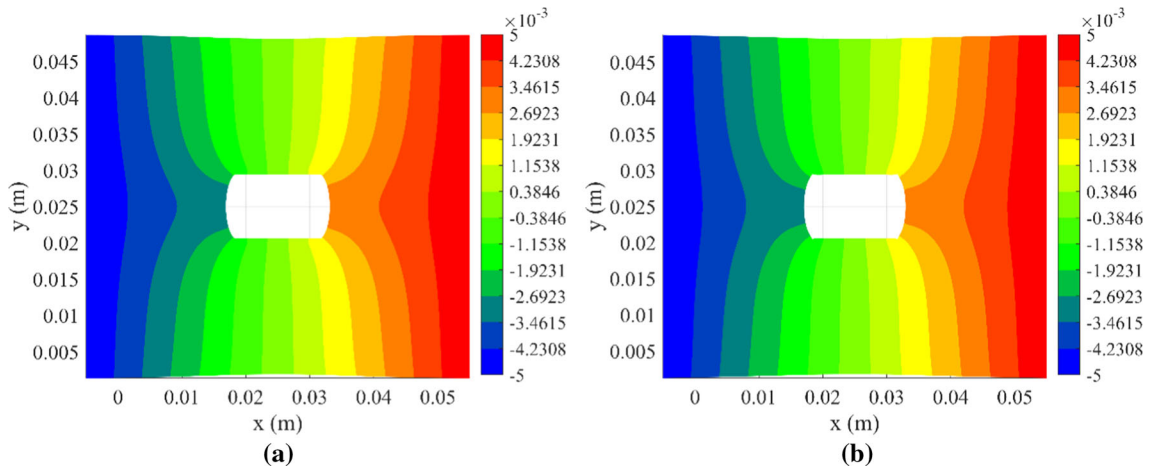


Fig. 21 Variation of displacement u (m) on the plate captured by **a** FEA, **b** coupled ML and PD models

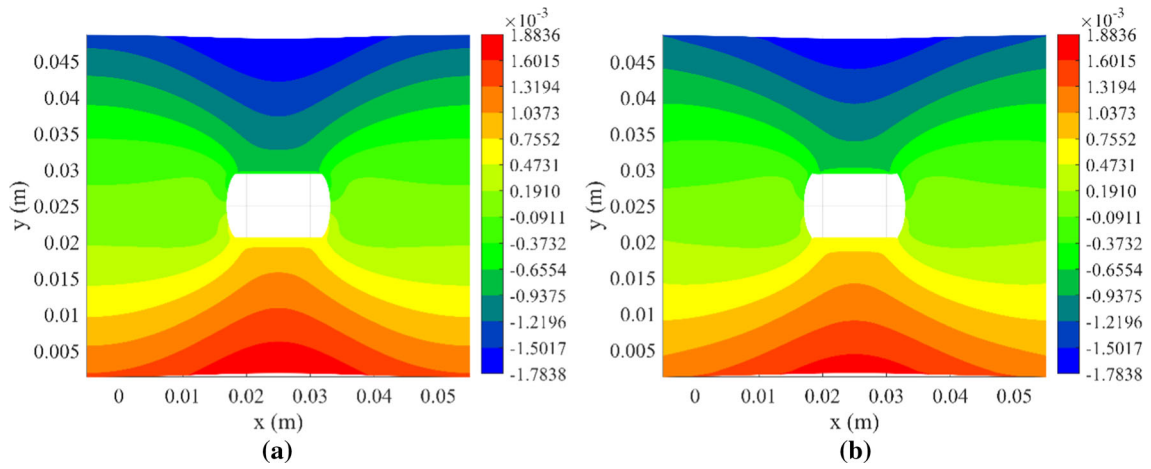


Fig. 22 Variation of displacement v (m) on the plate captured by **a** FEA, **b** coupled ML and PD models

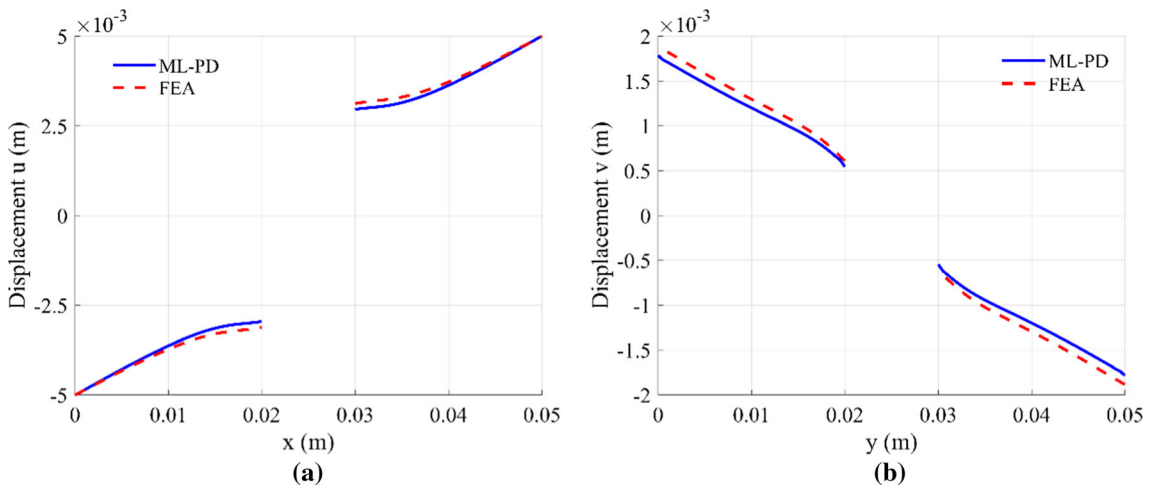


Fig. 23 Variations of displacements **a** u along $y = W/2$; **b** v along $x = L/2$ (ML-PD: coupled ML and PD models)

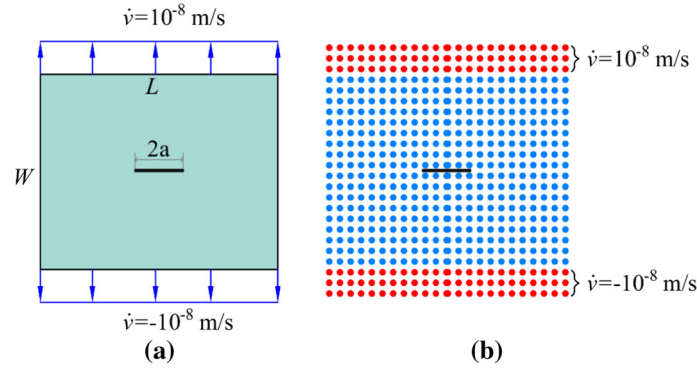


Fig. 24 Plate with preexisting crack subjected to tensional loading

6.3 Damage predictions

After verifying the accuracy of the hybrid approach by coupling of ML and PD models, in this section, damage predictions for 2D plates are presented. In order to properly capture the behavior of the structures with progressive damages, the PD regions and ML regions are updated adaptively. At each time step, material points with 28 intact interactions are updated and the behaviors of these material points are obtained by using the 2D ML model. On the other hand, the behaviors of material points with less than 28 intact interactions, which are either near boundary surfaces or near crack surfaces, are obtained by using PD solution. Similar to the previous examples, the surface correction [5] is adopted for the PD regions near boundary surfaces.

6.3.1 Plate with preexisting crack subjected to tension

In this example, a plate with dimensions of $L \times W = 0.5 \times 0.806 \text{ m}^2$ and thickness of $h = 0.01 \text{ m}$ is investigated as shown in Fig. 24 [64]. The plate has a preexisting crack with a length of $2a = 0.1 \text{ m}$ in the middle. The material properties are represented by the elastic modulus of $E = 2.16 \times 10^{11} \text{ N/m}^2$, Poisson's ratio of $\nu = 1/3$, fracture toughness $K_c = 70 \times 10^6 \text{ MPa}\sqrt{\text{m}}$ [65], and the critical energy release rate of $G_c = 2.1233 \times 10^4 \text{ J/m}^2$. The critical stretch can be calculated by using the relation given in Eq. (26) as $s_c = 0.0021$.

The plate is subjected to incremental displacements on top and bottom by $|\Delta v| = 10^{-8} \text{ m}$ per each time step as shown in Fig. 24. In the numerical model, the plate is discretized with a mesh size of $\Delta x = L/150$ and quasi-static solution described in Fig. 8 is used. As shown in Fig. 24b, in order to apply loading conditions, three layers of material points, shown in red, are added on top and bottom of the plate and incremental displacements are applied to these material points as

$${}^n v_{(\text{top})} = {}^{n-1} v_{(\text{top})} + 10^{-8} dt \text{ (m)} \quad (36a)$$

$${}^n v_{(\text{bot})} = {}^{n-1} v_{(\text{bot})} - 10^{-8} dt \text{ (m)} \quad (36b)$$

$${}^0 v_{(\text{bot})} = {}^0 v_{(\text{top})} = 0 \quad (36c)$$

where ${}^n v_{(\text{top})}$ and ${}^n v_{(\text{bot})}$ represent displacements at the current time step of material points on the top and bottom boundaries, respectively. The terms ${}^{n-1} v_{(\text{top})}$ and ${}^{n-1} v_{(\text{bot})}$ represent displacements at the previous time step of material points on the top and bottom boundaries, respectively. The time step size is chosen as $dt = 1 \text{ s}$.

Figure 25 shows the damage evolution on the plate. As shown in Fig. 25a, the crack starts propagating when the applied displacements equal to $3.5 \times 10^{-4} \text{ m}$. As the applied displacements increase, the crack propagates horizontally as expected and it nearly reaches two sides of the plate when applied displacements are $5.3 \times 10^{-4} \text{ m}$. This observation has good agreement with the experimental results captured by .Simonsen, Törnqvist [64].

Figure 26 shows the machine learning and PD regions at different load steps. As can be seen from the figure, the PD and ML regions are adaptively updated as damage progresses. All material points with less than 28 intact interactions are determined at each load step and they are defined as PD region. The remaining material

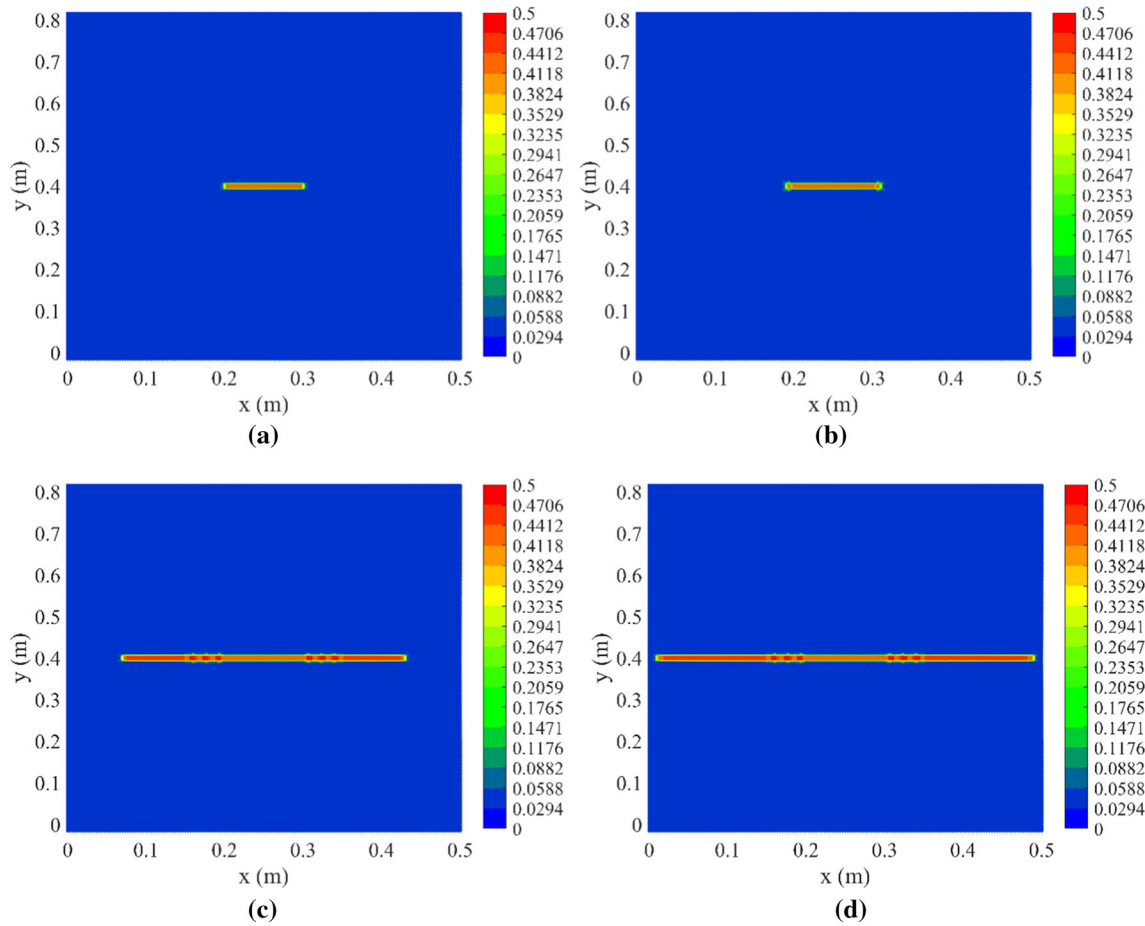


Fig. 25 Damage evolution on the plate when the applied displacement equals to **a** 3.5×10^{-4} m, **b** 4.5×10^{-4} m, **c** 5×10^{-4} m, **d** 5.3×10^{-4} m

points, shown in blue, are defined as the ML region. As shown in Fig. 26, there are a small number of material points, shown in red, that need to use PD solution. As shown in Fig. 26d, when the applied displacement equals to 5.3×10^{-4} m, the PD region includes 1602 material points; meanwhile, the total number of material points is 37750. Therefore, the computational cost for the simulation can be reduced by using this coupled approach.

6.3.2 Three-point bending test

In this example, a 2D representation of a three-point bending test for a concrete beam conducted by Jenq [66] is investigated as shown in Fig. 27. The dimensions of the plate are defined by $S = 0.3048$ m and $H = 0.0702$ m. The plate has an initial crack located at $X = 0.25S$ with a crack length of $a = 0.5H$. The material properties are represented by the elastic modulus of $E = 30 \times 10^9$ N/m², Poisson's ratio of $\nu = 1/4$, and the critical energy release rate $G_c = 20.7368$ J/m² [66]. The critical stretch can be calculated by using the relation given in Eq. 19) as $s_c = 4.4376 \times 10^{-4}$.

The plate is discretized with uniform 301×58 material points. The quasi-static loading is applied by increasing the displacement by $\Delta w = -10^{-9}$ at $x = 0.6S$, $y = H$ for each load step. A zero vertical displacement $v = 0$ is applied at $x = 0.1S$, $y = 0$ and $x = 1.1S$, $y = 0$. The numerical procedure given in Fig. 8 is used for this quasi-static problem.

Figure 28 shows the damage evolution on the plate. As expected, the crack propagates towards the middle position of the specimen. As shown in Fig. 28b, the angle between the crack path and vertical axis is approximate 35° which shows good agreement with the experimental result [66]. Figure 29 shows the adaptive PD and ML regions at different load steps. Similar to the previous example, the PD and ML regions are automatically updated based on the progressive damages on the structure. As shown in Fig. 29c, when the applied displacement

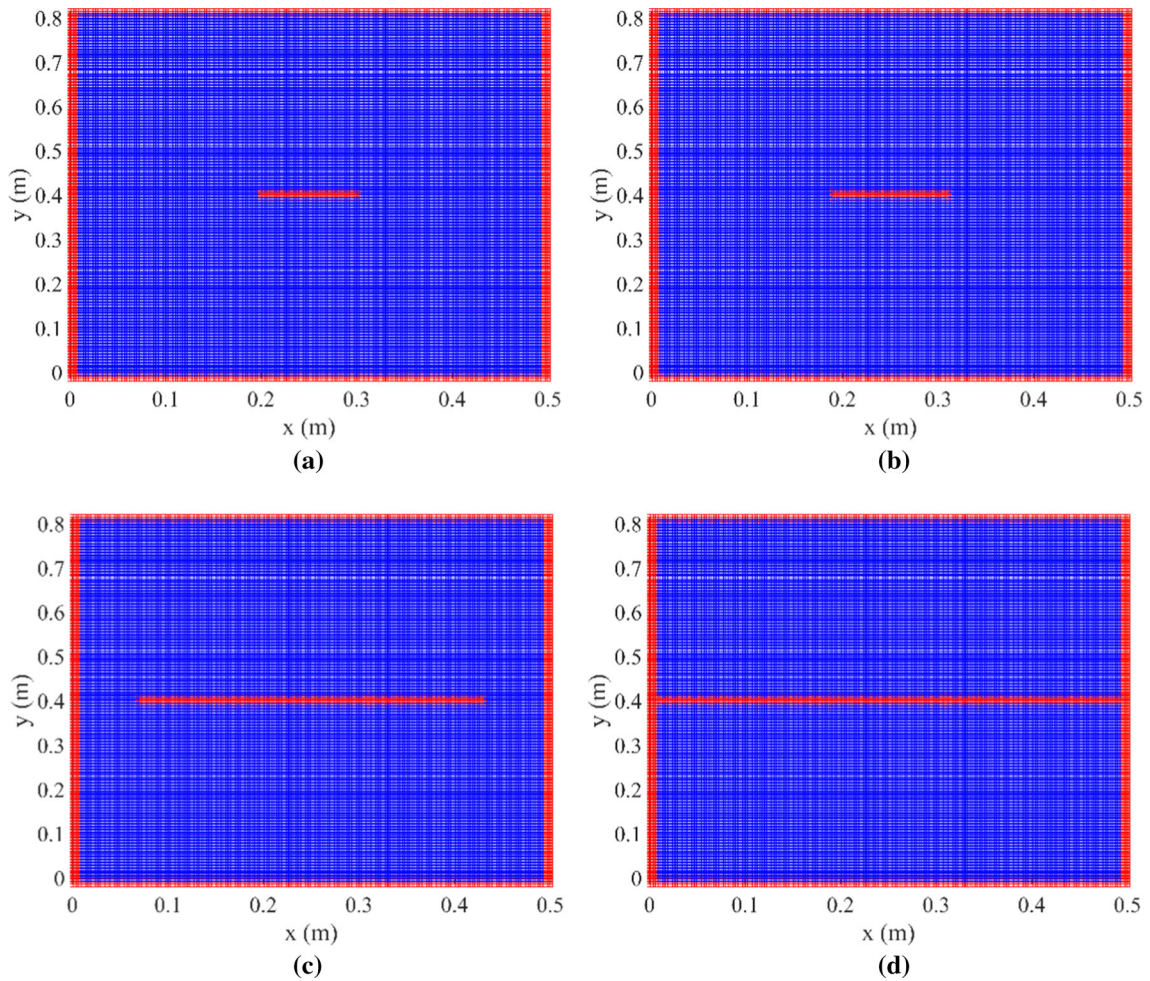


Fig. 26 Adaptive machine learning and PD regions when the applied displacement equals to **a** 3.5×10^{-4} m, **b** 4.5×10^{-4} m, **c** 5×10^{-4} m, **d** 5.3×10^{-4} m (PD regions are shown in red and ML regions are shown in blue)

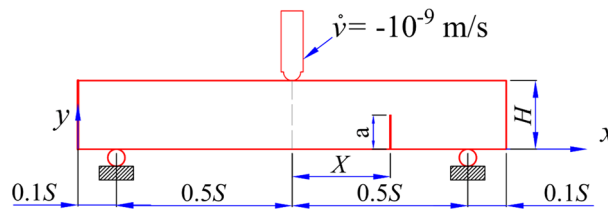


Fig. 27 Three-point bending problem

equals to 14×10^{-5} m, the number of material points in the PD region is 2523, which is 14.5% of the total number of material points in the discretized model, 17458.

6.3.3 Kalthoff experiment

In this section, the experiment presented by Kalthoff and Winkler [67], Kalthoff [65,68] for a prenotched plate subjected to dynamic load is simulated by using the coupled ML and PD models. Since the problem is symmetric, only the upper half plate with dimensions of $L = W = 0.1$ m and thickness of $h = 0.009$ m is modeled as shown in Fig. 30. The plate is made of steel with the elastic modulus of $E = 2 \times 10^{11}$ N/m², Poisson's ratio of $\nu = 1/3$. The fracture toughness of steel is $K_{Ic} = 70 \times 10^6$ Nm^{-3/2} [65], and the critical energy release rate is $G_c = 2.2714 \times 10^4$ J/m². The critical stretch can be calculated by using the relation given

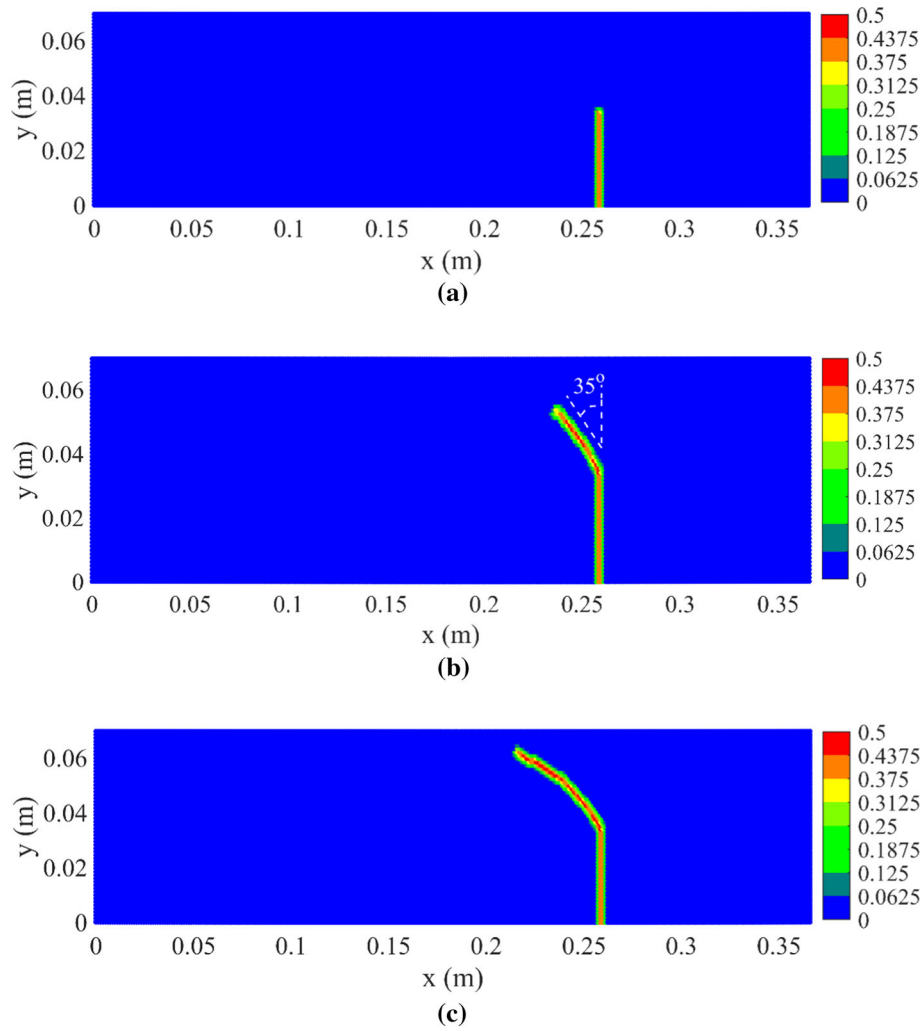


Fig. 28 Damage evolution on the beam when the applied displacement equals to **a** 8×10^{-5} m, **b** 11×10^{-5} m, **c** 14×10^{-5} m

in Eq. (14) as $s_c = 0.0089$. The left edge which is under the crack surface is subjected to velocity conditions as [69]

$$|\mathbf{v}| = \begin{cases} \frac{t}{t_0} v_0 & \text{for } t \leq t_0 \\ v_0 & \text{for } t > t_0 \end{cases} \quad (37)$$

with $v_0 = 16.5$ m/s, $t_0 = 1 \mu\text{s}$.

The plate is discretized into 150×150 material points, and the solution results are obtained by using a dynamic explicit time integration scheme with a time step of $0.01 \mu\text{s}$ and total simulation time of $80 \mu\text{s}$. Similar to the previous examples, the solution results are obtained by solving Eq. (11) for the PD region and by using Eq. (28) for the ML region.

Figure 31 presents the damage evolution on the plate. As can be seen from the figure, under dynamic loading conditions, the crack propagates up 67.3° orientation with respect to the horizontal axis. After $80 \mu\text{s}$, the crack propagates nearly to the top edge of the plate as shown in Fig. 31d. As can be seen from the figure, the crack paths captured by coupled ML and PD models match very well with the experimental results in [65,67,68]. Figure 32 shows the PD and ML regions at different time steps. As shown in Fig. 32d for the coupling model at $80 \mu\text{s}$, 12.68% of the total number of material points (2909 per 22950 material points) belong to the PD region.

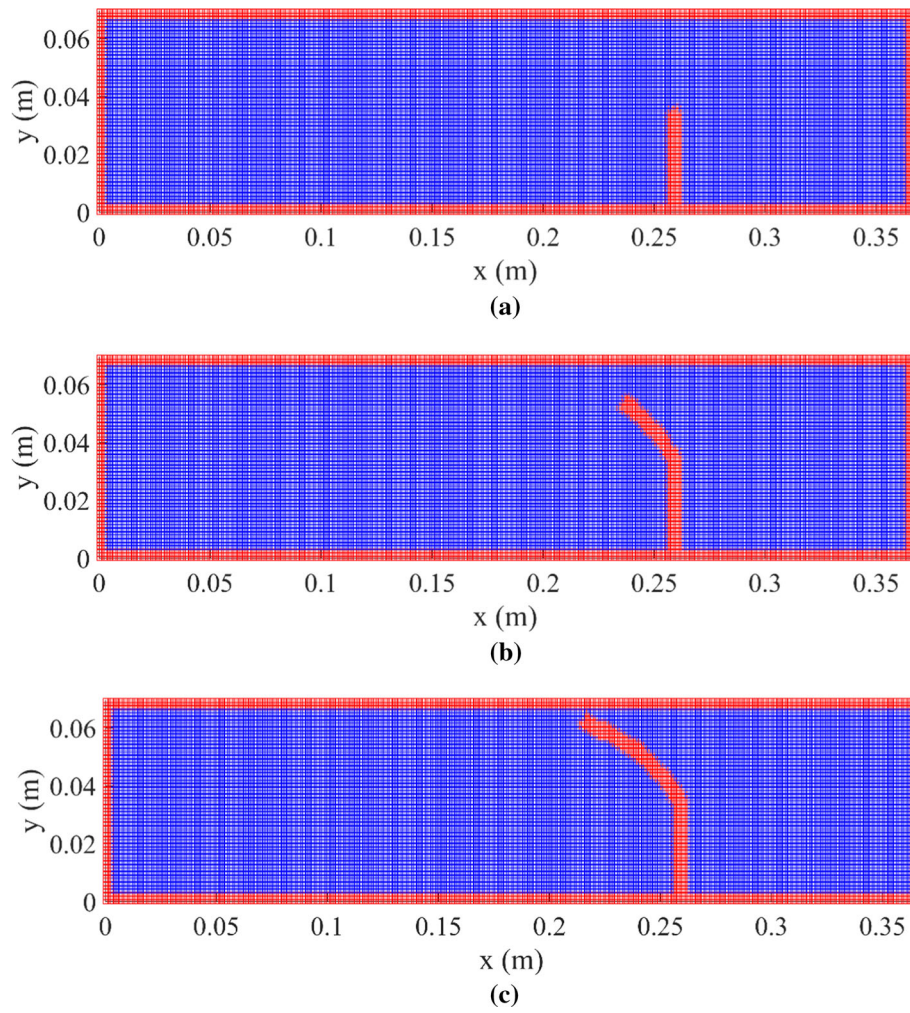


Fig. 29 Adaptive machine learning and PD regions when the applied displacement equals to **a** 8×10^{-5} m, **b** 11×10^{-5} m, **c** 14×10^{-5} m (PD regions are shown in red, and ML regions are shown in blue)

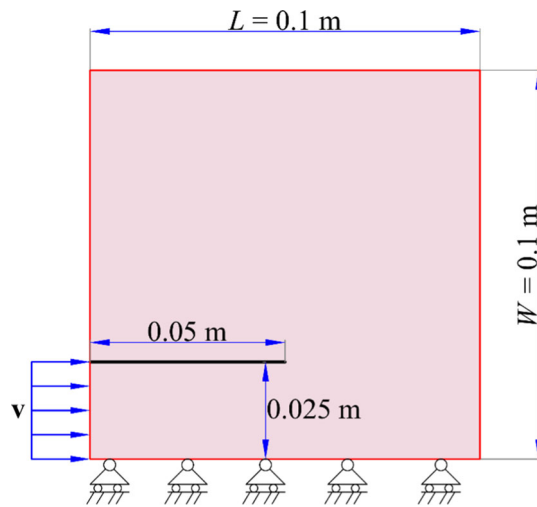


Fig. 30 The geometry and symmetrical boundary conditions for Kalthoff experiment

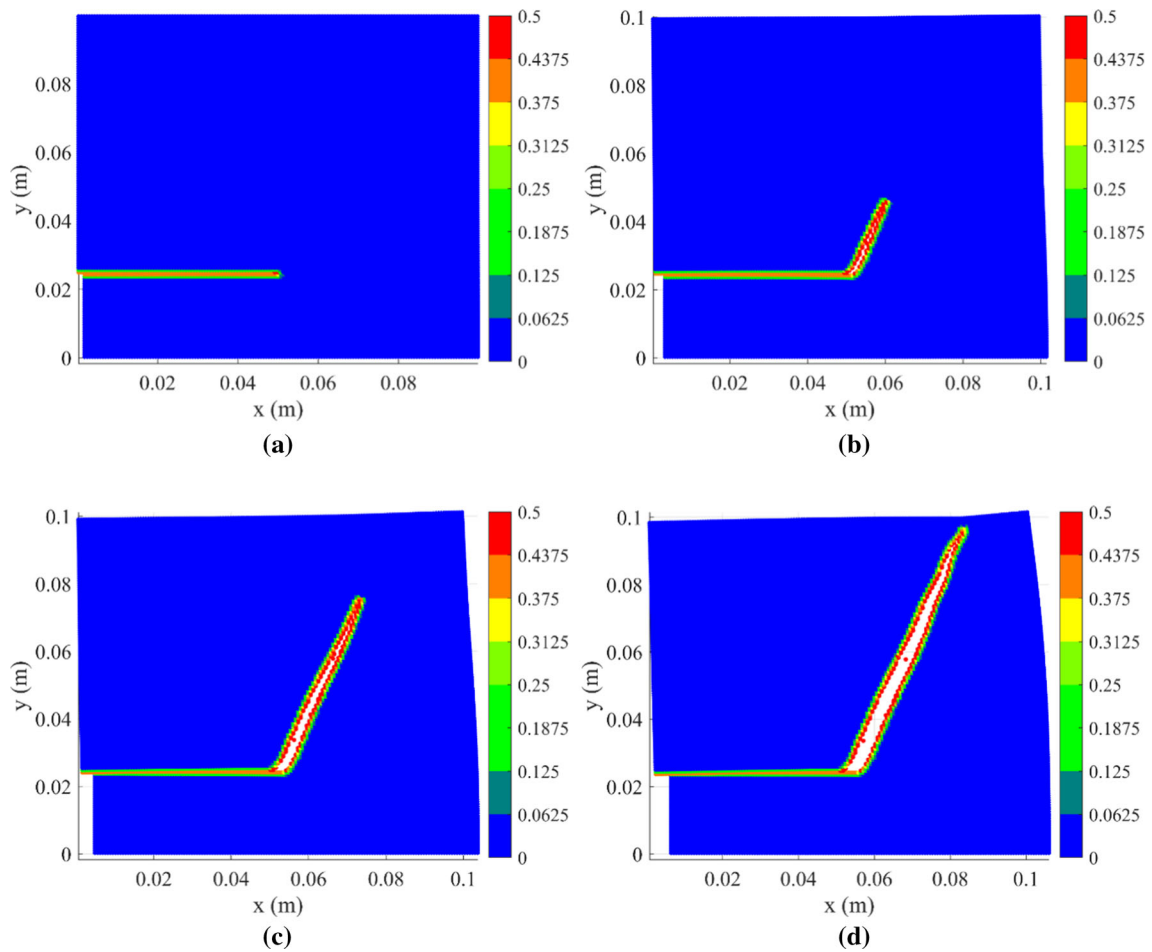


Fig. 31 Damage evolution on the plate at **a** $20\mu\text{s}$, **b** $40\mu\text{s}$, **c** $60\mu\text{s}$, **d** $80\mu\text{s}$ (displacements are magnified by 5 times for deformed configurations)

7 Conclusion

In this study, the PD-based machine learning models for 1D and 2D structures are presented and verified by conducting various problems. The main conclusions arising from the present study are listed below:

1. The PD-based machine learning models are obtained for linear elastic material for the first time in the PD literature by using linear regression. The linear relationships between displacements of a material point with displacements of its family members and its applied forces are presented in the analytical form which can be used straightforward.
2. The machine learning models for both 1D and 2D structures can be coupled with the traditional PD solutions. The numerical procedures for coupling the PD models and the machine learning models are also provided for both static and dynamic problems. The results captured by the coupling models show good agreements with both PD and FEA results.
3. The capability of the coupled model is also further verified by considering various fracture problems. The results captured by the coupled model have very good agreement with the experimental results.

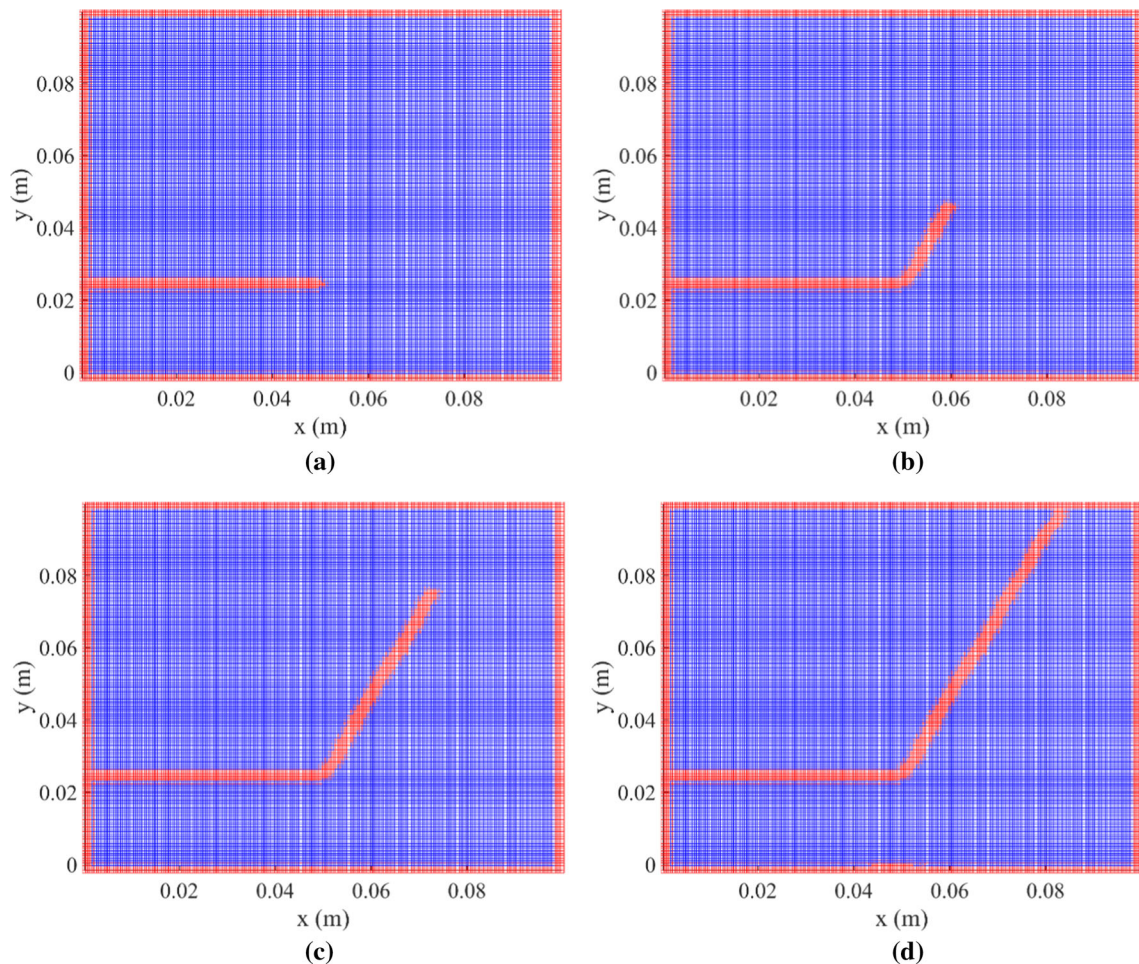


Fig. 32 Adaptive machine learning and PD regions at **a** $20\mu\text{s}$, **b** $40\mu\text{s}$, **c** $60\mu\text{s}$, **d** $80\mu\text{s}$ (PD regions are shown in red, and ML regions are shown in blue in the undeformed configuration)

4. The PD and ML regions can be updated adaptively at each time step. Therefore, the coupled ML and PD model can use the advantage of the PD model in terms of capturing complex fracture problems, and the advantages of the machine learning model in terms of saving computational cost.
5. The hybrid approach proposed in this study can be further applied in the PD literature for other types of structures.

Acknowledgements The authors are grateful for the support of the Ministry of Education and Training in Vietnam and the University of Strathclyde. The results were obtained using the ARCHIE-WeSt High-Performance Computer (www.archie-west.ac.uk).

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Silling, S.A.: Reformulation of elasticity theory for discontinuities and long-range forces. *J. Mech. Phys. Solids* **48**(1), 175–209 (2000)

2. Silling, S.A., Epton, M., Weckner, O., Xu, J., Askari, E.: Peridynamic states and constitutive modeling. *J. Elast.* **88**(2), 151–184 (2007)
3. Silling, S.A., Askari, E.: A meshfree method based on the peridynamic model of solid mechanics. *Comput. Struct.* **83**(17–18), 1526–1535 (2005)
4. Silling, S.A., Lehoucq, R.: Peridynamic theory of solid mechanics. In: *Advances in Applied Mechanics*, vol. 44, Elsevier, pp. 73–168 (2010)
5. Madenci, E., Oterkus, E.: *Peridynamic Theory and Its Applications*. Springer, Berlin (2014)
6. Foster, J.T., Silling, S.A., Chen, W.W.: Viscoplasticity using peridynamics. *Int. J. Numer. Methods Eng.* **81**(10), 1242–1258 (2010)
7. Mitchell, J.A.: A non-local, ordinary-state-based viscoelasticity model for peridynamics. *Sandia Natl. Lab. Rep.* **8064**, 1–28 (2011)
8. Madenci, E., Oterkus, S.: Ordinary state-based peridynamics for plastic deformation according to von Mises yield criteria with isotropic hardening. *J. Mech. Phys. Solids* **86**, 192–219 (2016)
9. Huang, Y., Oterkus, S., Hou, H., Oterkus, E., Wei, Z., Zhang, S.: Peridynamic model for visco-hyperelastic material deformation in different strain rates. *Contin. Mech. Thermodyn.* (2019). <https://doi.org/10.1007/s00161-019-00849-0.pdf>
10. Kružík, M., Mora-Corral, C., Stefanelli, U.: Quasistatic elastoplasticity via peridynamics: existence and localization. *Contin. Mech. Thermodyn.* **30**(5), 1155–1184 (2018)
11. Oterkus, E.: *Peridynamic theory for modeling three-dimensional damage growth in metallic and composite structures*. The University of Arizona (2010)
12. Oterkus, E., Madenci, E., Weckner, O., Silling, S., Bogert, P., Tessler, A.: Combined finite element and peridynamic analyses for predicting failure in a stiffened composite curved panel with a central slot. *Compos. Struct.* **94**(3), 839–850 (2012)
13. Hu, W., Ha, Y.D., Bobaru, F.: Peridynamic model for dynamic fracture in unidirectional fiber-reinforced composites. *Comput. Methods Appl. Mech. Eng.* **217**, 247–261 (2012)
14. Hu, Y., De Carvalho, N., Madenci, E.: Peridynamic modeling of delamination growth in composite laminates. *Compos. Struct.* **132**, 610–620 (2015)
15. Gao, Y., Oterkus, S.: Fully coupled thermomechanical analysis of laminated composites by using ordinary state based peridynamic theory. *Compos. Struct.* **207**, 397–424 (2019)
16. De Meo, D., Zhu, N., Oterkus, E.: Peridynamic modeling of granular fracture in polycrystalline materials. *J. Eng. Mater. Technol.* **138**(4), 041008 (2016)
17. Oterkus, S., Madenci, E., Oterkus, E.: Fully coupled poroelastic peridynamic formulation for fluid-filled fractures. *Eng. Geol.* **225**, 19–28 (2017)
18. Oterkus, S.: *Peridynamics for the solution of multiphysics problems*. The University of Arizona (2015)
19. Gao, Y., Oterkus, S.: Ordinary state-based peridynamic modelling for fully coupled thermoelastic problems. *Contin. Mech. Thermodyn.* **31**(4), 907–937 (2019)
20. Nguyen, C.T., Oterkus, S.: Ordinary state-based peridynamic model for geometrically nonlinear analysis. *Eng. Fract. Mech.* **224**, 106750 (2019)
21. Kefal, A., Sohoul, A., Oterkus, E., Yildiz, M., Suleman, A.: Topology optimization of cracked structures using peridynamics. *Contin. Mech. Thermodyn.* **31**(6), 1645–1672 (2019)
22. Askari, E., Bobaru, F., Lehoucq, R., Parks, M., Silling, S., Weckner, O.: Peridynamics for multiscale materials modeling. In: *Journal of Physics: Conference Series*, vol. 1, IOP Publishing, p. 012078 (2008)
23. Bobaru, F., Ha, Y.D.: Adaptive refinement and multiscale modeling in 2D peridynamics. *J. Multiscale Comput. Eng.* **2011**, 635–659 (2011)
24. Nguyen, C.T., Oterkus, S.: Peridynamics formulation for beam structures to predict damage in offshore structures. *Ocean Eng.* **173**, 244–267 (2019)
25. O’Grady, J., Foster, J.: Peridynamic beams: a non-ordinary, state-based model. *Int. J. Solids Struct.* **51**(18), 3177–3183 (2014)
26. Diyaroglu, C., Oterkus, E., Oterkus, S.: An Euler–Bernoulli beam formulation in an ordinary state-based peridynamic framework. *Math. Mech. Solids* **24**(2), 361–376 (2019)
27. Diyaroglu, C., Oterkus, E., Oterkus, S., Madenci, E.: Peridynamics for bending of beams and plates with transverse shear deformation. *Int. J. Solids Struct.* **69**, 152–168 (2015)
28. O’Grady, J., Foster, J.: Peridynamic plates and flat shells: a non-ordinary, state-based model. *Int. J. Solids Struct.* **51**(25–26), 4572–4579 (2014)
29. Yang, Z., Oterkus, E., Nguyen, C.T., Oterkus, S.: Implementation of peridynamic beam and plate formulations in finite element framework. *Contin. Mech. Thermodyn.* **31**(1), 301–315 (2019)
30. Chowdhury, S.R., Roy, P., Roy, D., Reddy, J.: A peridynamic theory for linear elastic shells. *Int. J. Solids Struct.* **84**, 110–132 (2016)
31. Nguyen, C.T., Oterkus, S.: Peridynamics for the thermomechanical behavior of shell structures. *Eng. Fract. Mech.* **219**, 106623 (2019)
32. Nguyen, C.T., Oterkus, S.: Investigating the effect of brittle crack propagation on the strength of ship structures by using peridynamics. *Ocean Eng.* (2020)
33. Madenci, E., Barut, A., Futch, M.: Peridynamic differential operator and its applications. *Comput. Methods Appl. Mech. Eng.* **304**, 408–451 (2016)
34. Madenci, E., Barut, A., Dorduncu, M.: *Peridynamic Differential Operator for Numerical Analysis*. Springer, Berlin (2019)
35. Madenci, E., Dorduncu, M., Gu, X.: Peridynamic least squares minimization. *Comput. Methods Appl. Mech. Eng.* **348**, 846–874 (2019)
36. Madenci, E., Dorduncu, M., Barut, A., Futch, M.: Numerical solution of linear and nonlinear partial differential equations using the peridynamic differential operator. *Numer. Methods Part. Differ. Equ.* **33**(5), 1726–1753 (2017)
37. Kilic, B., Madenci, E.: Coupling of peridynamic theory and the finite element method. *J. Mech. Mater. Struct.* **5**(5), 707–733 (2010)

38. Liu, W., Hong, J.-W.: A coupling approach of discretized peridynamics with finite element method. *Comput. Methods Appl. Mech. Eng.* **245**, 163–175 (2012)
39. Bie, Y., Cui, X., Li, Z.: A coupling approach of state-based peridynamics with node-based smoothed finite element method. *Comput. Methods Appl. Mech. Eng.* **331**, 675–700 (2018)
40. Fan, H., Bergel, G.L., Li, S.: A hybrid peridynamics-SPH simulation of soil fragmentation by blast loads of buried explosive. *Int. J. Impact Eng.* **87**, 14–27 (2016)
41. Fan, H., Li, S.: A Peridynamics-SPH modeling and simulation of blast fragmentation of soil under buried explosive loads. *Comput. Methods Appl. Mech. Eng.* **318**, 349–381 (2017)
42. Dalla Barba, F., Campagnari, P., Zaccariotto, M., Galvanetto, U., Picano, F.: A fluid-structure interaction model based on peridynamics and Navier–Stokes equations for hydraulic fracture problems. Paper Presented at the 7th European Conference on Computational Fluid Dynamics, Glasgow, UK
43. Javili, A., Morasata, R., Oterkus, E., Oterkus, S.: Peridynamics review. *Math. Mech. Solids* 1081286518803411 (2018)
44. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012)
45. LeCun, Y., Bengio, Y., Hinton, G.J.N.: Deep learning. *Nature* **521**(7553), 436–444 (2015)
46. Lake, B.M., Salakhutdinov, R., Tenenbaum, J.B.: Human-level concept learning through probabilistic program induction. *Science* **350**(6266), 1332–1338 (2015)
47. Alipanahi, B., Delong, A., Weirauch, M.T., Frey, B.J.: Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nat. Biotechnol.* **33**(8), 831 (2015)
48. Lukka, T.J., Tossavainen, T., Kujala, J.V., Raiko, T.: ZenRobotics recycler–robotic sorting using machine learning. In: *Proceedings of the International Conference on Sensor-Based Sorting (SBS)* (2014)
49. Kutz, J.N.: Deep learning in fluid dynamics. *J. Fluid Mech.* **814**, 1–4 (2017)
50. Tesche, C., De Cecco, C.N., Baumann, S., Renker, M., McLaurin, T.W., Duguay, T.M., Bayer 2nd, R.R., Steinberg, D.H., Grant, K.L., Canstein, C.: Coronary CT angiography-derived fractional flow reserve: machine learning algorithm versus computational fluid dynamics modeling. *Radiology* **288**(1), 64–72 (2018)
51. Farrar, C.R., Worden, K.: *Structural Health Monitoring: A Machine Learning Perspective*. Wiley, New York (2012)
52. Söderberg, R., Wärmeffjord, K., Carlson, J.S., Lindkvist, L.: Toward a digital twin for real-time geometry assurance in individualized production. *CIRP Ann.* **66**(1), 137–140 (2017)
53. Do, D.T., Lee, J., Nguyen-Xuan, H.: Fast evaluation of crack growth path using time series forecasting. *Eng. Fract. Mech.* **218**, 106567 (2019)
54. Unger, J.F., Könke, C.: Neural networks as material models within a multiscale approach. *Comput. Struct.* **87**(19–20), 1177–1186 (2009)
55. Kim, M., Winovich, N., Lin, G., Jeong, W.: Peri-Net: Analysis of Crack Patterns Using Deep Neural Networks. *J. Peridyn. Nonlocal Model.* **1**(2), 131–142 (2019)
56. Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics informed deep learning (part i): data-driven solutions of nonlinear partial differential equations. *arXiv preprint arXiv:1711.10561* (2017)
57. Montgomery, D.C., Peck, E.A., Vining, G.G.: *Introduction to Linear Regression Analysis*, vol. 821. Wiley, New York (2012)
58. Alpaydin, E.: *Introduction to Machine Learning*. MIT press, New York (2014)
59. Diyaroglu, C.: *Peridynamics and Its Applications in Marine Structures*. University of Strathclyde (2016)
60. Underwood, P.: Dynamic relaxation. *Comput. Method Trans. Anal.* **1**, 245–263 (1986)
61. Kilic, B., Madenci, E.: An adaptive dynamic relaxation method for quasi-static simulations using the peridynamic theory. *Theor. Appl. Fract. Mech.* **53**(3), 194–204 (2010)
62. Madenci, E., Oterkus, S.: Ordinary state-based peridynamics for thermoviscoelastic deformation. *Eng. Fract. Mech.* **175**, 31–45 (2017)
63. Oterkus, S., Madenci, E.: Peridynamics for antiplane shear and torsional deformations. *J. Mech. Mater. Struct.* **10**(2), 167–193 (2015)
64. Simonsen, B.C., Törnqvist, R.: Experimental and numerical modelling of ductile crack propagation in large-scale shell structures. *Mar. Struct.* **17**(1), 1–27 (2004)
65. Kalthoff, J.F.: Modes of dynamic shear failure in solids. *Int. J. Fract.* **101**(1–2), 1–31 (2000)
66. Jenq, Y., Shah, S.P.: Mixed-mode fracture of concrete. *Int. J. Fract.* **38**(2), 123–142 (1988)
67. Kalthoff, J., Winkler, S.: Failure mode transition at high rates of shear loading: DGM. *Impact Load. Dyn. Behav. Mater.* **1**, 185–195 (1988)
68. Kalthoff, J.F.: Shadow optical analysis of dynamic shear fracture. *Opt. Eng.* **27**(10), 271035 (1988)
69. Borden, M.J., Verhoosel, C.V., Scott, M.A., Hughes, T.J., Landis, C.M.: A phase-field description of dynamic brittle fracture. *Comput. Methods Appl. Mech. Eng.* **217**, 77–95 (2012)