

RESEARCH ARTICLE

# Physics-constrained local convexity data-driven modeling of anisotropic nonlinear elastic solids

Xiaolong He<sup>1</sup>, Qizhi He<sup>2</sup> , Jiun-Shyan Chen<sup>1,\*</sup> , Usha Sinha<sup>3</sup> and Shantanu Sinha<sup>4</sup>

<sup>1</sup>Department of Structural Engineering, University of California San Diego, La Jolla, CA 92093, USA

<sup>2</sup>Physical and Computational Sciences Directorate, Pacific Northwest National Laboratory, Richland, WA 99354, USA

<sup>3</sup>Department of Physics, San Diego State University, San Diego, CA 92182, USA

<sup>4</sup>Department of Radiology, University of California San Diego, La Jolla, CA 92093, USA

\*Corresponding author. E-mail: [js-chen@uscd.edu](mailto:js-chen@uscd.edu)

**Received:** 26 October 2020; **Revised:** 27 November 2020; **Accepted:** 29 November 2020

**Keywords:** Anisotropy; convexity-preserving reconstruction; data-driven computational mechanics; manifold learning; reproducing kernel (RK) approximation

## Abstract

As characterization and modeling of complex materials by phenomenological models remains challenging, data-driven computing that performs physical simulations directly from material data has attracted considerable attention. Data-driven computing is a general computational mechanics framework that consists of a physical solver and a material solver, based on which data-driven solutions are obtained through minimization procedures. This work develops a new material solver built upon the local convexity-preserving reconstruction scheme by He and Chen (2020) A physics-constrained data-driven approach based on locally convex reconstruction for noisy database. *Computer Methods in Applied Mechanics and Engineering* 363, 112791 to model anisotropic nonlinear elastic solids. In this approach, a two-level local data search algorithm for material anisotropy is introduced into the material solver in online data-driven computing. A material anisotropic state characterizing the underlying material orientation is used for the manifold learning projection in the material solver. The performance of the proposed data-driven framework with noiseless and noisy material data is validated by solving two benchmark problems with synthetic material data. The data-driven solutions are compared with the constitutive model-based reference solutions to demonstrate the effectiveness of the proposed methods.

## Impact Statement

While phenomenological constitutive models have been extensively employed in solving solid mechanics problems, formulation and characterization of models become difficult and often lose generality for complex material systems. Data-driven modeling that directly integrates material data with physical laws into simulations has shown promising potential in recent years. This study aims to develop a data-driven modeling approach for nonlinear anisotropic materials by introducing a two-level data search scheme that utilizes both stress–strain data and orientations of material anisotropy to discover the underlying anisotropic material manifold that represents the anisotropic material behaviors. This data search scheme is implemented with a local convexity-preserving reconstruction algorithm for enhanced robustness against noise and outliers in data-driven computing. Ultimately, the direct integration of discrete material data into simulations avoids the need for pre-defining

the stress–strain relationship. The proposed data-driven modeling approach has demonstrated satisfactory accuracy in predicting nonlinear anisotropic material behaviors compared with constitutive model-based solutions, showing promising potential in modeling nonlinear anisotropic material systems.

## 1. Introduction

Constitutive laws that explicitly describe a stress–strain relationship are one of the key components in computational mechanics. However, the phenomenological constitutive models for material modeling are typically based on empirical observations, hypothesis, and mathematical assumptions, often lack of generality to capture complex material behaviors, such as, anisotropy and nonlinearity. For example, it remains challenging to formulate conventional phenomenological constitutive laws for musculoskeletal systems consisting of multiple highly nonlinear and anisotropic biological materials (Huijing, 1999; Gillies and Lieber, 2011; Takaza et al., 2013; Zhang et al., 2020).

With advancements in computing power and significant progresses in data mining, machine learning based data-driven approaches, for example, manifold learning and artificial neural networks, have been extensively applied to various fields owing to their strong ability of feature/pattern extraction, including constitutive material modeling (Ghaboussi et al., 1991; Furukawa and Yagawa, 1998; Lefik and Schrefler, 2003; Wang and Sun, 2018; Liu et al., 2019; Mozaffar et al., 2019; Heider et al., 2020; He and Chen, 2020; He et al., 2020b), surrogate and reduced order modeling (Lee and Chen, 2013; Chen et al., 2015; He et al., 2019; Lee and Carlberg, 2020; Mendizabal et al., 2020; Rocha et al., 2020; Maulik et al., 2020), solving partial differential equations and system identification (Raissi et al., 2019; He et al., 2020a; Goswami et al., 2020), and structural design (Zhang and Ye, 2019; Lei et al., 2019). Recently, a data-driven computing paradigm has been developed to bypass the material modeling step by formulating boundary value problems as a new optimization problem to search for a solution that is closest to the material dataset and constrained by equilibrium and compatibility (Kirchdoerfer and Ortiz, 2016; Ibanez et al., 2018; Conti et al., 2018). Recent developments have extended the physics-constrained data-driven computing framework to dynamics (Kirchdoerfer and Ortiz, 2018), nonlinear elasticity (Nguyen and Keip, 2018; He et al., 2020b), inelasticity (Eggersmann et al., 2019), constitutive manifold construction (Ibañez et al., 2017; Stainier et al., 2019), and multiscale modeling (Xu et al., 2020; Mora-Macías et al., 2020).

To counteract the curse of dimensionality and instability due to noise and outliers present in material datasets, He and Chen (2020) introduced locally linear embedding manifold learning and convexity-preserving reconstruction into the data-driven solver, called local convexity data-driven (LCDD) computing. The LCDD framework has been extended to model nonlinear elastic solids and applied to model mechanical responses of biological heart valve tissues with experimental datasets by He et al. (2020b). Recently, Eggersmann et al. (2020) introduced a tensor voting machine learning technique into the entropy based distance-minimizing data-driven (DMDD) framework to construct locally linear tangent spaces in order to utilize underlying data structures to achieve high-order convergence of data-driven solutions, referred to as a second-order data-driven scheme. The aforementioned physics-constrained data-driven computing frameworks have demonstrated promising performances in various scientific fields. Despite these advances, these data-driven computing frameworks cannot effectively handle anisotropic material systems with various anisotropic orientations, that is, orientations of material anisotropy. This is due to the fact that the data-driven solvers of these frameworks do not consider information of material orientations, which prevents the applications of these data-driven computing frameworks to complex material systems with direction-dependent material behaviors, for example, musculoskeletal systems with significant differences in the muscle fiber direction leading to material anisotropy.

The objective of this study is to develop a new data-driven solver built upon the local convexity-preserving reconstruction scheme by He and Chen (2020) in order to model anisotropic nonlinear elastic solids. The remainder of this paper is organized as follows. Section 2 reviews basic formulations of

physics-constrained data-driven modeling of nonlinear elastic solids and makes a comparison of the material solvers of the DMDD by Kirchdoerfer and Ortiz (2016) and LCDD by He and Chen (2020). A local convexity-preserving material solver designed for anisotropic solids is introduced. Particularly, a rotated material database is constructed offline and a two-level data search is integrated into the material solver to capture direction-dependent material behaviors during online data-driven computing. In Section 3, the proposed data-driven computing framework is verified by modeling the deflection of a cantilever beam with layers containing different fiber directions and inflation of a cylinder where fiber directions vary along the circumferential direction of the cylinder. The data-driven solutions are compared with the constitutive model-based reference solutions to examine the effectiveness and robustness of the proposed methods. Concluding remarks and discussions are given in Section 4.

## 2. Methodologies

This section revisits the basic formulations of the physics-constrained data-driven computing framework for nonlinear elastic solids. The distinction between the material solvers of the DMDD by Kirchdoerfer and Ortiz (2016) and LCDD by He and Chen (2020) and He et al. (2020b) is discussed. To model anisotropic solids, we propose to integrate a two-level data search scheme into the local convexity-preserving material solver in LCDD in order to capture the anisotropic material properties with anisotropic orientations (orientations of material anisotropy) information.

### 2.1. Physics-constrained data-driven modeling of nonlinear elastic solids

We start by summarizing the physics-constrained data-driven computing formulation for nonlinear elastic solids. The deformation of a nonlinear elastic solid defined in a domain  $\Omega^X$  with a Neumann boundary  $\Gamma_t^X$  and a Dirichlet boundary  $\Gamma_u^X$  are described by two basic laws:

$$\text{Equilibrium: } \begin{cases} \text{DIV}(\mathbf{F}(\mathbf{u}) \cdot \mathbf{S}) + \mathbf{b} = \mathbf{0}, & \text{in } \Omega^X, \\ (\mathbf{F}(\mathbf{u}) \cdot \mathbf{S}) \cdot \mathbf{N} = \mathbf{t}, & \text{on } \Gamma_t^X, \end{cases} \quad (1)$$

$$\text{Compatibility: } \begin{cases} \mathbf{E} = \mathbf{E}(\mathbf{u}) = (\mathbf{F}^T \mathbf{F} - \mathbf{I})/2, & \text{in } \Omega^X, \\ \mathbf{u} = \mathbf{g}, & \text{on } \Gamma_u^X, \end{cases} \quad (2)$$

where the superscript “X” denotes the reference (undeformed) configuration,  $\mathbf{u}$  is the displacement vector,  $\mathbf{E}$  is the Green-Lagrangian strain tensor, and  $\mathbf{F}$  is the deformation gradient related to  $\mathbf{u}$ , defined as  $\mathbf{F}(\mathbf{u}) = \partial(\mathbf{X} + \mathbf{u})/\partial\mathbf{X}$ , where  $\mathbf{X}$  is the material coordinate,  $\mathbf{S}$  is the second Piola–Kirchhoff (second-PK) stress tensor, and  $\mathbf{b}$ ,  $\mathbf{N}$ ,  $\mathbf{t}$ , and  $\mathbf{g}$  are the body force, the surface normal on  $\Gamma_t^X$ , the traction on  $\Gamma_t^X$ , and the prescribed displacement on  $\Gamma_u^X$ , respectively. Let  $\mathcal{Z}$  denote the phase space that contains strain–stress pairs  $\mathbf{z} = (\mathbf{E}, \mathbf{S})$ . The physical laws, Equations (1) and (2), define a material independent subset:

$$\mathcal{C} = \{\mathbf{z} \in \mathcal{Z} : (1) \text{ and } (2)\} \quad (3)$$

which contains all states satisfying the physical laws and they are called *physical state*.

Consider that the behaviors of the elastic solid are described by a discrete material dataset  $\mathbb{E} = \left\{ (\widehat{\mathbf{E}}_j, \widehat{\mathbf{S}}_j) \right\}_{j=1}^M$ , where  $M$  is the number of measured material data and a hat symbol “^” is used to denote material quantities. The strain–stress pair  $\widehat{\mathbf{z}} = (\widehat{\mathbf{E}}, \widehat{\mathbf{S}}) \in \mathbb{E}$  is called the *material data*. Conventionally, the material dataset  $\mathbb{E}$  is used to characterize material constants associated with a predefined stress–strain relation,  $\mathbf{S} = \mathbf{f}(\mathbf{E})$  in  $\Omega^X$ . The constructed material model with properly estimated model coefficients is then combined with the physical laws (Equations 1 and 2) to solve the boundary value problem. However, as material models  $\mathbf{f}$  are usually based on empirical experimental observations, physical principles, and mathematical assumptions, with parameters calibrated from experimental data of material samples (Ghaboussi et al., 1991; Sussman and Bathe, 2009), the phenomenological material

modeling process inevitably lacks generality, especially for complex material systems (Latorre and Montáns, 2014; Ibanez et al., 2018).

The recently developed physics-constrained data-driven computing paradigm takes an alternative path to solve boundary value problems, which directly embeds material data into physical simulations, bypassing the need of phenomenological model construction (Kirchdoerfer and Ortiz, 2016; Ibanez et al., 2018; He and Chen, 2020). One class of data-driven computing framework consists of minimizing the distance between the material data  $\widehat{\mathbf{z}}$  and the strain–stress state  $\mathbf{z}$  satisfying the physical laws in Equations (1) and (2), which is called distance-minimizing data-driven (DMDD) computing (Kirchdoerfer and Ortiz, 2016). A global distance functional is defined to measure the distance between the material data  $\widehat{\mathbf{z}}$  and the physical state  $\mathbf{z}$

$$d(\mathbf{z}, \widehat{\mathbf{z}}) = \int_{\Omega^x} d_z^2(\mathbf{z}, \widehat{\mathbf{z}}) d\Omega \tag{4}$$

with

$$d_z^2(\mathbf{z}, \widehat{\mathbf{z}}) = d_E^2(\mathbf{E}(\mathbf{u}), \widehat{\mathbf{E}}) + d_S^2(\mathbf{S}, \widehat{\mathbf{S}}), \tag{5}$$

$$d_E^2(\mathbf{E}(\mathbf{u}), \widehat{\mathbf{E}}) = \frac{1}{2}(\mathbf{E}(\mathbf{u}) - \widehat{\mathbf{E}}) : \mathbf{M} : (\mathbf{E}(\mathbf{u}) - \widehat{\mathbf{E}}), \tag{6}$$

$$d_S^2(\mathbf{S}, \widehat{\mathbf{S}}) = \frac{1}{2}(\mathbf{S} - \widehat{\mathbf{S}}) : \mathbf{M}^{-1} : (\mathbf{S} - \widehat{\mathbf{S}}), \tag{7}$$

where  $\mathbf{M}$  is a predefined symmetric and positive-definite weight tensor used to regularize the distances between  $\mathbf{z}$  and  $\widehat{\mathbf{z}}$ . For a system with multiple materials, multiple material datasets are required for data-driven computing, with each dataset describing material behaviors of one material. An ensemble of these material datasets is called a material database,  $\mathbb{E}_{em} = \mathbb{E}_1 \times \mathbb{E}_2 \times \dots \times \mathbb{E}_m \subset \mathcal{Z}$ , where  $m$  denotes the number of material types,  $\mathbb{E}_p = \left\{ (\widehat{\mathbf{E}}_j^p, \widehat{\mathbf{S}}_j^p) \right\}_{j=1}^{M_p}$ , and  $M_p$  is the number of material data in  $\mathbb{E}_p, p = 1, \dots, m$ .

### 2.2. Data-driven two-step processes: material and physical solvers

The data-driven computing problem is formulated by stating:

$$\min_{\widehat{\mathbf{z}} \in \mathbb{E}_{em}} \min_{\mathbf{z} \in \mathcal{C}} d(\mathbf{z}, \widehat{\mathbf{z}}) = \min_{\mathbf{z} \in \mathcal{C}} \min_{\widehat{\mathbf{z}} \in \mathbb{E}_{em}} d(\mathbf{z}, \widehat{\mathbf{z}}), \tag{8}$$

where  $d(\mathbf{z}, \widehat{\mathbf{z}})$  is the global distance functional defined by Equation (4). The objective is to find the material data  $\widehat{\mathbf{z}} \in \mathbb{E}_{em}$  in the material database that is closest to the constraint set  $\mathcal{C}$  of physical states  $\mathbf{z}$  or equivalently to find the physical state  $\mathbf{z} \in \mathcal{C}$  that is closest to the material database  $\mathbb{E}_{em}$ . The data-driven problem in Equation (8) can be decomposed into a two-step problem:

$$\text{Physical step : } \mathbf{z}^* = \arg \min_{\mathbf{z} \in \mathcal{C}} d(\mathbf{z}, \widehat{\mathbf{z}}), \tag{9}$$

$$\text{Material step : } \widehat{\mathbf{z}}^* = \arg \min_{\widehat{\mathbf{z}} \in \mathbb{E}_{em}} d(\mathbf{z}^*, \widehat{\mathbf{z}}), \tag{10}$$

where  $\widehat{\mathbf{z}}^*$  is the optimal material data sought from the material step that is closest to the physical state  $\mathbf{z}^*$  computed from the physical step.

#### 2.2.1. Physical solver

A *physical solver* is used to solve the physical step defined in Equation (9), which searches for the physical state  $\mathbf{z} \in \mathcal{C}$  that is closest to a given material data  $\widehat{\mathbf{z}}$ . It can be reformulated as a constrained minimization problem:

$$\begin{aligned} \min_{\mathbf{z} \in \mathcal{C}} d(\mathbf{z}, \widehat{\mathbf{z}}) &= \min_{\mathbf{u}, \mathbf{S}} \int_{\Omega^x} \left( d_E^2(\mathbf{E}(\mathbf{u}), \widehat{\mathbf{E}}) + d_S^2(\mathbf{S}, \widehat{\mathbf{S}}) \right) d\Omega \\ \text{subject to: } & \text{DIV}(\mathbf{F}(\mathbf{u}) \cdot \mathbf{S}) + \mathbf{b} = \mathbf{0} \text{ in } \Omega^x, \\ & (\mathbf{F}(\mathbf{u}) \cdot \mathbf{S}) \cdot \mathbf{N} = \mathbf{t} \text{ on } \Gamma_t^x. \end{aligned} \tag{11}$$

Note that the Green–Lagrangian strain tensor  $\mathbf{E}$  is obtained from the displacement function of sufficient smoothness so the compatibility condition is naturally imposed in the physical solver. Enforcing the physical constraints by the Lagrange multiplier  $\lambda$  gives the following functional:

$$\int_{\Omega^x} \left[ d_E^2(\mathbf{E}(\mathbf{u}), \widehat{\mathbf{E}}) + d_S^2(\mathbf{S}, \widehat{\mathbf{S}}) \right] d\Omega + \lambda \left[ \int_{\Omega^x} [\text{DIV}(\mathbf{F}(\mathbf{u}) \cdot \mathbf{S}) + \mathbf{b}] d\Omega - \int_{\Gamma_t^x} [(\mathbf{F}(\mathbf{u}) \cdot \mathbf{S}) \cdot \mathbf{N} - \mathbf{t}] d\Gamma \right]. \tag{12}$$

The stationary condition of Equation (12) yields the following data-driven variational equations (He and Chen, 2020; He et al., 2020b; Nguyen et al., 2020):

$$\delta \mathbf{u} : \int_{\Omega^x} \delta \mathbf{E}(\mathbf{u}) : \mathbf{M} : (\mathbf{E}(\mathbf{u}) - \widehat{\mathbf{E}}) d\Omega = \int_{\Omega^x} \delta \mathbf{F}^T(\mathbf{u}) \cdot \nabla \lambda : \mathbf{S} d\Omega, \tag{13a}$$

$$\delta \mathbf{S} : \int_{\Omega^x} \delta \mathbf{S} : (\mathbf{M}^{-1} : \mathbf{S} - \mathbf{F}^T(\mathbf{u}) \cdot \nabla \lambda) d\Omega = \int_{\Omega^x} \delta \mathbf{S} : \mathbf{M}^{-1} : \widehat{\mathbf{S}} d\Omega, \tag{13b}$$

$$\delta \lambda : \int_{\Omega^x} \delta \nabla \lambda : (\mathbf{F}(\mathbf{u}) \cdot \mathbf{S}) d\Omega = \int_{\Omega^x} \delta \lambda \cdot \mathbf{b} d\Omega + \int_{\Gamma_t^x} \delta \lambda \cdot \mathbf{t} d\Gamma. \tag{13c}$$

The variational formulation in Equations (13a–c) can be solved by numerical solvers, for example, the Finite Element Method and the Reproducing Kernel Particle Method (RKPM) (Liu et al., 1995; Chen et al., 1996). In this study, the RKPM numerical solver is adopted for approximating the displacement field and the Lagrange multiplier field due to its nodal approximation of state and field variables that are particularly effective for data-driven computing, see more details of RKPM in Appendix. The stabilized conforming nodal integration (SCNI) by Chen et al. (2002) scheme is adopted such that the integration points share the same set of points as the discretization nodes, which allows all material data search and variable evaluation to be performed only at the nodal points, enhancing efficiency and accuracy of data-driven computing. Consistent to the nodal integration, the stress  $\mathbf{S}$  is approximated by indicator functions  $\mathcal{X}_i(\mathbf{X})$  so that its nodal values are directly associated with the material stress data without introducing additional interpolation errors, see He and Chen (2020) for details.

$$\mathbf{S}(\mathbf{X}) \approx \mathbf{S}^h(\mathbf{X}) = \sum_{i=1}^N \mathcal{X}_i(\mathbf{X}) \mathbf{S}_i, \tag{14}$$

where  $N$  is the number of integration points,  $\mathbf{S}_i$  is the nodal stress associated with the  $i$ th node, and  $\mathcal{X}_i(\mathbf{X})$  is expressed as

$$\mathcal{X}_i(\mathbf{X}) = \begin{cases} 1, & \text{if } \mathbf{X} \in \Omega_i^x, \\ 0, & \text{if } \mathbf{X} \notin \Omega_i^x. \end{cases} \tag{15}$$

Employing nodal integration and the stress approximation in Equation (14), the discrete form of Equation (13b) becomes

$$\sum_{i=1}^N V_i \delta \mathbf{S}_i^T (\mathbf{M}_i^{-1} \mathbf{S}_i - (\mathbf{F}^T(\mathbf{u}))_i \cdot \nabla \lambda_i) = \sum_{i=1}^N V_i \delta \mathbf{S}_i^T \mathbf{M}_i^{-1} \widehat{\mathbf{S}}_i, \quad \forall \delta \mathbf{S}_i, \quad i = 1, \dots, N, \tag{16}$$

where the subscript  $i$  denotes the terms evaluated at  $\mathbf{X}_i$ , e.g.,  $\mathbf{S}_i = \mathbf{S}(\mathbf{X}_i)$  and  $\mathbf{F}^T(\mathbf{u})_i = \mathbf{F}(\mathbf{u}(\mathbf{X}_i))$ , and  $V_i$  is

the integration weight associated with the  $i$ th node. Equation (16) yields the stress update for all integration points:

$$\mathbf{S} = \widehat{\mathbf{S}} + \mathbf{M} : (\mathbf{F}^T(\mathbf{u}) \cdot \nabla \lambda). \tag{17}$$

Substituting Equation (17) into Equations (13a) and (13c) yields the following nonlinear system of  $\mathbf{u}$  and  $\lambda$  (He et al., 2020b)

$$\int_{\Omega^x} \delta \mathbf{E}(\mathbf{u}) : \mathbf{M} : (\mathbf{E}(\mathbf{u}) - \widehat{\mathbf{E}}) d\Omega = \int_{\Omega^x} (\delta \mathbf{F}^T(\mathbf{u}) \cdot \nabla \lambda) : \mathbf{M} : (\mathbf{F}^T(\mathbf{u}) \cdot \nabla \lambda) d\Omega + \int_{\Omega^x} (\delta \mathbf{F}^T(\mathbf{u}) \cdot \nabla \lambda) : \widehat{\mathbf{S}} d\Omega, \tag{18a}$$

$$\int_{\Omega^x} (\mathbf{F}^T(\mathbf{u}) \cdot \delta \nabla \lambda) : [\mathbf{M} : (\mathbf{F}^T(\mathbf{u}) \cdot \nabla \lambda) + \widehat{\mathbf{S}}] d\Omega = \int_{\Omega^x} \delta \nabla \lambda \cdot \mathbf{b} d\Omega + \int_{\Gamma_f^x} \delta \nabla \lambda \cdot \mathbf{t} d\Gamma, \tag{18b}$$

which can be solved by the Newton–Raphson method (Belytschko et al., 2000)

2.2.2. Material solver

Following the physical step, a material step is performed to search for the optimal material data  $\widehat{\mathbf{z}}^*$  that is closest to the physical state  $\mathbf{z}^*$  obtained from the physical step by minimizing the distance functional. This is done by the following *material solver*, which decomposes the global minimization in Equation (10) into  $N$  local minimization problems (Kirchdoerfer and Ortiz, 2016).

$$\widehat{\mathbf{z}}_i^* = \arg \min_{\widehat{\mathbf{z}}_i \in \mathbb{E}_{em}} d_z^2(\mathbf{z}_i^*, \widehat{\mathbf{z}}_i), \quad i = 1, \dots, N, \tag{19}$$

where  $i$  denotes the indices of integration points and  $N$  is the total number of integration points.

The data-driven solutions are obtained through fixed-point iterations of the physical step (Equations 17 and 18) and the material step (Equation 19) until the variation in material-step solutions between two consecutive iterations is within a certain tolerance, see Kirchdoerfer and Ortiz (2016) and Conti et al. (2018) for discussion of convergence properties of this two-step fixed-point iteration solver. The DMDD computing process is depicted in Figure 1, where  $\mathcal{E}$  is the material admissible set,  $(v)$  is the iteration index, and one iteration contains one physical step and one material step. In DMDD computing, given a physical state  $\mathbf{z}^{*(v)}$  of a material point, the material solver searches for the closest material data  $\widehat{\mathbf{z}}^{*(v)}$  directly from the material database  $\mathbb{E}_{em}$ . However, it can result in unsatisfactory accuracy of data-driven solutions when noise and outliers presented in the material data (Kirchdoerfer and Ortiz, 2017; He and Chen, 2020).

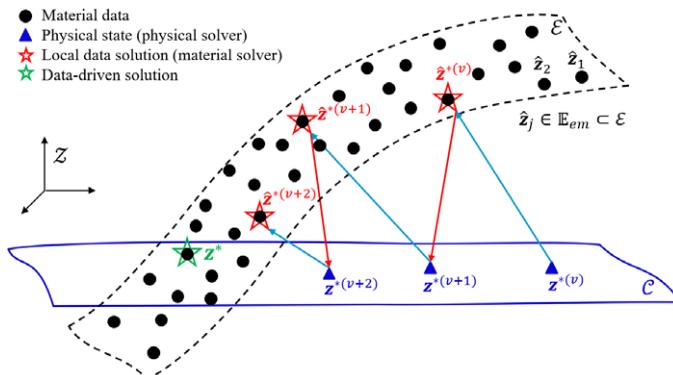


Figure 1. Geometric schematic of the distance-minimizing data-driven (DMDD) solver (Kirchdoerfer and Ortiz, 2016).

### 2.3. Local convexity-preserving material solver

The solutions from the material step are critical to data-driven computing, as they represent the material behaviors of the anisotropic solids. As discussed in Section 2.2, the conventional DMDD material solver (Equation 19) that directly searches for the closest material data can lead to inaccurate data-driven solutions when the material data contains noise and outliers. To enhance the robustness of data-driven computing against noise and outliers, the LCDD computing (He and Chen, 2020) was developed by introducing the underlying manifold structure of material data to the material solver, which solves the below material step:

$$\hat{\mathbf{z}}_i^* = \arg \min_{\hat{\mathbf{z}}_i \in \mathcal{E}(\mathbf{z}_i^*)} d_z^2(\mathbf{z}_i^*, \hat{\mathbf{z}}_i), \quad i = 1, \dots, N, \tag{20}$$

where  $\mathcal{E}(\mathbf{z}_i^*)$  is a local convex space formed by  $k$  material data points that are closest to a given physical state  $\mathbf{z}_i^*$  of material point  $\mathbf{X}_i$ , providing a smooth and bounded solution space for optimal material data search, and preserving the convexity of the constructed local material manifold for enhanced robustness and convergence stability. The material step defined in Equation (20) involves two substeps. First, given a physical state  $\mathbf{z}_i^*$ ,  $k$  nearest neighbors (material data points),  $\{\hat{\mathbf{z}}_\alpha\}_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)} \subset \mathbb{E}_{em}$ , are identified based on the distance measured by  $d_z^2(\mathbf{z}_i^*, \hat{\mathbf{z}})$ , where the indices of the nearest neighbors of  $\mathbf{z}_i^*$  are stored in  $\mathcal{N}_k(\mathbf{z}_i^*)$ . Then, a local convex space is constructed based on the collected nearest neighbors  $\{\hat{\mathbf{z}}_\alpha\}_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)}$  of  $\mathbf{z}_i^*$ , defined as

$$\mathcal{E}(\mathbf{z}_i^*) = \text{Conv}\left(\{\hat{\mathbf{z}}_\alpha\}_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)}\right) = \left\{ \sum_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)} w_\alpha \hat{\mathbf{z}}_\alpha \mid \sum_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)} w_\alpha = 1 \text{ and } w_\alpha \geq 0, \forall \alpha \in \mathcal{N}_k(\mathbf{z}_i^*) \right\}, \tag{21}$$

The optimal coefficients  $\mathbf{w}_i = \{w_\alpha\}_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)}$  are obtained by solving the following minimization problem:

$$\begin{aligned} \mathbf{w}_i^* &= \arg \min_{\mathbf{w}_i} d_z^2\left(\mathbf{z}_i^*, \sum_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)} w_\alpha \hat{\mathbf{z}}_\alpha\right) \\ \text{subject to :} & \sum_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)} w_\alpha = 1, \\ & w_\alpha \geq 0, \quad \forall \alpha \in \mathcal{N}_k(\mathbf{z}_i^*), \end{aligned} \tag{22}$$

where  $\mathbf{w}_i^* = \{w_\alpha^*\}_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)}$  denotes the optimal coefficients. Equation (22) is solved by means of a non-negative least-square algorithm with penalty relaxation, see details in He and Chen (2020). The optimal material data  $\hat{\mathbf{z}}_i^*$  can then be obtained by the following local convex construction:

$$\hat{\mathbf{z}}_i^* = \sum_{\alpha \in \mathcal{N}_k(\mathbf{z}_i^*)} w_\alpha^* \hat{\mathbf{z}}_\alpha, \tag{23}$$

which ensures that the optimal material data  $\hat{\mathbf{z}}_i^*$  always lie within the local convex space  $\mathcal{E}(\mathbf{z}_i^*)$ . The LCDD computing process is depicted in Figure 2, where the material solver finds the optimal material data within the local convex space  $\mathcal{E}(\mathbf{z}_i^*)$  (denoted by enclosed black dash lines) formed by the  $k$  selected data points closest to the given physical state, which expands the feasible solution space for robustness in data-driven iterations against noise and outliers in the material data (He and Chen, 2020).

### 2.4. Local convexity-preserving material solver for anisotropic solids

To capture direction-dependent material properties of anisotropic solids, a two-level local data search scheme is introduced in the local convexity-preserving material solver. Let us consider an anisotropic



Applying the rotation  $\mathbf{R}_i$  to the strain–stress data  $\mathbb{E} = \left\{ (\widehat{\mathbf{E}}_j, \widehat{\mathbf{S}}_j) \right\}_{j=1}^M$  yields the rotated strain–stress data  $\mathbb{E}_i^\theta = \left\{ (\widehat{\mathbf{E}}_j^\theta, \widehat{\mathbf{S}}_j^\theta) \right\}_{j=1}^M$  representing the material behaviors with the anisotropic orientation  $\theta_i$  under the reference frame:

$$\widehat{\mathbf{E}}_j^\theta = \mathbf{R}_i \cdot \widehat{\mathbf{E}}_j \cdot \mathbf{R}_i^T, \quad j = 1, \dots, M \tag{25a}$$

$$\widehat{\mathbf{S}}_j^\theta = \mathbf{R}_i \cdot \widehat{\mathbf{S}}_j \cdot \mathbf{R}_i^T, \quad j = 1, \dots, M. \tag{25b}$$

The rotated material datasets are normally obtained offline with various angles associated with material points in the physical system. Rotated material datasets are then used to reconstruct the optimal material data in the material solver during online data-driven computing.

However, it is impractical to prepare rotated material datasets when a system contains a large number of varying anisotropic orientations associated with material points as a collection of all possible rotated material datasets requires prohibitive memory. A typical example is muscle tissues in musculoskeletal systems that involve randomly oriented fibers. To effectively model anisotropic behaviors in complex material systems, we introduce a two-level local data search scheme into the material solver. To this end, the anisotropic orientation  $\theta$  is encoded as an additional feature in material data and physical states of material points. Consequently, the distance between the material data and physical state is not only measured by the distance between their strain–stress values, called *state distance*, but also the distance between their anisotropic orientations, called *anisotropic distance*, expressed as

$$d_{zf}^2 \left( (\mathbf{z}, \theta) (\mathbf{z}, \widehat{\theta}) \right) = d_z^2(\mathbf{z}, \widehat{\mathbf{z}}) + d_f^2(\theta, \widehat{\theta}). \tag{26}$$

The state distance  $d_z(\mathbf{z}, \widehat{\mathbf{z}})$  is computed by Equations (6) and (7) and the anisotropic distance  $d_f(\theta, \widehat{\theta})$  is defined as

$$d_f(\theta, \widehat{\theta}) = \|\theta - \widehat{\theta}\|, \tag{27}$$

where  $\widehat{\theta}$  and  $\theta$  denote the anisotropic orientations of material data and the material point, respectively.

#### 2.4.1. Level 1 data search

Let us consider a system constituted by an anisotropic material with various anisotropic orientations, for example, a musculoskeletal system consisting muscle fibers with various fiber orientations. The anisotropic material with various anisotropic orientations in the global reference frame exhibits the same material behaviors under the local fiber frame. We first prepare a rotated material database that contains  $\widehat{m}$  rotated material datasets obtained by rotating the original dataset with  $\widehat{m}$  different angles (anisotropic orientations). Each rotated material datasets contains strain–stress data  $\widehat{\mathbf{z}}$  and an anisotropic orientation  $\widehat{\theta}$ . The range of variations in anisotropic orientations of the rotated material database is sufficiently large to cover all material points in the system. Given a physical state of a material point,  $(\mathbf{z}_i, \theta_i)$ , where the subscript  $i$  denotes the index of a material (integration) point  $\mathbf{X}_i$ , we first compute the *anisotropic distance* (Equation 27) between the material point  $(\theta_i)$  and all rotated material datasets  $(\widehat{\theta}_p, p = 1, 2, \dots, \widehat{m})$ . Two rotated material datasets,  $\mathbb{E}_p^\theta$  and  $\mathbb{E}_q^\theta$ , will then be selected so that  $\widehat{\theta}_p \leq \theta_i \leq \widehat{\theta}_q$ , as illustrated by the blue dash block in Figure 4. The selected material datasets have the anisotropic orientations closest to that of the material point and therefore are considered to best represent the anisotropic material behaviors of the material point.

#### 2.4.2. Level 2 data search

Given the two selected rotated material datasets by the Level 1 search scheme, we search for  $k$  nearest neighbors (*KNN*) within each dataset based on the state distance between the physical state of the

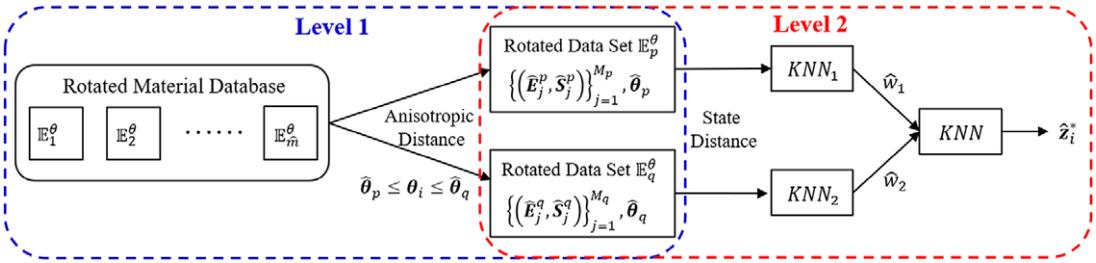


Figure 4. Illustration of two-level local data search in the proposed material solver.

material point and the selected material datasets. The lists of  $k$  material data points closest to the physical state of the material point from the first and second selected material dataset are denoted as  $KNN_1$  and  $KNN_2$ , respectively, as shown in Figure 4. To properly consider the effect of the two  $KNN$  datasets, we propose to use a linear weighting scheme based on the anisotropic orientation information as follows:

$$\hat{w}_2 = \frac{d_f(\theta_i, \hat{\theta}_p)}{d_f(\hat{\theta}_q, \hat{\theta}_p)} = \frac{\|\theta_i - \hat{\theta}_p\|}{\|\hat{\theta}_q - \hat{\theta}_p\|}, \tag{28a}$$

$$\hat{w}_1 = 1 - \hat{w}_2. \tag{28b}$$

The final list of  $k$  nearest neighbors, which is formed by  $\text{int}(k \cdot \hat{w}_1)$  (rounded to the nearest integer) nearest neighbors from  $KNN_1$  and  $\text{int}(k \cdot \hat{w}_2)$  nearest neighbors from  $KNN_2$ , is used for local convexity-preserving reconstruction of the optimal material data that is closest to the physical state of the material point by

$$(\hat{z}_i^*, \hat{\theta}_i^*) = \arg \min_{(\hat{z}_i, \hat{\theta}_i) \in \tilde{\mathcal{E}}(z_i^*, \theta_i)} d_z^2(z_i^*, \hat{z}_i) + d_f^2(\theta_i, \hat{\theta}_i), \quad i = 1, \dots, N, \tag{29}$$

where  $z_i^*$  is the optimal strain–stress state obtained from the physical step (Equation 9) for the material point  $X_i$ ,  $\theta_i$  is the anisotropic orientation of the material point  $X_i$ ,  $\tilde{\mathcal{E}}(z_i^*, \theta_i)$  is a local convex space formed by the selected  $k$  nearest neighbors  $\{(\hat{z}_\alpha, \hat{\theta}_\alpha)\}_{\alpha \in \mathcal{N}_k(z_i^*, \theta_i)}$  of the physical state  $(z_i^*, \theta_i)$  based on the distance measured by  $d_z^2(z_i^*, \hat{z}) + d_f^2(\theta_i, \hat{\theta})$ . The anisotropic orientations  $\theta_i$  and  $\hat{\theta}_i$  are normalized by  $\hat{\theta}_q$  before the calculation of anisotropic distance.

Note that the construction of convexity-preserving local manifold takes the anisotropic orientations of the selected nearest neighbors into account in forming the local convex space  $\tilde{\mathcal{E}}(z_i^*, \theta_i)$  so that the anisotropic orientations of nearest neighbors play an important role in reconstructing the optimal material data  $(\hat{z}_i^*, \hat{\theta}_i^*)$  closest to the physical state  $(z_i^*, \theta_i)$ . Let  $\mathbf{a}_i^*$  and  $\hat{\mathbf{a}}_i$  denote  $(z_i^*, \theta_i)$  and  $(\hat{z}_i, \hat{\theta}_i)$ , respectively. The total distance between  $\mathbf{a}_i^*$  and  $\hat{\mathbf{a}}_i$  is denoted by  $d_{zf}^2(\mathbf{a}_i^*, \hat{\mathbf{a}}_i)$ . The local convex space  $\tilde{\mathcal{E}}(\mathbf{a}_i^*)$  is constructed based on the collected nearest neighbors  $\{\hat{\mathbf{a}}_\alpha\}_{\alpha \in \mathcal{N}_k(\mathbf{a}_i^*)}$ , defined as

$$\tilde{\mathcal{E}}(\mathbf{a}_i^*) = \text{Conv}\left(\{\hat{\mathbf{a}}_\alpha\}_{\alpha \in \mathcal{N}_k(\mathbf{a}_i^*)}\right) = \left\{ \sum_{\alpha \in \mathcal{N}_k(\mathbf{a}_i^*)} \tilde{w}_\alpha \hat{\mathbf{a}}_\alpha \mid \sum_{\alpha \in \mathcal{N}_k(\mathbf{a}_i^*)} \tilde{w}_\alpha = 1 \text{ and } \tilde{w}_\alpha \geq 0, \forall \alpha \in \mathcal{N}_k(\mathbf{a}_i^*) \right\}, \tag{30}$$

The optimal coefficients  $\tilde{\mathbf{w}}_i = \{\tilde{w}_\alpha\}_{\alpha \in \mathcal{N}_k(\mathbf{a}_i^*)}$  are obtained by solving the following minimization problem:

$$\begin{aligned}
 \tilde{\mathbf{w}}_i^* = \arg \min_{\tilde{\mathbf{w}}_i} & d_{zf}^2 \left( \mathbf{a}_i^*, \sum_{\alpha \in \mathcal{N}_k(\mathbf{a}_i^*)} \tilde{w}_\alpha \hat{\mathbf{a}}_\alpha \right) \\
 \text{subject to:} & \sum_{\alpha \in \mathcal{N}_k(\mathbf{a}_i^*)} \tilde{w}_\alpha = 1, \\
 & \tilde{w}_\alpha \geq 0, \forall \alpha \in \mathcal{N}_k(\mathbf{a}_i^*),
 \end{aligned}
 \tag{31}$$

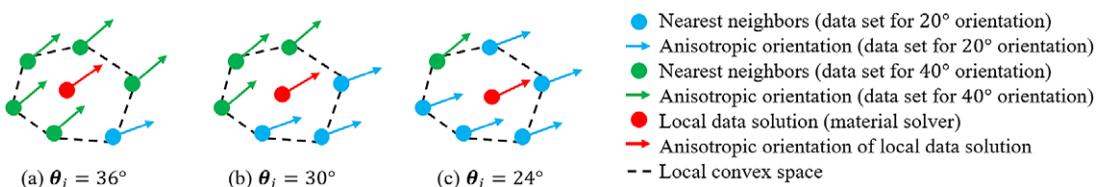
where  $\tilde{\mathbf{w}}_i^* = \{\tilde{w}_\alpha^*\}_{\alpha \in \mathcal{N}_k(\mathbf{a}_i^*)}$  denote the optimal coefficients. Equation (31) is solved by the non-negative least-square algorithm with penalty relaxation used to solve for the optimal coefficients of the local convex space in LCDD’s material solver (Equation 22), see more details in He and Chen (2020). The optimal material data  $\hat{\mathbf{a}}_i^* = (\hat{\mathbf{z}}_i^*, \hat{\boldsymbol{\theta}}_i^*)$  can then be obtained by the following local convex construction:

$$\hat{\mathbf{a}}_i^* = \sum_{\alpha \in \mathcal{N}_k(\mathbf{a}_i^*)} \tilde{w}_\alpha^* \hat{\mathbf{a}}_\alpha,
 \tag{32}$$

The optimal material data sought from the material solver is considered to be “closest” to the physical state in terms of both state distance and anisotropic distance and thus best represent the anisotropic material properties of the material point. The optimal material data will then be input to the physical step (Equations 17 and 18) to solve for the closest physical state in the next data-driven iteration. Note that anisotropic properties are embedded in material database, which are independent of physical laws. Hence, the physical solver only requires the optimal strain–stress material data,  $\hat{\mathbf{z}}_i^*$ , reconstructed from the material solver.

### 2.4.3. Schematic illustration

Figure 5 shows schematic examples of the material step in a two-dimensional space, where  $\hat{\boldsymbol{\theta}}_p = 20^\circ$ ,  $\hat{\boldsymbol{\theta}}_q = 40^\circ$ , and  $k = 6$  are considered. Three different anisotropic orientations  $\theta_i$  of the material point  $i$  are considered, that is,  $36^\circ$ ,  $30^\circ$ , and  $24^\circ$ . The linear weights for selecting the nearest neighbors ( $KNN$  weights) from each material dataset are computed based on the anisotropic orientation information by Equation (28). The number of nearest neighbors from each material dataset are computed by multiplying their  $KNN$  weights with the total number of nearest neighbors  $k$ , as listed in Table 1, which are rounded to the nearest integer. This approach follows the idea of instance-based learning (Mitchell et al., 1997) to learn the underlying relation from the neighboring manifold of observation data, but additionally, it assigns different number of votes to the selected datasets based on their anisotropic distance  $d_f$  (Equation 27). For example, in the case that the anisotropic orientation of the material point  $i$  is  $\theta_i = 36^\circ$ , which is closer to that of Dataset 2 ( $\hat{\boldsymbol{\theta}}_q = 40^\circ$ ) and thus the computed  $KNN$  weight for Dataset 2 is  $\hat{w}_2 = 0.8$ , larger than  $\hat{w}_1 = 0.2$  of Dataset 1 ( $\hat{\boldsymbol{\theta}}_p = 20^\circ$ ). Hence, there are 1 and 5 nearest neighbors selected from Dataset 1 and Dataset 2, respectively, which are used to form a local convex space capturing the underlying anisotropic data structure, as shown in Figure 5a. The optimal material data (red circle in



**Figure 5.** Schematic illustration of the proposed material solver for anisotropic solids. Dataset 1 has an anisotropic orientation of  $\hat{\boldsymbol{\theta}}_p = 20^\circ$ . Dataset 2 has an anisotropic orientation of  $\hat{\boldsymbol{\theta}}_q = 40^\circ$ . The total number of nearest neighbors  $k$  is 6. The material step of a material point with different anisotropic orientations are compared: (a)  $\theta_i = 36^\circ$ ; (b)  $\theta_i = 30^\circ$ ; (c)  $\theta_i = 24^\circ$ .

**Table 1.** The number of nearest neighbors from each dataset in the examples shown in Figure 5 with  $\hat{\theta}_p = 20^\circ$ ,  $\hat{\theta}_q = 40^\circ$ , and  $k = 6$ .

$\theta_i$	$\hat{w}_1$	$\hat{w}_2$	$k \cdot \hat{w}_1$	$k \cdot \hat{w}_2$
$36^\circ$	0.2	0.8	1	5
$30^\circ$	0.5	0.5	3	3
$24^\circ$	0.8	0.2	5	1

The weights are computed by Equation (28).  $k \cdot \hat{w}_1$  and  $k \cdot \hat{w}_2$  are rounded to the nearest integer.

Figure 5a) is then reconstructed from the local convex space. As the anisotropic orientation of the material point is closer to that of Dataset 2, more nearest neighbors are selected from Dataset 2 for local convex reconstruction and therefore the optimal material data that is closest to the given physical state will have anisotropic properties closer to that of Dataset 2. In the case that the anisotropic orientation is  $\theta_i = 30^\circ$ , both datasets have the same KNN weights and contribute the same number of nearest neighbors to the construction of local convex space, as shown in Figure 5b, indicating that they have the same effects on reconstructing the optimal material data.

Note that the relation between the anisotropic properties and anisotropic orientations is nonlinear. More sophisticated anisotropic distance metrics and local data reconstruction schemes are required in order to achieve higher accuracy in anisotropic material data reconstruction if the selected material datasets have a large anisotropic distance, which will be investigated in our future study. In the cases that the anisotropic distance of the selected rotated material datasets is small, the  $L_2$ -based metric for anisotropic distance (Equation 27) and the linear local convexity-preserving reconstruction scheme (Equation 32) adopted in this study can yield desirable accuracy in the data-driven modeling of anisotropic materials, which will be demonstrated in the numerical examples in the following sections.

### 2.5. Local convexity-preserving data-driven solver for anisotropic solids

Given an anisotropic material dataset  $\mathbb{E}$  and anisotropic orientations of material points in the system, the proposed data-driven solver for modeling anisotropic solids is summarized as follows.

#### Offline stage:

*Step 1.* Define a range for variations in anisotropic orientations for material data rotation so that it covers that of the material points in the system. Depending on the distribution of the anisotropic orientations in the system, anisotropic orientations for data rotation can be evenly distributed in the defined range or follow certain statistical distributions in order to better capture anisotropic properties of the system.

*Step 2.* Construct a rotated material database  $\mathbb{E}_{em} = \mathbb{E}_1^\theta \times \dots \times \mathbb{E}_m^\theta$  by rotating the original anisotropic material dataset  $\mathbb{E}$  with the anisotropic orientations defined in *Step 1*. The rotation of strain and stress data is performed by Equation (25).

#### Online stage:

*Step 1.* Randomly initialize  $\hat{\mathbf{z}}_i^{(0)}$ ,  $i = 1, \dots, N$ , where  $i$  denote the indices of material points and  $N$  is the number of material points in the system, and set the data-driven iteration index  $v = 0$ .

*Step 2.* Solve the physical step (Equations 13–19) for physical states of all material points,  $\mathbf{z}_i^{*(v)}$ ,  $i = 1, \dots, N$ .

*Step 3.* For each physical state at each integration point,  $(\mathbf{z}_i^{*(v)}, \theta_i)$ , perform two-level local data search (material step):

3.1. Search for two rotated material datasets with anisotropic orientations  $\hat{\theta}_p$  and  $\hat{\theta}_q$  so that  $\hat{\theta}_p \leq \theta_i \leq \hat{\theta}_q$ .

3.2. From each selected material dataset, search for  $k$  material data points that are closest to the physical state based on the state distance  $d_z^2(\mathbf{z}_i^{*(v)}, \hat{\mathbf{z}})$  and obtain the final  $KNN$  using the weights computed by Equation (28).

3.3. Construct local convex space by solving Equation (31) and obtain the optimal material data  $\hat{\mathbf{z}}_i^{*(v)}$  within the local convex space by Equation (32).

Step 4. Update  $v = v + 1$ . If  $\max_{i=1, \dots, N} d_z^2(\hat{\mathbf{z}}_i^{*(v)}, \hat{\mathbf{z}}_i^{*(v-1)}) > \text{tolerance}$ , repeat Step 2–4.

Step 5. Data-driven solution:  $\mathbf{z}_i^* = \hat{\mathbf{z}}_i^{*(v)}$ ,  $i = 1, \dots, N$ .

**Remark:** Modeling anisotropic materials by classical computational mechanics requires online rotation of element stiffness matrices from local fiber frames to the global reference frame before global assembly, which could lead to high computational cost especially for large systems. In the proposed data-driven modeling of anisotropic materials, online rotation is replaced with an **offline rotation**, where anisotropic material datasets are rotated by various anisotropic orientations under the global reference frame to form a rotated material database for efficient online data-driven computing. In the general three-dimensional orthotropic case, each anisotropic orientation is associated with three Euler angles and thus for every anisotropic orientation considered, offline rotation operations associated with three Euler angles are required for every stress–strain data, which could be CPU intensive but can be performed in parallel especially when dealing with a large amount of data. Once the rotated material database is constructed offline, it can be efficiently applied to different material points in data-driven modeling and to different systems with the same anisotropic materials. Note that when the size of anisotropic material datasets is large and the range of variations in anisotropic orientations of the system is very broad, the offline rotated material database may require a large amount of memory.

If memory resources are not available, one can also adopt online rotation in the data-driven computing for modeling anisotropic materials. In this case, at each data-driven iteration, the physical step remains unchanged. In the material step, the physical states are first rotated from the global reference frame to local fiber frames of material points and then the LCDD material solver (Equations 20–23) can be applied to find the optimal material data from the unrotated anisotropic material dataset. The optimal material data is then rotated back to the global reference frame for the next data-driven iteration. Compared to the proposed data-driven approach with offline rotation, this online rotation approach requires higher CPU but less memory.

### 3. Results and Discussion

#### 3.1. Preparation of material datasets

For the following numerical demonstration, the two-dimensional Saint Venant–Kirchhoff phenomenological model with isotropic and orthotropic elastic tensors are respectively considered as the reference models and used to generate synthetic data for data-driven computing. The plane-stress isotropic elastic stress–strain relation in the Voigt notation is expressed as

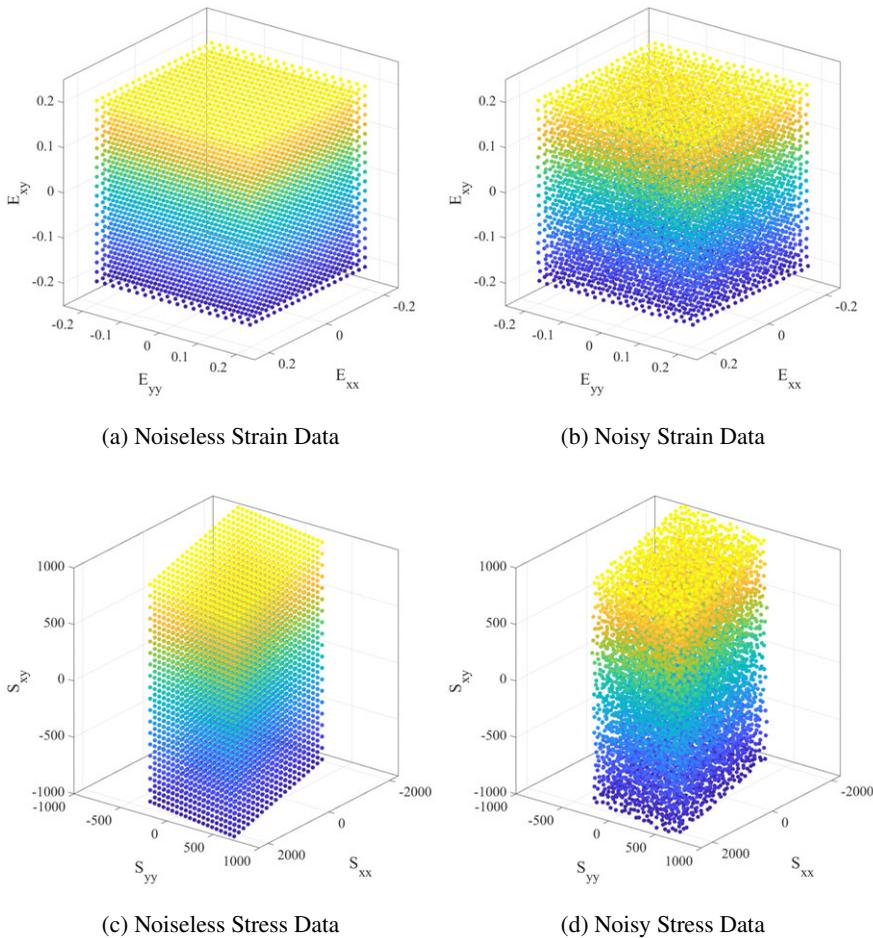
$$\begin{bmatrix} S_{11} \\ S_{22} \\ S_{12} \end{bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \begin{bmatrix} E_{11} \\ E_{22} \\ 2E_{12} \end{bmatrix}, \tag{33}$$

where  $E$  denotes the Young’s modulus and  $\nu$  is the Poisson’s ratio. As another reference material model, the plane-stress orthotropic elastic stress–strain relation in the Voigt notation is expressed as

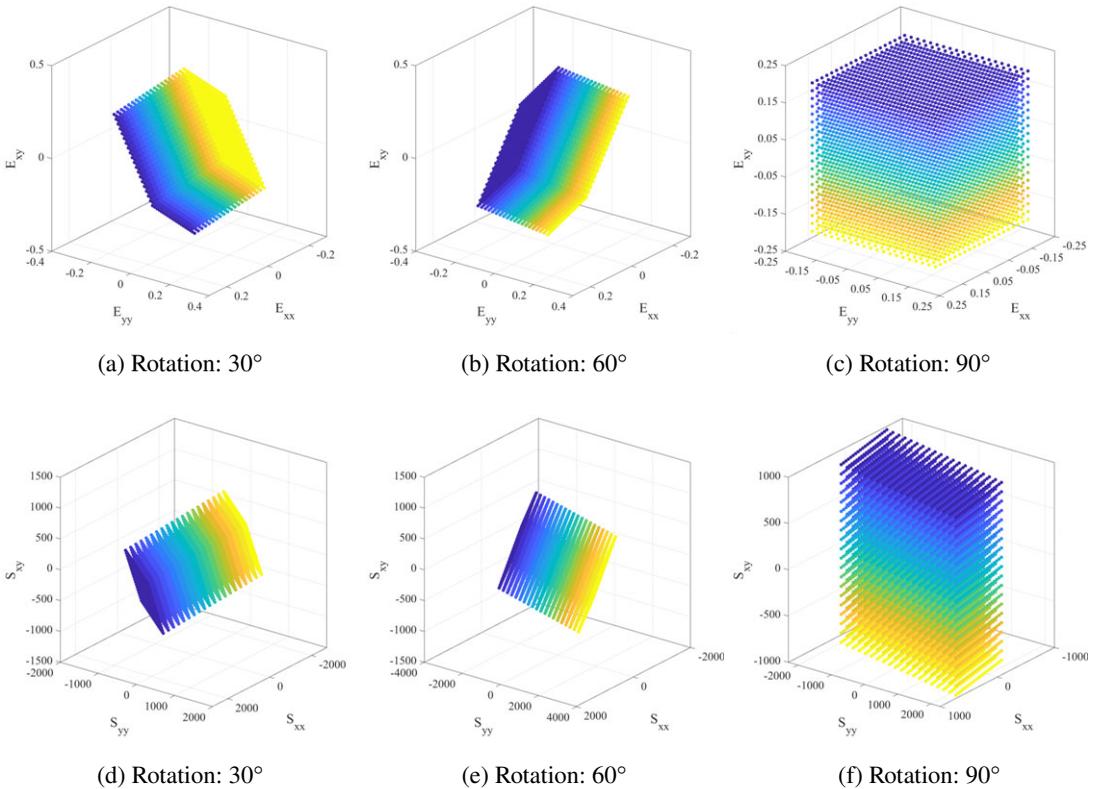
$$\begin{bmatrix} S_{11} \\ S_{22} \\ S_{12} \end{bmatrix} = \frac{1}{1-\nu_{12}\nu_{21}} \begin{bmatrix} E_1 & \nu_{21}E_1 & 0 \\ \nu_{12}E_2 & E_2 & 0 \\ 0 & 0 & G_{12}(1-\nu_{12}\nu_{21}) \end{bmatrix} \begin{bmatrix} E_{11} \\ E_{22} \\ 2E_{12} \end{bmatrix}, \tag{34}$$

where  $E_1$  and  $E_2$  are Young's moduli in the  $\mathbf{e}_1^g$  and  $\mathbf{e}_2^g$  directions of the reference frame, respectively, see Figure 3a.  $\nu_{12}$  and  $\nu_{21}$  are Poisson's ratios.  $G_{12}$  is the shear modulus.

To demonstrate the robustness of the proposed data-driven computing framework against noise presented in given material datasets, noisy material datasets are generated and the procedure is described below. First, a noiseless material dataset,  $\mathbb{E}^0 = \left\{ (\hat{\mathbf{E}}_i^0, \hat{\mathbf{S}}_i^0) \right\}_{i=1}^M$  with  $M = 20^3$ , is generated, where each strain component is uniformly distributed within a certain range, for example,  $[-0.2, 0.2]$ , and the stress components are obtained by using the orthotropic material model in Equation (34).  $M = 20^3$  strain and stress data points with  $E_1 = 10^4, E_2 = 2.5 \times 10^3, \nu_{21} = 0.1, \nu_{12} = 0.4$ , and  $G_{12} = 4.8 \times 10^3$  are shown in Figure 6a,c, respectively, serving as a noiseless base dataset without any noise and rotation. The anisotropic orientation of the noiseless base dataset is along the horizontal direction of the reference frame, as shown in Figure 3a. Then, each component of the noiseless base dataset is perturbed by Gaussian noise with a scaling factor,  $0.4\bar{z}_{max}/\sqrt[3]{M}$ , where  $\bar{z}_{max}$  is a vector of the maximum values for each component among the noiseless dataset. Figure 6b,d show the strain and stress of the noisy base dataset,



**Figure 6.** Material datasets with 8,000 data points,  $E_1 = 10^4, E_2 = 2.5 \times 10^3, \nu_{12} = 0.1, \nu_{21} = 0.4$ , and  $G_{12} = 4.8 \times 10^3$ : (a) noiseless strain data; (b) noisy strain data; (c) noiseless stress data; and (d) noisy stress data.



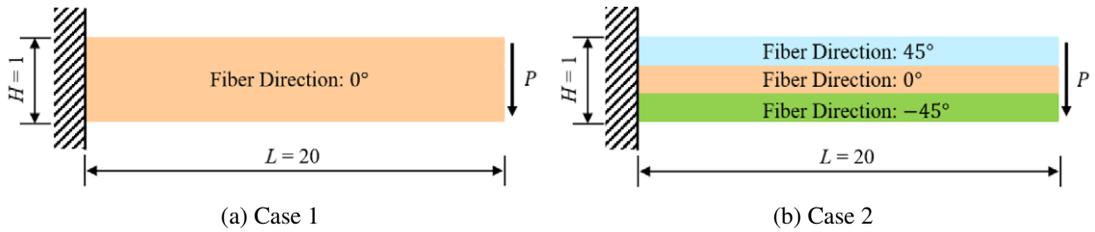
**Figure 7.** Rotated material datasets with 8,000 data points,  $E_1 = 10^4$ ,  $E_2 = 2.5 \times 10^3$ ,  $\nu_{12} = 0.1$ ,  $\nu_{21} = 0.4$ , and  $G_{12} = 4.8 \times 10^3$ : (a) strain data rotated by 30°; (b) strain data rotated by 60°; (c) strain data rotated by 90°; (d) stress data rotated by 30°; (e) stress data rotated by 60°; and (f) stress data rotated by 90°.

$\mathbb{E} = \left\{ (\widehat{\mathbf{E}}_i, \widehat{\mathbf{S}}_i) \right\}_{i=1}^M$  with  $M = 20^3$ , generated from the noiseless base dataset  $\mathbb{E}^0$  shown in Figure 6a,c, respectively.

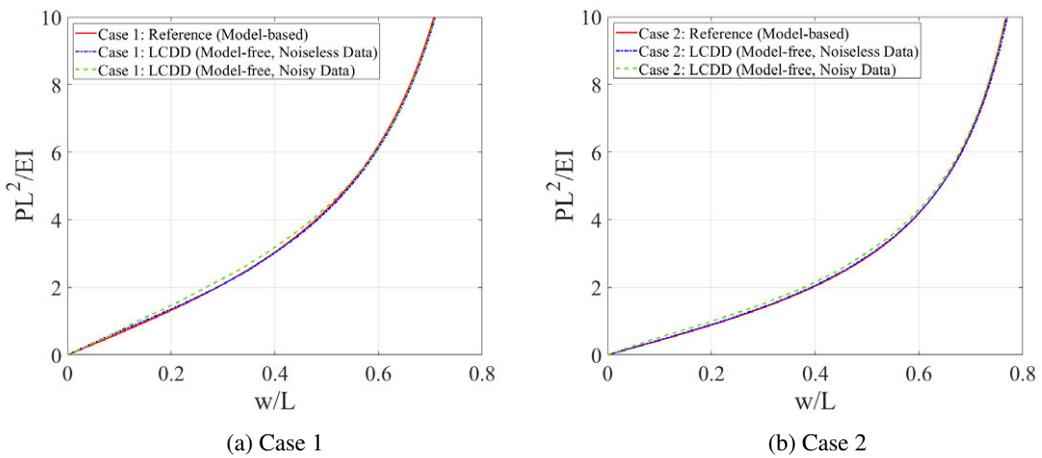
Given anisotropic orientations of material points in a physical system, for example,  $\theta_i$  of the material point  $i$  in Figure 3b, the strain and stress data of the base dataset can be rotated using Equation (25). Figure 7 shows the strain and stress data rotated from the base dataset by three different angles 30°, 60°, and 90°. In the following numerical examples, rotation of material datasets is performed offline by various angles (anisotropic orientations) to cover the range of variations in anisotropic orientations of systems. A collection of rotated material datasets form a rotated material database, which is then used for online data-driven computing with the material solver equipped with two-level data search designed for capturing material’s directional dependence.

### 3.2. Multi-layer anisotropic cantilever beam subjected to a tip shear load

We first investigate a multi-layer anisotropic cantilever beam benchmark problem to verify the proposed data-driven modeling framework for anisotropic nonlinear solids, where three layers of orthotropic materials are considered. Two cases are examined and compared. In Case 1, all material points in the beam have an identical anisotropic orientation, which is along the horizontal direction, as shown in Figure 8a. In Case 2, the beam is made of three layers of anisotropic materials with different anisotropic orientations. The material points within each layer have the same anisotropic orientation. Figure 8b shows that the bottom, the middle, and the top layers of the beam have anisotropic orientations of  $-45^\circ$ ,  $0^\circ$ , and  $45^\circ$ , respectively. A synthetic noiseless



**Figure 8.** Schematic of cantilever beam subjected to a tip shear load: (a) Case 1: material points have only one anisotropic orientation  $0^\circ$ ; (b) Case 2: material points located in different layers of the beam have different anisotropic orientations. The anisotropic orientations of the bottom, the middle, and the top layers are  $-45^\circ$ ,  $0^\circ$ , and  $45^\circ$ , respectively.  $P = 10E_1I/L^2$ , and  $I = H^3/12$ .

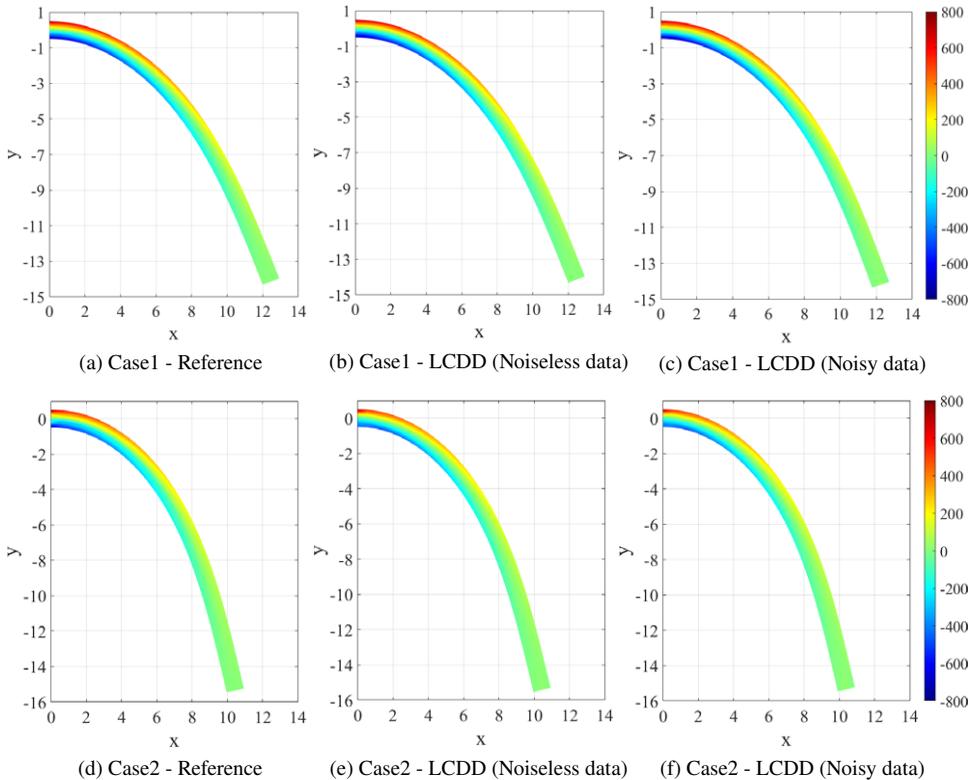


**Figure 9.** Comparison of data-driven solutions with constitutive model-based reference solutions. Normalized tip deflection-loading, where  $I = H^3/12$ : (a) Case 1 and (b) Case 2.

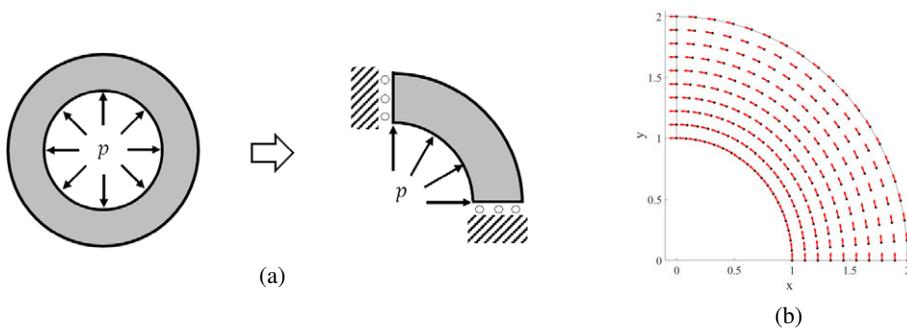
orthotropic material dataset with 8,000 strain–stress data points is first generated from the phenomenological orthotropic elastic model with  $E_1 = 10^4$ ,  $E_2 = 2.5 \times 10^3$ ,  $\nu_{21} = 0.1$ ,  $\nu_{12} = 0.4$ ,  $G_{12} = 4.8 \times 10^3$ , and a range of  $[-0.2, 0.2]$  for each strain component, as shown in Figure 6a,c. Then, a noisy orthotropic material dataset is generated from the noiseless dataset using the procedure described in Section 3.1, as shown in Figure 6b,d.

For Case 1, no rotation of the material dataset is required as the anisotropic orientations of materials points in the beam are identical with that of the synthetic dataset, which is along the horizontal direction. For Case 2, a rotated material database is constructed by rotating the generated synthetic noiseless/noisy dataset from  $-60^\circ$  to  $60^\circ$  with a  $10^\circ$  interval, that is,  $-60^\circ$ ,  $-50^\circ$ ,  $-40^\circ$ ,  $-30^\circ$ ,  $-20^\circ$ ,  $-10^\circ$ ,  $0^\circ$ ,  $10^\circ$ ,  $20^\circ$ ,  $30^\circ$ ,  $40^\circ$ ,  $50^\circ$ , and  $60^\circ$ . The rotated material database is then used for online data-driven computing. The numerical studies of both cases are performed with noiseless or noisy material datasets. A tip shear load  $P = 10E_1I/L^2$  with  $I = H^3/12$  is applied and the data-driven analysis is performed with 20 loading steps.

The normalized tip deflection-loading and stress distribution of data-driven solutions are compared with those of the constitutive model-based reference solutions obtained from an in-house Finite Element code, as shown in Figures 9 and 10. The data-driven solutions of both cases show a satisfactory agreement with the model-based reference solutions, which demonstrates the effectiveness of the proposed data-driven computing framework for modeling anisotropic nonlinear elastic materials. The data-driven solutions obtained by using the noisy material datasets agree well with the model-based reference solutions, showing the robustness of the proposed framework against noise in the datasets, which is achieved by the local convexity-preserving reconstruction in the material solver.



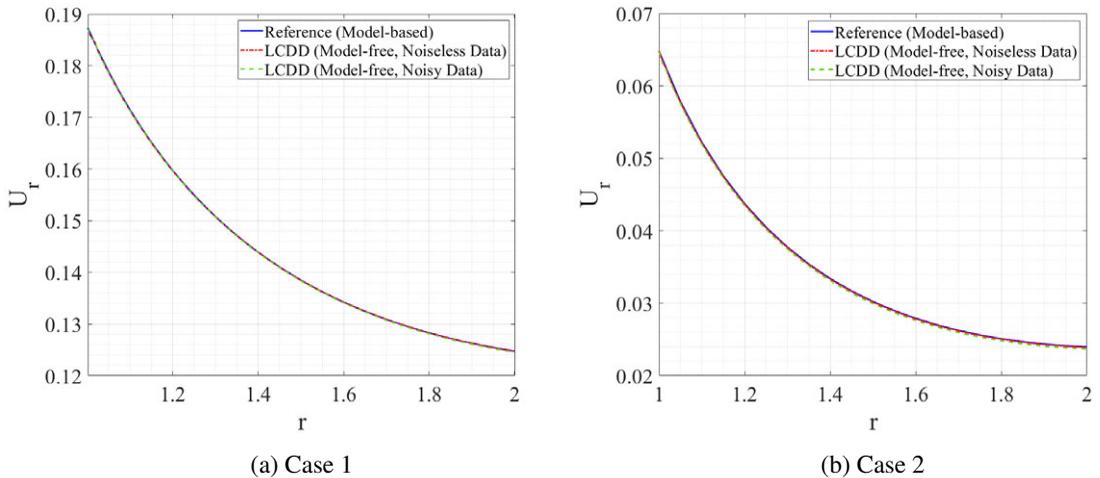
**Figure 10.** Comparison of data-driven solutions with constitutive model-based reference solutions. Distribution of  $S_{xx}$ : (a) Case 1: reference solution; (b) Case 1: data-driven solution with noiseless data; (c) Case 1: data-driven solution with noisy data; (d) Case 2: reference solution; (e) Case 2: data-driven solution with noiseless data; and (f) Case 2: data-driven solution with noisy data.



**Figure 11.** (a) Schematic of a cylinder subjected to internal pressure and a quarter model to be simulated and (b) red arrows denote nodal anisotropic orientations of material points in a discretization with  $10 \times 20$  nodes.

### 3.3. Anisotropic cylinder subjected to internal pressure

To further evaluate the effectiveness of the proposed data-driven computing framework in handling physical systems with a large variation in anisotropic orientations, an anisotropic cylinder subjected to internal pressure is analyzed, as shown in Figure 11a, which is composed of an orthotropic material with



**Figure 12.** Comparison of data-driven solutions with constitutive model-based reference solutions. Cross-sectional radial displacement  $U_r$ : (a) Case 1 and (b) Case 2.

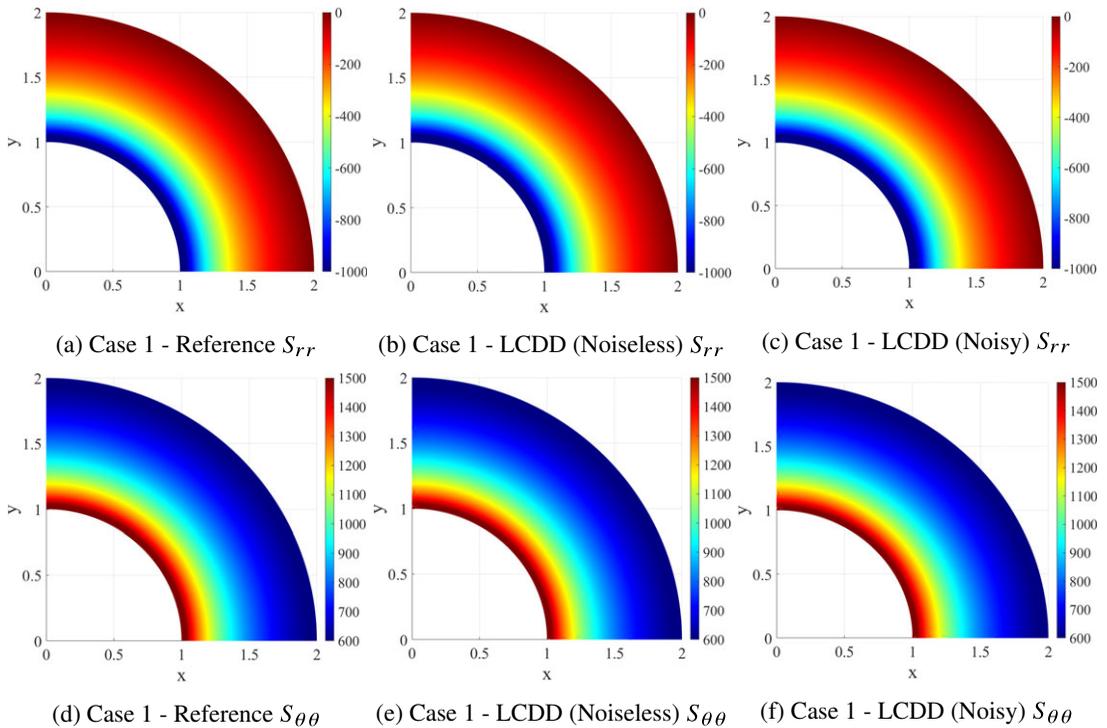
anisotropic orientations along the circumferential direction of the cylinder. Considering axisymmetry of the geometry and the applied load, only a quarter model shown in Figure 11b is modeled. The inner and outer radius are 1 and 2, respectively. In Figure 11c, the anisotropic orientations of the  $10 \times 20$  discretization points are denoted by red arrows. Two cases are examined and compared.

In Case 1, an isotropic elastic material is applied, with Young's modulus  $E = 9 \times 10^3$  and Poisson's ratio  $\nu = 0.2$ . A noiseless synthetic isotropic material dataset with 8,000 strain–stress data points is generated by the phenomenological isotropic elastic material model with a range of  $[-0.4, 0.4]$  for each strain component. In Case 2, an orthotropic elastic material is applied, with  $E_1 = 4 \times 10^4$ ,  $E_2 = 9 \times 10^3$ ,  $\nu_{21} = 0.045$ ,  $\nu_{12} = 0.2$ ,  $G_{12} = 2 \times 10^4$ , and anisotropic orientations along the circumferential direction of the cylinder. A noiseless synthetic orthotropic material dataset with 8,000 strain–stress data points is generated by the phenomenological orthotropic elastic material model with a range of  $[-0.4, 0.4]$  for each strain component. Considering the uniform distribution of anisotropic orientations of material points in the cylinder, a rotated material database is constructed by rotating the generated synthetic dataset from  $90^\circ$  to  $180^\circ$  with a  $5^\circ$  interval. Noisy material datasets are generated from the noiseless material datasets by the procedure described in Section 3.1. Internal pressure  $p = 1,000$  is applied and the data-driven analysis is performed with 20 loading steps.

The cross-sectional ( $y = 0$ ) radial displacement ( $U_r$ ) as well as the radial and the circumferential stress distributions of the data-driven solutions are compared with those of the constitutive model-based reference solutions, as shown in Figures 12–14. The data-driven solutions obtained from using noisy material datasets are very close to those of reference solutions, which again shows the robustness of the proposed data-driven framework to deal with noisy datasets. The results of both cases show a good agreement with the reference solutions, which demonstrates the effectiveness of the proposed data-driven framework in dealing with systems with a large variation in anisotropic orientations.

#### 4. Conclusion

In this study, we develop a new data-driven material solver built upon the local convexity-preserving reconstruction scheme by He and Chen (2020) to capture anisotropic material behaviors and enable data-driven modeling of anisotropic nonlinear elastic solids. The proposed data-driven approach assumes that the material data of anisotropic materials with a specific anisotropic orientation in a reference frame is accessible. The information of anisotropic orientations, for example, the rotation angles between local

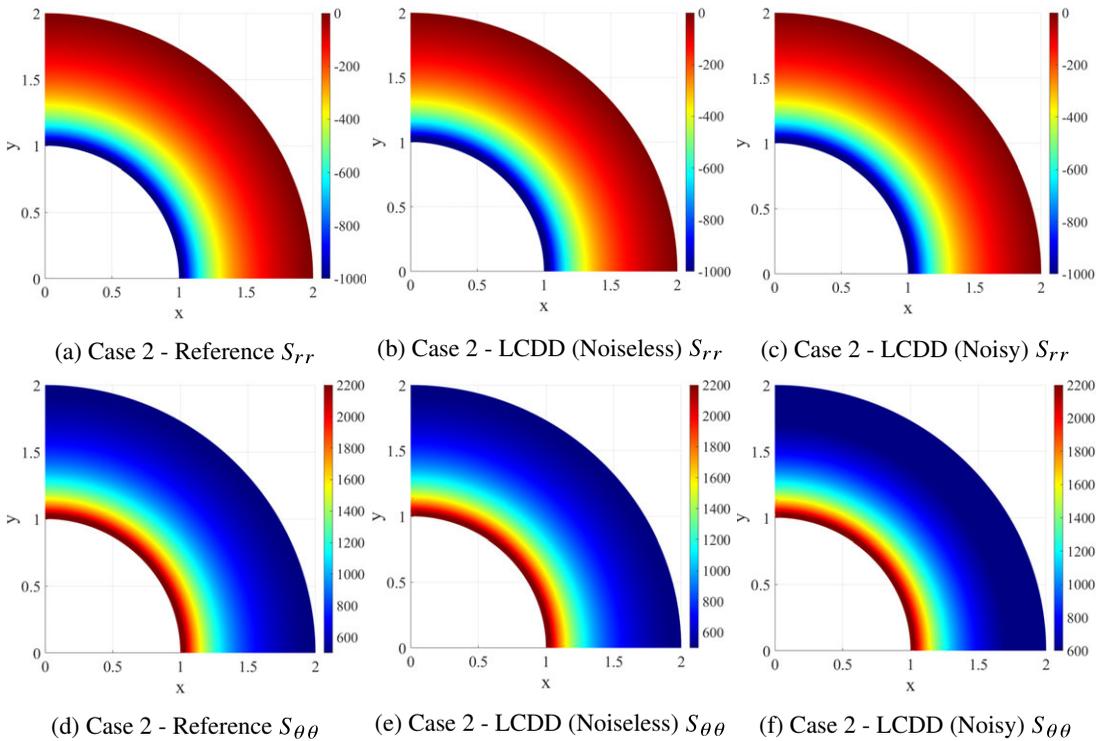


**Figure 13.** Comparison of data-driven solutions with constitutive model-based reference solutions of Case 1. Distribution of  $S_{rr}$  (stress in the radial direction) and  $S_{\theta\theta}$  (stress in the circumferential direction): (a) Case 1: reference  $S_{rr}$ ; (b) Case 1: data-driven solution with noiseless data of  $S_{rr}$ ; (c) Case 1: data-driven solution with noisy data of  $S_{rr}$ ; (d) Case 1: reference  $S_{\theta\theta}$ ; (e) Case 1: data-driven solution with noiseless data of  $S_{\theta\theta}$ ; and (f) Case 1: data-driven solution with noisy data of  $S_{\theta\theta}$ .

fiber frames and the reference frame of the material data are utilized to construct an offline material database, which contains rotated material datasets representing anisotropic material properties with various anisotropic orientations. The offline rotated material database can be efficiently constructed and applied to data-driven simulations of anisotropic materials.

During online data-driven computing, a two-level local data search is integrated into the local convexity-preserving material solver. In the Level 1 data search, two rotated material datasets with minimum anisotropic distance (Equation 27) to the material anisotropic orientation are identified as the local datasets. The selected rotated material datasets are considered to contain anisotropic material properties closest to that of the material point. In the Level 2 data search,  $k$  data points closest to the given physical state based on strain–stress state distance (Equation 5) are obtained separately from the two rotated material datasets selected from the Level 1 data search. A linear weighting scheme based on anisotropic distance is adopted to determine the number nearest data points from each of the selected material datasets. The final  $k$  nearest data points that contain information of strain, stress, and anisotropic orientations are used to construct a local convex space capturing the underlying anisotropic data structure, within which the optimal material data is reconstructed by the material solver. The optimally reconstructed material data is closest to the physical state of the material point in terms of both anisotropic distance and state distance and thus is considered to best represent the anisotropic properties of the material point.

The performance of the proposed data-driven computing framework is examined by two numerical examples, including deflection of a multi-layer anisotropic cantilever beam made of materials with different anisotropic orientations and inflation of an anisotropic cylinder where anisotropic orientations are along the circumferential direction of the cylinder. Synthetic noiseless and noisy material data are



**Figure 14.** Comparison of data-driven solutions with constitutive model-based reference solutions of Case 1. Distribution of  $S_{rr}$  (stress in the radial direction) and  $S_{\theta\theta}$  (stress in the circumferential direction): (a) Case 2: reference  $S_{rr}$ ; (b) Case 2: data-driven solution with noiseless data of  $S_{rr}$ ; (c) Case 2: data-driven solution with noisy data of  $S_{rr}$ ; (d) Case 2: reference  $S_{\theta\theta}$ ; (e) Case 2: data-driven solution with noiseless data of  $S_{\theta\theta}$ ; and (f) Case 2: data-driven solution with noisy data of  $S_{\theta\theta}$ .

generated from the phenomenological material models and employed for the data-driven analysis for accuracy assessment of the proposed method. The data-driven solutions show a good agreement with the constitutive model-based reference solutions, which demonstrates its effectiveness and robustness of the proposed data-driven framework against noise present in the material data.

The proposed two-level data search can achieve high computational efficiency if the rotated material database is constructed offline such that online computation does not involve any frame transformation of states or data. For applications with a large amount of anisotropic material data and strong variations in anisotropic orientations, constructing a rotated material database with small anisotropic distance between rotated datasets requires a large amount of memory. In this case, more sophisticated metrics for anisotropic distance and reconstruction schemes will be investigated in order to achieve high accuracy in reconstructing anisotropic material properties from given material datasets that have a large anisotropic distance. In future studies, the proposed data-driven anisotropic modeling framework will be further examined with real-world data on three-dimensional material systems with anisotropic material behaviors, e.g., musculoskeletal systems consisting of muscle fibers with varying anisotropic orientations.

**Funding Statement.** The support of this work by the National Science Foundation under Award Number CCF-1564302 to the first, second, and the third authors, and the National Institute of Health under grant number 5R01AG056999-03 to the first, third, fourth, and fifth authors is very much appreciated.

**Competing Interests.** The authors declare no competing interests.

**Data Availability Statement.** All data involved in this study are synthetically generated and data generation procedures are described in Section 3.1.

**Author contributions.** Conceptualization, J.-S.C., Q.H., X.H.; Methodology, X.H., Q.H., J.-S.C.; Formal Analysis, X.H.; Data Curation, X.H.; Writing–Original Draft, X.H.; Writing–Review and Editing, X.H., Q.H., J.-S.C., U.S., S.S.; Supervision, J.-S.C.; Funding Acquisition, J.-S.C., U.S., S.S.; All authors approved the final submitted draft.

## References

- Belytschko T, Liu WK and Moran B** (2000) *Nonlinear Finite Elements for Continua and Structures*. Chichester: John Wiley & Sons, Ltd.
- Chen J-S, Marodon C and Hu H-Y** (2015) Model order reduction for meshfree solution of poisson singularity problems. *International Journal for Numerical Methods in Engineering* 102(5), 1211–1237.
- Chen J-S, Pan C, Wu C-T and Liu WK** (1996) Reproducing kernel particle methods for large deformation analysis of non-linear structures. *Computer methods in applied mechanics and engineering* 139(1–4), 195–227.
- Chen J-S, Yoon S and Wu C-T** (2002) Non-linear version of stabilized conforming nodal integration for galerkin mesh-free methods. *International Journal for Numerical Methods in Engineering* 53(12):2587–2615.
- Conti S, Müller S and Ortiz M** (2018) Data-driven problems in elasticity. *Archive for Rational Mechanics and Analysis* 229(1), 79–123.
- Eggersmann R, Kirchdoerfer T, Reese S, Stainier L and Ortiz M** (2019) Model-free data-driven inelasticity. *Computer Methods in Applied Mechanics and Engineering* 350, 81–99.
- Eggersmann R, Stainier L, Ortiz M and Reese S** (2021) Model-free data-driven computational mechanics enhanced by tensor voting. *Computer Methods in Applied Mechanics and Engineering* 373, 113499.
- Furukawa T and Yagawa G** (1998) Implicit constitutive modelling for viscoplasticity using neural networks. *International Journal for Numerical Methods in Engineering* 43(2), 195–219.
- Ghaboussi J, Garrett Jr J and Wu X** (1991) Knowledge-based modeling of material behavior with neural networks. *Journal of Engineering Mechanics* 117(1), 132–153.
- Gillies AR and Lieber RL** (2011) Structure and function of the skeletal muscle extracellular matrix. *Muscle & Nerve* 44(3), 318–331.
- Goswami S, Anitescu C, Chakraborty S and Rabczuk T** (2020) Transfer learning enhanced physics informed neural network for phase-field modeling of fracture. *Theoretical and Applied Fracture Mechanics* 106, 102447.
- He Q, Barajas-Solano D, Tartakovsky G and Tartakovsky AM** (2020a) Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport. *Advances in Water Resources* 141, 103610.
- He Q and Chen J-S** (2020) A physics-constrained data-driven approach based on locally convex reconstruction for noisy database. *Computer Methods in Applied Mechanics and Engineering* 363, 112791.
- He Q, Chen J-S and Marodon C** (2019) A decomposed subspace reduction for fracture mechanics based on the meshfree integrated singular basis function method. *Computational Mechanics* 63(3), 593–614.
- He Q, Laurence DW, Lee C-H and Chen J-S** (2020b) Manifold learning based data-driven modeling for soft biological tissues. *Journal of Biomechanics*, in press.
- Heider Y, Wang K and Sun W** (2020) So (3)-invariance of informed-graph-based deep neural network for anisotropic elastoplastic materials. *Computer Methods in Applied Mechanics and Engineering* 363, 112875.
- Huijing PA** (1999) Muscle as a collagen fiber reinforced composite: a review of force transmission in muscle and whole limb. *Journal of Biomechanics* 32(4), 329–345.
- Ibanez R, Abisset-Chavanne E, Aguado JV, Gonzalez D, Cueto E and Chinesta F** (2018). A manifold learning approach to data-driven computational elasticity and inelasticity. *Archives of Computational Methods in Engineering* 25(1), 47–57.
- Ibañez R, Borzacchiello D, Aguado JV, Abisset-Chavanne E, Cueto E, Ladevèze P and Chinesta F** (2017) Data-driven non-linear elasticity: constitutive manifold construction and problem discretization. *Computational Mechanics* 60(5), 813–826.
- Kirchdoerfer T and Ortiz M** (2016) Data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering* 304, 81–101.
- Kirchdoerfer T and Ortiz M** (2017) Data driven computing with noisy material data sets. *Computer Methods in Applied Mechanics and Engineering* 326, 622–641.
- Kirchdoerfer T and Ortiz M** (2018) Data-driven computing in dynamics. *International Journal for Numerical Methods in Engineering* 113(11), 1697–1710.
- Latorre M and Montáns FJ** (2014) What-You-Prescribe-Is-What-You-Get orthotropic hyperelasticity. *Computational Mechanics* 53(6), 1279–1298.
- Lee C-H and Chen J-S** (2013) RBF-POD reduced-order modeling of DNA molecules under stretching and bending. *Interaction and Multiscale Mechanics* 6(4), 395–409.
- Lee K and Carlberg KT** (2020) Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics* 404, 108973.

- Lefik M and Schrefler BA** (2003) Artificial neural network as an incremental non-linear constitutive model for a finite element code. *Computer Methods in Applied Mechanics and Engineering* 192(28–30), 3265–3283.
- Lei X, Liu C, Du Z, Zhang W and Guo X** (2019) Machine learning-driven real-time topology optimization under moving morphable component-based framework. *Journal of Applied Mechanics* 86(1), 011004.
- Liu WK, Jun S and Zhang YF** (1995) Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids* 20(8–9), 1081–1106.
- Liu Z, Wu C and Koishi M** (2019) A deep material network for multiscale topology learning and accelerated nonlinear modeling of heterogeneous materials. *Computer Methods in Applied Mechanics and Engineering* 345, 1138–1168.
- Maulik R, Lusch B and Balaprakash P** (2020) Reduced-order modeling of advection-dominated systems with recurrent neural networks and convolutional autoencoders. arXiv preprint arXiv:2002.00470.
- Mendizabal A, Márquez-Neila P and Cotin S** (2020) Simulation of hyperelastic materials in real-time using deep learning. *Medical Image Analysis* 59, 101569.
- Mitchell RK, Agle BR and Wood DJ** (1997) Toward a theory of stakeholder identification and salience: Defining the principle of who and what really counts. *Academy of Management Review* 22(4), 853–886.
- Mora-Macías J, Ayensa-Jiménez J, Reina-Romo E, Doweidar M, Domínguez J, Doblare M and Sanz-Herrera J** (2020) A multiscale data-driven approach for bone tissue biomechanics. *Computer Methods in Applied Mechanics and Engineering* 368, 113136.
- Mozaffar M, Bostanabad R, Chen W, Ehmann K, Cao J and Bessa M** (2019) Deep learning predicts path-dependent plasticity. *Proceedings of the National Academy of Sciences* 116(52), 26414–26420.
- Nguyen LTK and Keip M-A** (2018) A data-driven approach to nonlinear elasticity. *Computers & Structures* 194, 97–115.
- Nguyen LTK, Rambauck M and Keip M-A** (2020) Variational framework for distance-minimizing method in data-driven computational mechanics. *Computer Methods in Applied Mechanics and Engineering* 365, 112898.
- Raissi M, Perdikaris P and Karniadakis GE** (2019) Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics* 378, 686–707.
- Rocha I, Kerfriden P and van der Meer F** (2020) Micromechanics-based surrogate models for the response of composites: a critical comparison between a classical mesoscale constitutive model, hyper-reduction and neural networks. *European Journal of Mechanics-A/Solids* 82, 103995.
- Stainier L, Leygue A and Ortiz M** (2019) Model-free data-driven methods in mechanics: material data identification and solvers. *Computational Mechanics* 64(2), 381–393.
- Sussman T and Bathe KJ** (2009) A model of incompressible isotropic hyperelastic material behavior using spline interpolations of tension-compression test data. *Communications in Numerical Methods in Engineering* 25(1), 53–63.
- Takaza M, Moerman KM, Gindre J, Lyons G, and Simms CK** (2013) The anisotropic mechanical behaviour of passive skeletal muscle tissue subjected to large tensile strain. *Journal of the Mechanical Behavior of Biomedical Materials* 17, 209–220.
- Wang K and Sun W** (2018) A multiscale multi-permeability poroplasticity model linked by recursive homogenizations and deep learning. *Computer Methods in Applied Mechanics and Engineering* 334, 337–380.
- Xu R, Yang J, Yan W, Huang Q, Giunta G, Belouettar S, Zahrouni H, Zineb TB and Hu H** (2020) Data-driven multiscale finite element method: From concurrence to separation. *Computer Methods in Applied Mechanics and Engineering* 363, 112893.
- Zhang Y, Chen J-S., He Q, He X, Basava RR, Hodgson J, Sinha U and Sinha S** (2020). Microstructural analysis of skeletal muscle force generation during aging. *International Journal for Numerical Methods in Biomedical Engineering* 36(1), e3295.
- Zhang Y and Ye W** (2019) Deep learning-based inverse method for layout design. *Structural and Multidisciplinary Optimization* 60(2), 527–536.

## Appendix. Reproducing Kernel Particle Method

The reproducing kernel (RK) approximations (Liu et al., 1995; Chen et al., 1996) is adopted in the physical solver of the proposed data-driven framework due to its nodal approximation of state and field variables that are particularly effective for data-driven computing. The RK approximation functions can be constructed to possess desired completeness and continuity, which are determined by basis functions and kernel functions, respectively. Figure 15a shows a domain  $\Omega$  discretized by a set of nodes.

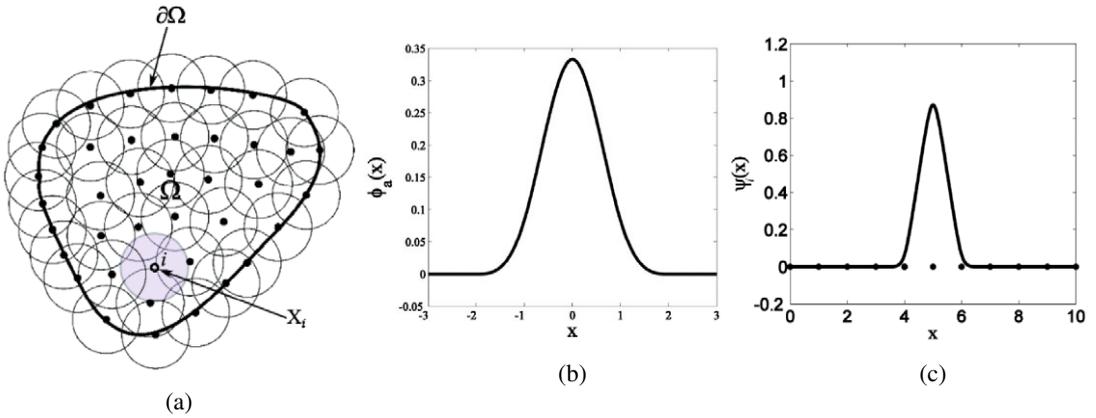
The RK approximation of a field variable  $\mathbf{u}(\mathbf{X})$  is given by

$$\mathbf{u}(\mathbf{X}) \approx \mathbf{u}^h(\mathbf{X}) = \sum_{i=1}^{NP} \Psi_i(\mathbf{X}) \mathbf{d}_i, \quad (35)$$

where  $NP$  is the number of discretization nodes,  $\mathbf{d}_i$  is the nodal coefficient associated with the  $i$ th node, and  $\Psi_i(\mathbf{X})$  is the RK approximation function expressed as

$$\Psi_i(\mathbf{X}) = \mathbf{H}^T(\mathbf{X} - \mathbf{X}_i) \mathbf{b}(\mathbf{X}) \phi_a(\mathbf{X} - \mathbf{X}_i), \quad (36)$$

where  $\mathbf{H}^T(\mathbf{X} - \mathbf{X}_i) = [1, X_1 - X_{1i}, X_2 - X_{2i}, X_3 - X_{3i}, \dots, (X_3 - X_{3i})^n]$  is a vector of monomial basis functions up to  $n$ th order, and  $\phi_a(\mathbf{X} - \mathbf{X}_i)$  is a kernel function with a local support size “ $a$ ,” which controls the smoothness of the RK approximation function. For example, the cubic B-spline kernel function is widely used as a kernel function in RK approximation, see Figure 15b,



**Figure 15.** (a) A domain  $\Omega$  discretized by the a set of RK nodes; (b) a cubic B-spline function widely used as a kernel function in RK approximation; and (c) an example of RK approximation function centered at  $X = 5$  with a support size  $a = 1.5 \times$  (nodal spacing).

$$\phi_a(y) = \begin{cases} \frac{2}{3} - 4y^2 + 4y^3, & 0 \leq y < \frac{1}{2} \\ \frac{4}{3} - 4y + 4y^2 - \frac{4}{3}y^3, & \frac{1}{2} \leq y < 1 \\ 0, & y \geq 1 \end{cases} \text{ with } y = \frac{\|\mathbf{X} - \mathbf{X}_i\|}{a}. \tag{37}$$

The vector  $\mathbf{b}(\mathbf{X})$  in Equation (36) is a parameter vector determined by imposing the  $n$ -th order reproducing conditions (Liu et al., 1995; Chen et al., 1996),

$$\sum_{i=1}^{NP} \Psi_i(\mathbf{X}) X_{1i}^j X_{2i}^j X_{3i}^k = X_1^j X_2^j X_3^k, \quad |i+j+k|=0, 1, \dots, n. \tag{38}$$

Substituting Equation (36) into Equation (38) yields  $\mathbf{b}(\mathbf{X}) = \mathbf{M}^{-1}(\mathbf{X})\mathbf{H}(\mathbf{0})$ , where  $\mathbf{M}(\mathbf{X})$  is a moment matrix given by

$$\mathbf{M}(\mathbf{X}) = \sum_{i=1}^{NP} \mathbf{H}(\mathbf{X} - \mathbf{X}_i) \mathbf{H}^T(\mathbf{X} - \mathbf{X}_i) \phi_a(\mathbf{X} - \mathbf{X}_i). \tag{39}$$

The RK approximation function is expressed as, see Figure 15c,

$$\Psi_i(\mathbf{X}) = \mathbf{H}^T(\mathbf{0}) \mathbf{M}^{-1}(\mathbf{X}) \mathbf{H}(\mathbf{X} - \mathbf{X}_i) \phi_a(\mathbf{X} - \mathbf{X}_i). \tag{40}$$