

On physics-informed data-driven isotropic and anisotropic constitutive models through probabilistic machine learning and space-filling sampling

Jan N. Fuhg^a, Nikolaos Bouklas^{a,b,*}

^a Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853, United States

^b Center for Applied Mathematics, Cornell University, Ithaca, NY 14853, United States

Received 26 September 2021; received in revised form 15 March 2022; accepted 19 March 2022

Available online 9 April 2022

Abstract

Data-driven constitutive modeling is an emerging field in computational solid mechanics with the prospect of significantly relieving the computational costs of hierarchical computational methods. Additionally, this data-driven paradigm could enable a seamless connection of experimental data probing material responses with numerical simulations at the structural level. Traditionally, these surrogates have just been trained using datasets which map strain inputs to stress outputs for elastic and inelastic materials directly. Recently, artificial neural networks (ANNs) have instead been trained to additionally incorporate the underlying physical laws in the construction of these models. However, ANNs do not offer convergence guarantees from an engineering point of view and are majorly reliant on user-specified parameters. In contrast to ANNs, Gaussian process regression (GPR) is based on nonparametric modeling principles as well as on fundamental statistical knowledge and hence allows for strict convergence guarantees. Motivated by the recent work by Frankel et al. (2021) which is based on rewriting the stress output as a linear combination of an irreducible integrity basis, in this work we present a physics-informed and data-driven constitutive modeling approach for isotropic and anisotropic hyperelastic materials at finite strain. The trained surrogates are able to respect physical principles such as material frame indifference, material symmetry, thermodynamic consistency, stress-free undeformed configuration, and the local balance of angular momentum. Our approach is based on probabilistic machine learning and uniquely can be used in the big data context while maintaining the benefits of GPR. As sampling in the mixed invariant space poses a unique challenge, we additionally present the first sampling approach that directly generates space-filling points in the invariant space corresponding to a bounded domain of the deformation gradient tensor. The sampling technique is based on simulated annealing and provides more efficient and reliable physics-informed constitutive models. Overall, the presented approach is tested on synthetic data from isotropic and anisotropic constitutive laws and shows surprising accuracy even far beyond the limits of the training domain, indicating that the resulting surrogates can efficiently generalize as they incorporate knowledge about the underlying physics.

© 2022 Elsevier B.V. All rights reserved.

Keywords: Physics-informed machine learning; Solid mechanics; Hyperelasticity; Finite strain; Data-driven constitutive models

* Correspondence to: 153 Hoy Rd., Ithaca, NY, 14853, United States.

E-mail address: nbouklas@cornell.edu (N. Bouklas).

1. Introduction

There has been an increased interest in machine learning (ML) tools in the computational sciences the last few years. This rise in popularity is due to multiple reasons: the ability of machine learning models to directly utilize experimental data in simulation environments, generalization capabilities of the machine learning tools, potential speed up in comparison to traditional numerical methods and their automatic differentiation framework. For these reasons, machine learning tools have recently been used as a solution scheme for forward and inverse problems involving partial differential equations [1–4] or for the development of intrusive and non-intrusive reduced order modeling schemes for accelerated solutions of PDEs [5–7]. On the other hand the use of ML “black-box” models for constitutive modeling has been extensively studied for over 20 years. Starting from the influential works of Wu and Ghaboussi [8] and Ghaboussi et al. [9,10] for concrete in a small-strain biaxial state of stress, these tools have been employed for different material models with increasing complexity over the years [11–16]. Recently, in the hope of needing less data and in order to generate models with higher generalization capability, efforts have been made to train data-driven constitutive models that do not only train with raw stress–strain data but incorporate additional physics-based restrictions to the trained model [17–20]. When dealing with hyperelastic materials, where no rate-dependence is considered, these models try to include some of the following physics-informed principles:

- Stress-free undeformed configuration: A rigid-body motion induces no strains and consequently no stresses.
- Material frame indifference: Tensor fields such as the stress and the strain should be objective under a change of observer.
- Material symmetry: Strain energy and stresses are consistent with existing symmetry groups in the material.
- Local balance of angular momentum: The Cauchy-stress tensor and the second Piola–Kirchhoff stress tensor should be symmetric.
- Thermodynamic consistency: Fulfillment of the Clausius–Planck inequality.

The idea behind physics-informed or physics-guided data-driven constitutive models is that the trained surrogate should abide to these conditions and not rely solely on raw data. Additionally, a large majority of the proposed works in the literature for physics-guided constitutive models are based on artificial neural networks (ANNs) [17–19,21]. For general information about ANNs we refer to Goodfellow et al. [22]. However, ANNs have some characteristics that make them suboptimal with regard to training constitutive models from data:

1. ANNs are parametric, which means that they are majorly dependent on user-specified parameter values such as hidden layers or the number of neurons per layer. Typically these are chosen based on user experience or in a grid-search fashion. However, this is widely viewed as one of the problematic aspects of ANNs.
2. Even though ANNs are known to be universal approximators [23,24], this is only true if neural networks of either arbitrary width or arbitrary depths are available, which is not common for general engineering applications due to limited computational resources. Hence, there is a clear lack of convergence guarantees with regard to ANNs.
3. ANNs do not have the ability to exactly represent points of the trained dataset due to (typically) being defined as a mean-error minimizer. This is a significant concern for the preservation of the stress-free undeformed configuration and requires particular attention such as special stress representations [25].

Due to these reasons, Gaussian process regression (GPR) also known as Kriging [26] has recently gained more attention as a tool to fit constitutive data. The major factor for this is that in contrast to ANNs, GPR is based on nonparametric modeling principles as well as on fundamental statistical knowledge and hence allows for strict convergence guarantees and to obtain as output probabilistic information such as the mean and variance of the trained model. Recently, the authors [14] have proposed a model-data-driven approach using GPR which enhances analytical constitutive models by local corrections based on data. Rocha et al. [27] present a method relying on the adaptive construction of GPR models with application to elastoplastic multiscale mechanics. Wang et al. [28] use a method based on proper orthogonal decomposition (POD) and GPR known as POD-Kriging to build a surrogate for a time dependent constitutive model for a viscoelastic hydrogel. One solution for probably the most significant problem associated with GPR in the big data regime has recently been proposed by the authors in [29] where local approximate GPR (laGPR) has been presented for constitutive modeling applications and corresponding multiscale calculations. Frankel et al. [30] presented the only work so far where GPR and physics-guided data-driven constitutive modeling is attempted. They propose an approach for isotropic hyperelastic materials which

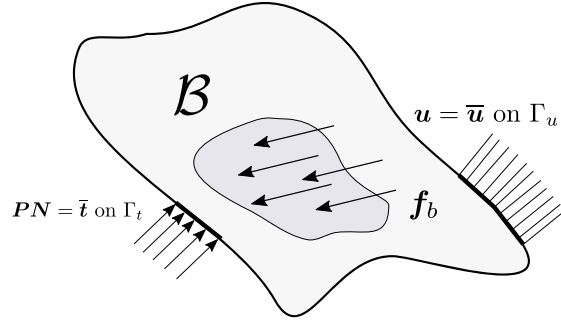


Fig. 1. Solid domain with boundary conditions.

is generally based on building a surrogate model that maps from the space of invariants of the right Cauchy–Green deformation tensor to the space of coefficients linked with the stress generators. In this work we utilize the approach discussed in [30] as a starting point. We generalize it, extend it to anisotropic materials and by employing laGPR we allow the framework to be applicable in the big data context. By doing so we are able to capture five major physical constraints: the preservation of the stress-free undeformed configuration, material frame indifference, material symmetry conditions, thermodynamic consistency and local balance of angular momentum. Furthermore, we introduce the first space-filling sampling approach which directly generates samples in principal and pseudo invariant space. Thereby, creating metamodels which prove to be more efficient.

The paper is structured as follows. The general framework for modeling hyperelastic materials and the corresponding essential physical principles are introduced in Section 2. The physics-informed approach for hyperelastic data-driven constitutive laws for isotropic and anisotropic materials is discussed and explained in Section 3. A space-filling sampling approach in the invariant spaces of isotropic and anisotropic materials is introduced in Section 4. Local approximate Gaussian process regression as well as a consistent form to approximate the material tangent is presented in Section 5. The presented framework is studied on two numerical examples with different material symmetries in Section 6. The paper is concluded in Section 7.

2. Physics-based constraints for data-driven mappings

Consider an elastic body $\mathcal{B} \subset \mathbb{R}^3$. Let the boundary of the body Γ be composed of two parts Γ_t and Γ_u such that $\Gamma = \Gamma_t \cup \Gamma_u$ (Fig. 1). Here, Γ_u and Γ_t describe the boundary sections that displacement and traction boundary conditions are prescribed. The time-dependent motion between the referential position \mathbf{X} and the current position \mathbf{x} can be defined by

$$\mathbf{x} = \boldsymbol{\varphi}(\mathbf{X}, t) = \mathbf{X} + \mathbf{u}(\mathbf{X}, t) \quad (1)$$

where \mathbf{u} describes the time-dependent displacement field and $\boldsymbol{\varphi}(\mathbf{X}, t)$ denotes the motion of the body. This allows to define the deformation gradient

$$\mathbf{F} = \text{Grad}\boldsymbol{\varphi}(\mathbf{X}) \quad (2)$$

and the right Cauchy–Green tensor

$$\mathbf{C} = \mathbf{F}^T \mathbf{F}. \quad (3)$$

From the balance of linear momentum, the local equilibrium equation

$$\text{div}\boldsymbol{\sigma} + \mathbf{b} = \mathbf{0} \quad (4)$$

is fulfilled where $\boldsymbol{\sigma}$ is the Cauchy stress tensor and \mathbf{b} are body forces. The balance of angular momentum leads to the symmetry of the Cauchy stress tensor

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^T. \quad (5)$$

Furthermore, knowing that $\mathbf{S} = \sqrt{\det(\mathbf{C})} \mathbf{F}^{-1} \boldsymbol{\sigma} \mathbf{F}^{-T}$ [31], where \mathbf{S} is the second Piola–Kirchhoff stress tensor, we can see that $\mathbf{S} = \mathbf{S}^T$ must hold as well. Hence, any data-driven constitutive law with \mathbf{S} as an output must ensure its symmetry.

In the hyperelastic framework, the existence of the strain energy function Ψ is postulated, which is assumed to be defined per unit reference volume [31]. The formulation of an explicit strain energy function is dependent on the symmetry group that complies with the symmetries that correspond to a specific material. A symmetry group of a material is a set of transformations that allow for material symmetry to be preserved. The three dimensional orthogonal group **Orth** is defined as the group of 3×3 orthogonal matrices

$$\mathbf{Orth} = \{\mathbf{R} \in \mathbb{R}^3 \otimes \mathbb{R}^3 | \mathbf{R}^T = \mathbf{R}^{-1}\}. \quad (6)$$

Let there be some structural tensors \mathbf{A}_i , $i = 1, \dots, p$ that determine the symmetry group \mathcal{G} of an anisotropic material with

$$\mathcal{G} = \{\mathbf{R} \in \mathbf{Orth} | \mathbf{R}^T \mathbf{A}_i \mathbf{R}, \forall i = 1, \dots, p\}. \quad (7)$$

According to Ehret and Itskov [32] a convenient way to describe the structural tensors is by defining them as

$$\mathbf{A}_i = \mathbf{a}_i \otimes \mathbf{a}_i, \quad i = 1, \dots, p \quad (8)$$

where the $\mathbf{a}_i \in \mathbb{R}^3$ are unit vectors. For a general material consider the strain energy density to be at least a function of the Cauchy–Green tensor

$$\Psi = \Psi(\mathbf{C}, \bullet). \quad (9)$$

Under this assumption a necessary condition for material symmetry in terms of Eq. (7) is given by

$$\Psi(\mathbf{R}^T \mathbf{C} \mathbf{R}) = \Psi(\mathbf{C}), \quad \forall \mathbf{R} \in \mathcal{G}. \quad (10)$$

Following Zhang and Rychlewski [33] and Itskov and Aksel [34], this condition can only be ensured if the strain energy density function has the structural tensors as additional arguments, i.e. $\Psi(\mathbf{C}, \mathbf{A}_i)$, $i = 1, \dots, p$. With regard to the symmetry group \mathcal{G} it is therefore necessary to require

$$\Psi(\mathbf{R}^T \mathbf{C} \mathbf{R}, \mathbf{R}^T \mathbf{A}_i \mathbf{R}) = \Psi(\mathbf{C}, \mathbf{A}_i), \quad i = 1, \dots, p, \quad \forall \mathbf{R} \in \mathcal{G} \quad (11)$$

for material symmetry. Data-driven constitutive laws should therefore aim to satisfy Eq. (11). Even though it is not a theoretical requirement, hyperelastic constitutive laws are constructed so that they satisfy that the reference configuration corresponds to a stress-free state with zero strain energy density. This can be achieved by requiring that

$$\Psi(\mathbf{I}, \bullet) = 0 \quad (12)$$

in the reference configuration where $\mathbf{F} = \mathbf{C} = \mathbf{I}$. This condition and the physically sound assumption that the strain energy function increases under deformation, i.e. $\Psi(\mathbf{C}, \bullet) \geq 0$, ensures that the stress in the reference configuration is zero, i.e.

$$\mathbf{S}(\mathbf{I}, \bullet) = \mathbf{0} \quad (13)$$

which is another physical constraint for data-driven material models. If a scalar-valued tensor function, such as the strain energy density function, is invariant under a rotation it may be written in terms of the invariants (I_1, \dots, I_J) of its arguments [31]. For purely mechanical processes of perfectly elastic materials, the second law of thermodynamics requires that the Clausius–Planck inequality turns into an equality, satisfying

$$\mathbf{S} : \dot{\mathbf{E}} - \dot{\Psi} = 0 \quad (14)$$

which with arbitrary strain measures requires that

$$\begin{aligned} \mathbf{S} &= \Phi(\mathbf{C}, \mathbf{A}_1, \dots, \mathbf{A}_p) = 2 \frac{\partial \Psi(\mathbf{C}, \mathbf{A}_1, \dots, \mathbf{A}_p)}{\partial \mathbf{C}} \\ &= 2 \frac{\partial \Psi(I_1, \dots, I_J)}{\partial \mathbf{C}} = 2 \sum_{i=1}^J \frac{\partial \Psi}{\partial I_i} \frac{\partial I_i}{\partial \mathbf{C}}, \end{aligned} \quad (15)$$

where Φ denotes an analytical tensor-valued tensor function. Hence, any data-driven constitutive model that fulfills this relationship is thermodynamically consistent, at least for the case of perfectly elastic materials.

Another important requirement for data-driven material models is material frame indifference, i.e. independence of the observer, which reads [35]

$$\Phi(\mathbf{R}^T \mathbf{C} \mathbf{R}, \mathbf{R}^T \mathbf{A}_1 \mathbf{R}, \dots, \mathbf{R}^T \mathbf{A}_p \mathbf{R}) = \mathbf{R}^T \Phi(\mathbf{C}, \mathbf{A}_1, \dots, \mathbf{A}_p) \mathbf{R}, \quad \forall \mathbf{R} \in \text{Orth}. \quad (16)$$

From the last term of Eq. (15) it can be seen that the stress can be written as a linear combination of some tensors $\frac{\partial I_i}{\partial \mathbf{C}}$. These terms can be condensed to build an integrity basis \mathbb{G} for the stress, constructed from some symmetric components $\mathbf{H}_i \in \mathbb{R}^{3 \times 3}$, $i = 1, \dots, D$, as

$$\mathbb{G} = \{\mathbf{H}_1, \dots, \mathbf{H}_D\}. \quad (17)$$

This in turn allows us to write the functional mapping of the stress $\mathbf{S} = \Phi(\mathbf{C}, \mathbf{A}_1, \dots, \mathbf{A}_p)$ as a linear combination of the components of the integrity basis as

$$\mathbf{S} = c_1(I_1, \dots, I_J) \mathbf{H}_1(\mathbf{C}, \mathbf{A}_1, \dots, \mathbf{A}_p) + \dots + c_D(I_1, \dots, I_J) \mathbf{H}_D(\mathbf{C}, \mathbf{A}_1, \dots, \mathbf{A}_p). \quad (18)$$

If none of the components of \mathbb{G} is expressible as a linear combination of the others this representation is called an irreducible representation. It can be shown that any representation of the form of Eq. (18) fulfills the condition of material frame indifference, i.e. Eq. (16), as well [36]. Additionally, by construction any stress generated by Eq. (18) respects the material symmetry condition of Eq. (11) [36,37]. One problem with this approach is that an irreducible integrity basis has to be known. However, in the last 50 years a lot of effort has been put into finding suitable irreducible function basis for different kinds of material anisotropies. In particular, Zheng [36] presented a unified invariant approach for the representation of tensor functions for different cases of anisotropy (isotropy, hemitropy, transversal isotropy, orthotropy) in two and three dimensions. The findings of this work have been extensively studied and applied [34,38,39]. Therefore, the integrity bases for standard anisotropic cases are known.

Now consider a material model dataset given by strain inputs and stress outputs of the form

$$\mathcal{D} = \{\mathbf{C}^i, \mathbf{S}^i\}_{i=1}^N \quad (19)$$

and assume that the material anisotropy and corresponding structural tensors $(\mathbf{A}_1, \dots, \mathbf{A}_p)$ are known. This allows us to obtain the relevant invariants for every input (I_1^i, \dots, I_J^i) . Furthermore assume that by knowing the \mathbf{C}_i and \mathbf{S}_i the respective coefficients (c_1^i, \dots, c_D^i) of Eq. (18) can be obtained. Hence, we can generate the alternative dataset

$$\mathcal{D}_{alt} = \{[I_1^i, \dots, I_J^i], [c_1^i, \dots, c_D^i]\}_{i=1}^N \quad (20)$$

and build a surrogate model

$$\hat{\mathcal{M}} : \mathcal{I} \in \mathbb{R}^J \rightarrow c \in \mathbb{R}^D \quad (21)$$

for this dataset. This metamodel allows us to obtain an approximation of the stress output with

$$\hat{\mathbf{S}}^i = \hat{c}_1^i(I_1^i, \dots, I_J^i) \mathbf{H}_1^i + \dots + \hat{c}_D^i(I_1^i, \dots, I_J^i) \mathbf{H}_D^i \quad (22)$$

where \hat{c}_j^i refers to the j th output of the trained metamodel for the i th input. Here, the symbol $\hat{\bullet}$ indicates an approximated value.

Following the mapping approach of Eq. (21) we can easily incorporate the following physics-based constraints into a data-driven constitutive model:

- Local balance of angular momentum: Because the integrity basis is necessarily symmetric,
- Material frame indifference : By estimating stress with Eq. (22),
- Material symmetry conditions: By design the stress output of Eq. (22) respects the symmetry of isotropic and anisotropic materials,
- Stress-free undeformed configuration: When one set of input invariants are chosen that correspond to the undeformed configuration and a surrogate modeling technique with exact inference (such as GPR) is used,
- Thermodynamic consistency: By approximating the stress with Eq. (22) thermodynamic consistency holds at the training points when a metamodeling method with exact inference properties is used.

In the following we explicitly highlight the details of this mapping approach for isotropic and transversally isotropic materials and compare it to the predominant approach of training a metamodel through strain and stress data.

3. Physics-informed mapping for hyperelastic materials

In this section we discuss and highlight different variations of data-preprocessing for hyperelastic laws.

3.1. Classical mapping approach

The classical mapping approach, see e.g. [40–42], is based on the symmetry condition of the right Cauchy–Green tensor and the second Piola–Kirchhoff stress. This allows to postulate a mapping between upper triangular components of the two tensors, i.e.

$$\begin{bmatrix} C_{11} \\ C_{12} \\ C_{13} \\ C_{22} \\ C_{23} \\ C_{33} \end{bmatrix} \rightarrow \begin{bmatrix} S_{11} \\ S_{12} \\ S_{13} \\ S_{22} \\ S_{23} \\ S_{33} \end{bmatrix}. \quad (23)$$

This principle can equivalently be applied using other symmetric strain and stress measures, e.g. engineering strain and Cauchy stress. If the data is only available in a non-symmetric tensor form, for example as the deformation gradient and the first Piola–Kirchhoff stress, then they can simply be converted to equivalent symmetric tensors. The mapping of Eq. (23) is independently utilizable regardless of any type of anisotropy implicitly present in the data and allows to easily capture two physics informed-principles: local balance of angular momentum and preservation of the stress-free undeformed configuration.

3.2. Physics-informed mapping approach for isotropic materials

For the isotropic case we follow we the approach proposed by Frankel et al. [30]. The isotropic case is fully defined by the 3 invariants

$$\begin{aligned} I_1 &= \text{tr}(\mathbf{C}) \\ I_2 &= 0.5(\text{tr}(\mathbf{C})^2 - \text{tr}(\mathbf{C}^2)) \\ I_3 &= \det(\mathbf{C}). \end{aligned} \quad (24)$$

The second Piola–Kirchhoff stress response of an isotropic material can always be decomposed into the three stress generators [31]

$$\mathbb{G} = \{\mathbf{I}, \mathbf{C}, \mathbf{C}^{-1}\} \quad (25)$$

with

$$\mathbf{S} = c_1(I_1, I_2, I_3)\mathbf{I} + c_2(I_1, I_2, I_3)\mathbf{C} + c_3(I_1, I_2, I_3)\mathbf{C}^{-1}. \quad (26)$$

Therefore instead of learning a mapping between the symmetric components of \mathbf{C} and \mathbf{S} as described in Eq. (23), we can learn a functional mapping of the form

$$\begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} \rightarrow \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}. \quad (27)$$

Hence, instead of mapping $\mathbb{R}^6 \rightarrow \mathbb{R}^6$, we map $\mathbb{R}^3 \rightarrow \mathbb{R}^3$. Since all the stress generators of Eq. (25) are symmetric, any output of Eq. (26) is symmetric as well, and hence, the local balance of angular momentum is ensured. Furthermore, the presented mapping yields an output which necessitates the fulfillment of material symmetry and material frame indifference. To prove the latter, consider the rotation tensor \mathbf{R} , then we see that the following needs to hold

$$\mathbf{S}' = \mathbf{R}\mathbf{S}\mathbf{R}^T = \mathbf{R}\Phi(\mathbf{C})\mathbf{R}^T = \Phi(\mathbf{R}\mathbf{C}\mathbf{R}^T). \quad (28)$$

This can be proven by using the tensor generators

$$\begin{aligned}
 \mathbf{S}' &= \mathbf{R} \mathbf{S} \mathbf{R}^T \\
 &= \mathbf{R}(c_1 \mathbf{I} + c_2 \mathbf{C} + c_3 \mathbf{C}^{-1}) \mathbf{R}^T \\
 &= c_1 \mathbf{R} \mathbf{R}^T + c_2 \mathbf{R} \mathbf{C} \mathbf{R}^T + c_3 \mathbf{R} \mathbf{C}^{-1} \mathbf{R}^T \\
 &= \Phi(\mathbf{R} \mathbf{C} \mathbf{R}^T).
 \end{aligned} \tag{29}$$

The scalar values c_1, c_2, c_3 can be obtained by observing that

$$\mathbf{Q} \mathbf{S} \mathbf{Q}^T = \begin{bmatrix} \lambda_1^s & 0 & 0 \\ 0 & \lambda_2^s & 0 \\ 0 & 0 & \lambda_3^s \end{bmatrix}, \quad \mathbf{Q} \mathbf{C} \mathbf{Q}^T = \begin{bmatrix} \lambda_1^c & 0 & 0 \\ 0 & \lambda_2^c & 0 \\ 0 & 0 & \lambda_3^c \end{bmatrix} \tag{30}$$

which following Frankel et al. [30] allows to define an equation system for the unknown scalar values of the form

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 1 & \lambda_1^c & \frac{1}{\lambda_1^c} \\ 1 & \lambda_2^c & \frac{1}{\lambda_2^c} \\ 1 & \lambda_3^c & \frac{1}{\lambda_3^c} \end{bmatrix}^{-1} \begin{bmatrix} \lambda_1^s \\ \lambda_2^s \\ \lambda_3^s \end{bmatrix}. \tag{31}$$

The matrix might be severely ill-conditioned depending on the number of unique principal strains. Frankel et al. [30] describe an algorithmic way to avoid this problem.

The consistent material tangent of the general stress formulation of Eq. (26) is given by

$$\begin{aligned}
 \mathbb{C} &= 2 \frac{\partial \mathbf{S}}{\partial \mathbf{C}} \\
 &= 2 \left(\frac{\partial c_1}{\partial \mathbf{C}} \otimes \mathbf{I} + c_1 \frac{\partial \mathbf{I}}{\partial \mathbf{C}} + \frac{\partial c_2}{\partial \mathbf{C}} \otimes \mathbf{C} + c_2 \frac{\partial \mathbf{C}}{\partial \mathbf{C}} + \frac{\partial c_3}{\partial \mathbf{C}} \otimes \mathbf{C}^{-1} + c_3 \frac{\partial \mathbf{C}^{-1}}{\partial \mathbf{C}} \right)
 \end{aligned} \tag{32}$$

where

$$\frac{\partial c_i}{\partial \mathbf{C}} = \frac{\partial c_i}{\partial I_1} \mathbf{I} + \frac{\partial c_i}{\partial I_2} (I_1 \mathbf{I} - \mathbf{C}) + \frac{\partial c_i}{\partial I_3} I_3 \mathbf{C}^{-1}. \tag{33}$$

All remaining unknown derivatives of Eq. (32) can be found in Eq. (76). Additionally, from Eq. (32) it can be seen that if a surrogate model is trained taking as input the principal invariants of \mathbf{C} and as output the scalar coefficients of Eq. (26), taking the output derivative with regard to the input ($\frac{\partial c_i}{\partial I_j}$) allows us to obtain an approximation of the consistent material tangent. Both ANNs and GPR have the ability to obtain these derivatives. Other surrogate modeling techniques might need to rely on numerical differentiation, e.g. in the form of finite difference schemes [43].

3.3. Physics-informed mapping approach for transversely isotropic materials

Transverse isotropy is characterized by a single unit direction \mathbf{a}_0 that characterizes the material symmetries. Hence, only one structural tensor $\mathbf{A} = \mathbf{a}_0 \otimes \mathbf{a}_0$ is needed to fulfill the material symmetry of the strain energy function. The principal invariants of the right Cauchy–Green tensor read

$$\begin{aligned}
 I_1 &= \text{tr}(\mathbf{C}) \\
 I_2 &= 0.5(\text{tr}(\mathbf{C})^2 - \text{tr}(\mathbf{C}^2)) \\
 I_3 &= \det(\mathbf{C}).
 \end{aligned} \tag{34}$$

We furthermore need to consider the two independent components of the pseudo invariants

$$\begin{aligned}
 I_4 &= \text{tr}(\mathbf{A} \mathbf{C}) \\
 I_5 &= \text{tr}(\mathbf{A} \mathbf{C}^2).
 \end{aligned} \tag{35}$$

Following Zheng [36] the stress output can be decomposed into the six generators collected in the set \mathbb{G}

$$\mathbb{G} = \{\mathbf{I}, \mathbf{C}, \mathbf{A}, \mathbf{C}^2, (\mathbf{A} \mathbf{C} + \mathbf{C} \mathbf{A}), (\mathbf{A} \mathbf{C}^2 + \mathbf{C}^2 \mathbf{A})\} \tag{36}$$

with

$$\mathbf{S} = c_1 \mathbf{I} + c_2 \mathbf{C} + c_3 \mathbf{A} + c_4 \mathbf{C}^2 + c_5 (\mathbf{A}\mathbf{C} + \mathbf{C}\mathbf{A}) + c_6 (\mathbf{A}\mathbf{C}^2 + \mathbf{C}^2 \mathbf{A}). \quad (37)$$

Therefore, in the transversely isotropic case we can learn a function mapping from

$$\begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \end{bmatrix} \rightarrow \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix}. \quad (38)$$

So instead of mapping $\mathbb{R}^6 \rightarrow \mathbb{R}^6$ as in the classical mapping case, we map $\mathbb{R}^5 \rightarrow \mathbb{R}^6$ and are able to ensure local balance of angular momentum (all stress generators are symmetric) and maintain material symmetry conditions, thermodynamic consistency and material frame indifference. The latter can easily be proven by considering

$$\mathbf{S}' = \mathbf{R}\mathbf{S}\mathbf{R}^T = \mathbf{R}\Phi(\mathbf{C}, \mathbf{A})\mathbf{R}^T = \Phi(\mathbf{R}\mathbf{C}\mathbf{R}^T, \mathbf{R}\mathbf{A}\mathbf{R}^T) \quad (39)$$

and

$$\begin{aligned} \mathbf{S}' &= \mathbf{R}\mathbf{S}\mathbf{R}^T \\ &= \mathbf{R} (c_1 \mathbf{I} + c_2 \mathbf{C} + c_3 \mathbf{A} + c_4 \mathbf{C}^2 + c_5 (\mathbf{A}\mathbf{C} + \mathbf{C}\mathbf{A}) + c_6 (\mathbf{A}\mathbf{C}^2 + \mathbf{C}^2 \mathbf{A})) \mathbf{R}^T \\ &= c_1 \mathbf{R}\mathbf{R}^T + c_2 \mathbf{R}\mathbf{C}\mathbf{R}^T + c_3 \mathbf{R}\mathbf{A}\mathbf{R}^T + c_4 \mathbf{R}\mathbf{C}^T \mathbf{R}^T \mathbf{R}\mathbf{C}\mathbf{R}^T \\ &\quad + c_5 (\mathbf{R}\mathbf{A}\mathbf{R}^T \mathbf{R}\mathbf{C}\mathbf{R}^T + \mathbf{R}\mathbf{C}\mathbf{R}^T \mathbf{R}\mathbf{A}\mathbf{R}^T) \\ &\quad + c_6 (\mathbf{R}\mathbf{A}\mathbf{R}^T \mathbf{R}\mathbf{C}^T \mathbf{R}^T \mathbf{R}\mathbf{C}\mathbf{R}^T + \mathbf{R}\mathbf{C}^T \mathbf{R}^T \mathbf{R}\mathbf{C}\mathbf{R}^T \mathbf{R}\mathbf{A}\mathbf{R}^T) \\ &= \Phi(\mathbf{R}\mathbf{C}\mathbf{R}^T, \mathbf{R}\mathbf{A}\mathbf{R}^T). \end{aligned} \quad (40)$$

Obtaining the scalars for the isotropic case was straightforward using principal spaces. However, for the transversely anisotropic case this would result in an underdetermined equation system. Hence, given a stress output and a corresponding right Cauchy–Green tensor value the scalar values c_1, \dots, c_6 can instead be approximated using a least squares approach. By rewriting Eq. (37) we get the overdetermined equation system

$$\underbrace{[\text{vc}(\mathbf{S})]}_b = \underbrace{[\text{vc}(\mathbf{I}) \quad \text{vc}(\mathbf{C}) \quad \text{vc}(\mathbf{A}) \quad \text{vc}(\mathbf{C}^2) \quad \text{vc}(\mathbf{A}\mathbf{C} + \mathbf{C}\mathbf{A}) \quad \text{vc}(\mathbf{A}\mathbf{C}^2 + \mathbf{C}^2 \mathbf{A})]}_A \underbrace{\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \end{bmatrix}}_x \quad (41)$$

where the function $\text{vc}(\bullet)$ performs a vectorization of the tensors. By defining the following optimization problem we can constrain the solution space to the linear least squares solution

$$\min_x \quad \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 \quad (42)$$

which can for example be solved with QR-decomposition. As a sidenote, instead of setting up a fully-determined equation system with six symmetric components of Eq. (37), we found that adding all 9 possible equations makes the solution algorithm more robust.

Similarly to the isotropic case, the consistent material tangent for the transversally isotropic case can also be derived analytically and is only dependent on an approximation of the derivatives $(\frac{\partial c_i}{\partial I_j})$ which can be obtained straightforwardly with some surrogate modeling techniques (such as Gaussian process regression or neural networks) since predictions are made using closed form analytical formulations. Derivatives of these can for example be found using automatic differentiation Baydin et al. [44].

The tangent is given as

$$\begin{aligned}\mathbb{C} &= 2 \frac{\partial \mathcal{S}}{\partial \mathbf{C}} \\ &= 2 \left(\frac{\partial c_1}{\partial \mathbf{C}} \otimes \mathbf{I} + c_1 \frac{\partial \mathbf{I}}{\partial \mathbf{C}} + \frac{\partial c_2}{\partial \mathbf{C}} \otimes \mathbf{C} + c_2 \frac{\partial \mathbf{C}}{\partial \mathbf{C}} + \frac{\partial c_3}{\partial \mathbf{C}} \otimes \mathbf{A} + c_3 \frac{\partial \mathbf{A}}{\partial \mathbf{C}} \right. \\ &\quad + \frac{\partial c_4}{\partial \mathbf{C}} \otimes \mathbf{C}^2 + c_4 \frac{\partial \mathbf{C}^2}{\partial \mathbf{C}} \frac{\partial c_5}{\partial \mathbf{C}} \otimes (\mathbf{AC} + \mathbf{CA}) + c_5 \frac{\partial (\mathbf{AC} + \mathbf{CA})}{\partial \mathbf{C}} \\ &\quad \left. + \frac{\partial c_6}{\partial \mathbf{C}} \otimes (\mathbf{AC}^2 + \mathbf{C}^2 \mathbf{A}) + c_6 \frac{\partial (\mathbf{AC}^2 + \mathbf{C}^2 \mathbf{A})}{\partial \mathbf{C}} \right) \end{aligned} \quad (43)$$

where

$$\begin{aligned}\frac{\partial c_i}{\partial \mathbf{C}} &= \frac{\partial c_i}{\partial I_1} \mathbf{I} + \frac{\partial c_i}{\partial I_2} (I_1 \mathbf{I} - \mathbf{C}) + \frac{\partial c_i}{\partial I_3} I_3 \mathbf{C}^{-1} + \frac{\partial c_i}{\partial I_4} \mathbf{A} \\ &\quad + \frac{\partial c_i}{\partial I_5} (\mathbf{a}_0 \otimes \mathbf{C} \mathbf{a}_0 + \mathbf{a}_0 \mathbf{C} \otimes \mathbf{a}_0). \end{aligned} \quad (44)$$

All unknown tensor derivatives of Eq. (43) are listed in Eq. (76).

As a sidenote, orthotropic materials are not explicitly discussed in this paper. However they have seven invariants and seven generators [36]. Hence, the scalar values of orthotropic materials can be obtained in a similar manner to Eq. (42). After this the setup of the data-driven constitutive model approach for this anisotropic material class is straightforward. The next section proposes an approach of sampling new points in the invariant space.

4. Space-filling sampling approach in invariant space

A problem associated with training a mapping where inputs are principal and pseudo invariants of the right Cauchy–Green deformation tensor, is that the sample placement in this space might not be spaced evenly even when sampling the deformation tensors in a space-filling way. Furthermore, generating samples in a space-filling fashion in a specific domain of the invariant space has not been sufficiently explored since it is not clear how to define a relevant region. This is important in order to evaluate how much trust we put into the output of a metamodel, i.e. if we have sampled points inside a restricted domain of the deformation gradient space and we train our surrogate with these points we expect the model to predict the correct output somewhat accurately if the new input point is inside the bounded training domain. However, outside this training domain we should not blindly trust the predicted surrogate output. Hence, when using points in the invariant space as the model input we need to be able to understand which bounded domain of the deformation gradient space we are representing with our input data in order to judge how trustful a trained model is with regard to a certain deformation gradient input as is needed when employing the trained model in a FEM framework.

To highlight all these points, consider that a trained model should be able to accurately predict the constitutive law when the bounds of the deformation gradient components are given by

$$\overline{F}_{ij} \in [F_{ij}^L, F_{ij}^U] \text{ where } \begin{cases} 1 - \delta \leq 1 \leq 1 + \delta, & \text{when } i = j \\ -\delta \leq 0 \leq \delta, & \text{when } i \neq j. \end{cases} \quad (45)$$

with $\delta > 0$. This defines a nine-dimensional bounded space that samples need to be generated in. In order to generate distributed points in a bounded space in computational engineering applications, latin hypercube sampling (LHS) [45] or some form of optimal latin hypercube sampling (e.g. latin hypercube samples obtained with a translational propagation algorithm (TPLHD) [46]) are typically applied. In order to highlight the problems associated with generating evenly-spaced samples in the deformation gradient space, i.e. in the bounds of Eq. (45), and transferring these samples into principal or pseudo invariant space (using Eq. (34) or Eq. (35) equivalently) assume in the following that $\delta = 0.175$. Further on, we will refer to this training domain as the 17.5% input domain.

Next, we want to visually inspect the bounded domain that the 17.5% input region in deformation gradient space corresponds to in the principal invariant space. For this task we generate 50,000 samples in the bounds of Eq. (45) with LHS and plot them in a scatter plot, see blue dots in Fig. 2. It can be seen that the effective area in invariant

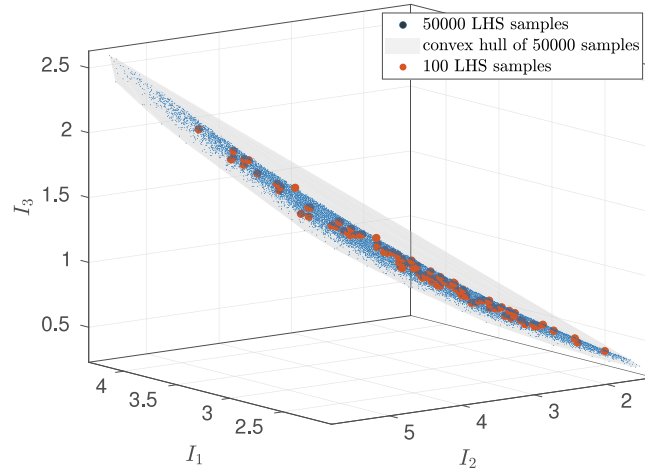


Fig. 2. Example of unevenly spread samples in principal invariant space when mapping from the space of the deformation gradient space which was evenly-sampled with LHS in the 17.5% input domain. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

space stretches over a narrow 3D band. Next, we generate 100 samples in the deformation gradient space with LHS and transfer them into invariant space, see red dots in Fig. 2. We can observe that the resulting points are unevenly distributed which will likely result in poor surrogate model performances in areas with low sample point density. There is one additional problem associated with sampling in deformation gradient space as the mapping between the two spaces is not injective, i.e. that two seemingly different samples in the deformation gradient space might point towards exactly the same point in invariant space. Physically this can be understood by two distinct right Cauchy–Green tensors corresponding to the same principal stretches. However, this is problematic since we do not gain any new information when using two right Cauchy–Green tensors which point to the same invariants as inputs to our stress response experiment. Two ways to overcome this issue are to:

- Sample in the space of the principal strains.
- Sample directly in invariant space.

However, when naively sampling with either one of these ideas we crucially lose any information about the training domain that we have defined in the space of the deformation gradient. To illustrate this problem, Fig. 3 shows the spread of 20,000 points in principal invariant space sampled from differently bounded regions of the nine-dimensional deformation gradient space of Eq. (45). Additionally, it is obvious from Fig. 3 that the points in principal space follow a specific pattern, i.e. some combinations of $I_1 - I_2 - I_3$ values are not obtainable because they do not correspond to a physical deformation gradient. Therefore, we aim to develop an approach that evenly spreads points in the invariant space corresponding to a certain deformation gradient training domain following Eq. (45), while maintaining that every generated sample corresponds to a physically attainable deformation.

In this paper we are the first to present a sampling strategy that allows generating evenly distributed physical samples in a pre-determined region of the invariant space. Consequently, any trained surrogate model will be more proficient by requiring less data. The proposed approach is based on simulated annealing [47]. Furthermore, the presented approach generates evenly spread samples in isotropic invariant space. Keeping these samples fixed we present a second level of sampling in the space of pseudo-invariants. This way it is possible to efficiently build upon the sample points for isotropic materials when needing to extend to anisotropic materials, and the framework allows efficient storing of sampled points in databases. In the following we will first restrict ourselves to generating samples for the principal invariants $I_1 - I_2 - I_3$.

The first requirement for the sampling approach is that a bounded region in the principal invariant space should be known based on the required deformation gradient input domain. As there are different ways to achieve this, in this work we propose to generate sufficiently many samples in the deformation gradient space, i.e. here 100,000, such that it can be assumed that the resulting points in invariant space adequately span the bounded domain. Our

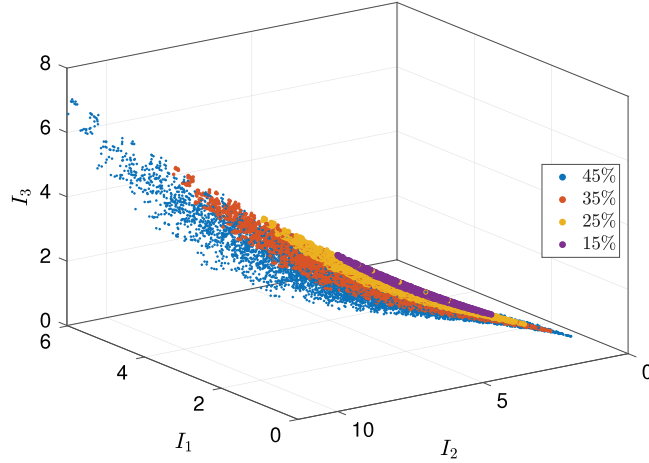


Fig. 3. Spread samples in principal invariant space when mapped from 20,000 samples generated in deformation gradient space with different δ bounds according to Eq. (45).

approximation is that the convex hull of all of these points is equivalent to the convex hull that we aim to sample in. To generate the convex hull from a set of points in three-dimensions we refer to Chazelle [48]. The steps undertaken to generate the convex hull are summarized in Algorithm 1. We could have alternatively chosen to the alpha-shape method [49] to approximate the hull of the sampled points. However, this framework depends on user-chosen parameter values which are not necessarily trivial to choose. Hence, for the purpose of this work we found the convex hull approximation to be more robust.

Algorithm 1: Obtain convex hull of permissible invariant space points based on deformation gradient bounds.

Result: $convI$: Convex hull of permissible points in invariant space

Input: Deformation gradient component bounds F_{ij}^L and F_{ij}^U , Number of random samples n

Sample F n -times

for $i = 1 : n$ **do**

 Obtain $J = \det F_i$

if $J > 0$ **then**

$C_i = F_i^T F_i$

 Obtain I_1, I_2, I_3 from C_i

 Store invariants in $II[i, :] = [I_1, I_2, I_3]$;

end

end

return $convI = conv(II)$

A convex hull built this way is shown in Fig. 2. From this visualization it can be noted that even though the convex hull is an enclosing envelope of all the sample points, it also encloses a part of the domain that does not correspond to physical deformations, e.g. the upper part inside the convex hull that has no samples in it. Hence, even when generating new points that lie inside the convex envelope we still need to ensure that these points are physical. A simple but often overlooked check for the physicality of an invariant set in isotropic materials is derived from the fact that all principal strains ($\lambda_1, \lambda_2, \lambda_3$) defined as the square root of the positive eigenvalues of the right Cauchy–Green tensor have to be real-valued. According to Currie [50] and Burnside and Panton [51] and under consideration that three principal invariants (I_1, I_2, I_3) are given we can define

$$\begin{aligned} H &= \frac{1}{9}(I_1^2 - 3I_2) \\ G &= \frac{1}{3}I_1I_2 - I_3 - \frac{2}{27}I_1^3 \\ \beta &= \arccos\left(-\frac{G}{2H^{\frac{3}{2}}}\right) \end{aligned} \quad (46)$$

Following the theorem from [51][p.84, “Criterion of the Nature of the Roots of a Cubic.”] we can say that when the two conditions

1. $G^2 + 4H^3 \leq 0$
 2. $\beta \in \mathbb{R}$
- (47)

are fulfilled, the triplet of invariants (I_1, I_2, I_3) correspond to a physical deformation. When these conditions are met, the squares of the principal strains can be reconstructed as

$$\begin{aligned}\lambda_{1,rec}^2 &= \frac{1}{3}I_1 - 2\sqrt{H} \cos\left(\frac{\pi - \beta}{3}\right) \\ \lambda_{2,rec}^2 &= \frac{1}{3}I_1 - 2\sqrt{H} \cos\left(\frac{\pi + \beta}{3}\right) \\ \lambda_{3,rec}^2 &= \frac{1}{3}I_1 + 2\sqrt{H} \cos\left(\frac{\beta}{3}\right).\end{aligned}$$
(48)

This, crucially (for the determination of the resulting stresses) allows us to reconstruct a right Cauchy–Green tensor that corresponds to (I_1, I_2, I_3)

$$\mathbf{C}_{rec} = \begin{bmatrix} \lambda_{1,rec}^2 & 0 & 0 \\ 0 & \lambda_{2,rec}^2 & 0 \\ 0 & 0 & \lambda_{3,rec}^2 \end{bmatrix}.$$
(49)

Now that we have defined an approach to evaluate the physicality of the points in the principal invariant space and are also able to ensure that the points are inside the input domain in deformation gradient space, we can outline the simulated annealing algorithm.

Let us assume that our goal is to generate N evenly sampled points in the invariant space. The algorithm starts with $N - 1$ randomly sampled points that are grouped in an array $\mathbf{II} \in \mathbb{R}^{(N-1) \times 3}$ inside the bounded invariant space, which could for example be coming from $N - 1$ sampled points in the deformation gradient space. When those initial $N - 1$ points are obtained, extend \mathbf{II} with one more sample which has the invariant values of the undeformed configuration ($I_1 = 3, I_2 = 3, I_3 = 1$). This last sample is kept unchanged throughout the simulated annealing process in order to ensure the preservation of the stress-free undeformed configuration.

The basic idea of the algorithm is then to iteratively adjust the positions of the generated samples over N_T annealing steps based on three checks:

1. The new position of the current sample increases the distance to the nearest neighbor of the remaining dataset in comparison to its previous position.
2. The new sample position of the current point is inside the convex hull of the predefined bounded deformation gradient domain.
3. The new sample position of the current sample corresponds to a physical point in principal invariant space.

To do this in each annealing step we loop over every point of the current sample set \mathbf{II} . For the current sample \mathbf{II}_j , we find the distance d to the closest data point in the remaining sample set $\{\mathbf{II} \setminus \mathbf{II}_j\}$. After that, we utilize the Box–Muller transform [52] in order to generate a random three dimensional step direction value \mathbf{n} , which is located on the surface of a three dimensional unit sphere. The Box–Muller transform is defined by the following steps

$$\begin{aligned}\text{Step 1 : } & u, v, w \in \mathcal{N}(0, 1) \\ \text{Step 2 : } & s = \sqrt{u^2 + v^2 + w^2} \\ \text{Step 3 : } & \mathbf{n} = \frac{1}{s}(u, v, w).\end{aligned}$$
(50)

Next, we sample a step magnitude $s \in [0, T]$ where T is the current step size. Now that we know the step size and step direction we can obtain the next possible position of the sample \mathbf{II}_j with $\mathbf{p} = \mathbf{II}_j + s\mathbf{n}$. This point position is accepted ($\mathbf{II}_j = \mathbf{p}$) when (i) its distance to the nearest neighbor in the remaining dataset is larger than d , when (ii) it passes the physicality test of Eq. (47), and (iii) it is inside the convex hull of the pre-sampled points of algorithm 1. If one of these conditions is not met, \mathbf{II}_j remains unchanged. This process is repeated N_T times

for every single point in the dataset. After each iteration the step size T is reduced by a constant factor $T = \alpha T$ where $0 < \alpha < 1$. The full algorithm is summarized in Algorithm box 2.

Algorithm 2: Algorithm to obtain evenly spread points in invariant space based on physical constraints and deformation gradient bounds using a simulated annealing approach. Values chosen by the authors are provided.

Result: Π : Matrix of evenly spread points in invariant space

Input: Convex hull of permissible points $convI$, Target number of points in invariant space N , Number of annealing steps $N_T = 7,000$, Step size $T = 1$, Step size factor $\alpha = 0.9995$

Sample $N - 1$ points randomly in $convI$ generating Π

Add vector of unstressed configuration $I_1 = 3, I_2 = 3, I_3 = 1$ to Π

for $i = 1 : N_T$ **do**

for $j = 1 : N$ **do**

 Set $\Pi^{-j} = \{\Pi \setminus \Pi_j\}$

 Set d as distance to closest neighbor of Π_j in Π^{-j}

 Randomly sample 3-dimensional unit sphere point \mathbf{n} (eq. (50))

 Set $s \in \mathcal{U}[0, T]$

$\mathbf{p} = \Pi_j + s\mathbf{n}$

 Set d_{test} as distance to closest neighbor of \mathbf{p} in Π^{-j}

$t = 0, t_u = 1$

if $\Pi_j == [3, 3, 1]$ **then**

$t_u = 0$

end

if \mathbf{p} in $convI$ && $d_{test} > d$ && \mathbf{p} is physical (see eq. (47)) **then**

$t = 1$

end

$\Pi_j = \Pi_j + s t t_u \mathbf{n}$

end

$T = \alpha T$

end

return Π

The effectiveness of the proposed approach is highlighted by an example. Consider a 17.5% training domain in which we aim to generate 200 samples. We use an initial step size of $T = 1$. The relevant hyperparameters that influence the outcome of the optimization are the number of time steps and the step size factor. If the change of the step size from one time step to another is small enough and the number of time steps is sufficiently high the presented approach should always result in proficently even sampling positions. We found that for the presented case 7500 time steps and a step size factor of $\alpha = 0.995$ fulfills these requirements. Alternatively, the process can be stopped after the particles no longer change their positions, i.e. the step size is small enough. The sample positions over the process after a different number of steps have been completed are shown in Fig. 4(c) along with the convex hull of the bounded domain in gray. It can be seen that the initially generated samples are not spread evenly in the intended region. However after 7500 steps a space-filling sample distribution can be observed.

4.1. Extension to anisotropic materials

In the previous section we have discussed how to generate evenly spaced physical samples inside a bounded domain. However, the presented algorithm is only applied to the principal invariants (I_1, I_2, I_3) and is not applicable to finding equivalent space-filling components of the pseudo invariant space that includes pseudo-invariants (e.g. in the transversally isotropic case I_4, I_5). This is due to the fact that there is, to the best of the authors knowledge, no direct analytical check for the physicality/reachability of the quintuple (I_1, I_2, I_3, I_4, I_5), i.e. if a specific combination of the five invariants is realizable with real eigenvalues of the Cauchy–Green tensor where (I_1, I_2, I_3) are inside a bounded domain. In Fig. 5 we sampled 10,000 samples in deformation gradient space for different training domains

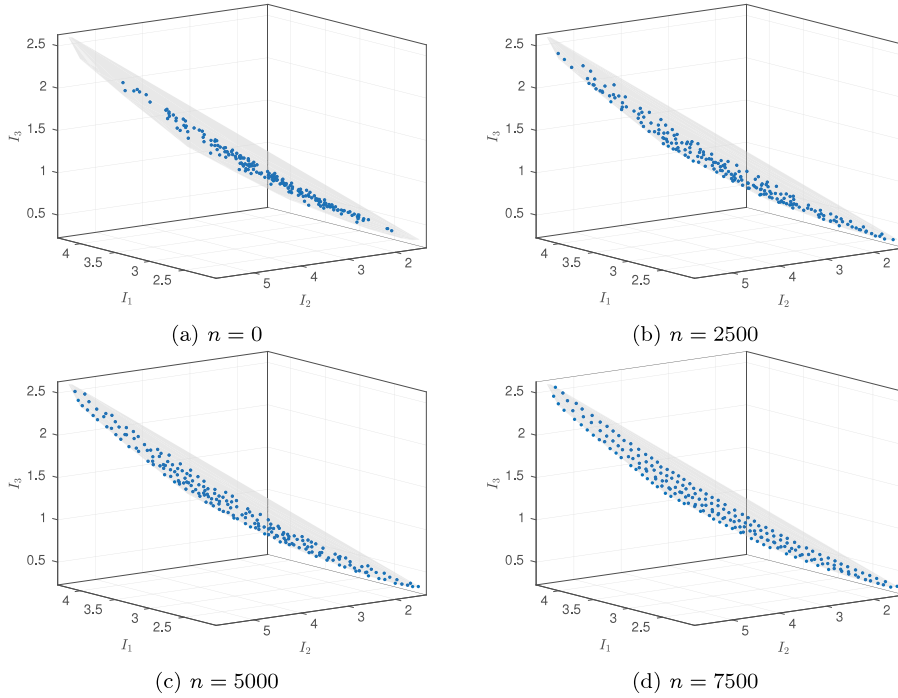


Fig. 4. Selected steps of the proposed minimax (space-filling) distance sampling approach with 200 sample points in invariant space ($I_1 - I_2 - I_3$). Convex hull of permissible sample point positions of 17.5% input range in deformation gradient space in gray. Step size $T = 1$, step size factor $\alpha = 0.9995$.

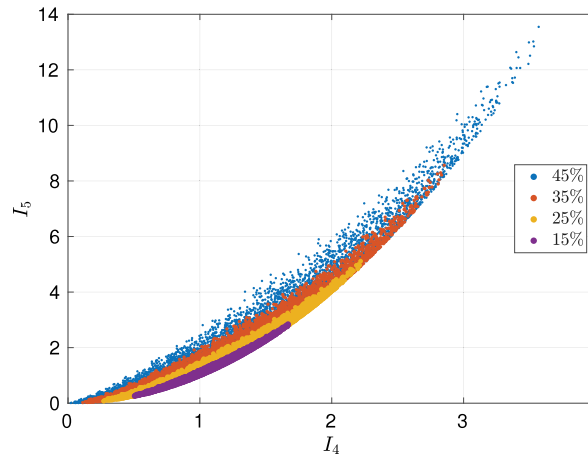


Fig. 5. Spread of pseudo-invariants with increasing training domain and $\mathbf{a}_0 = \frac{1}{\sqrt{6}}[1, 2, 1]^T$. 10,000 samples are plotted in deformation gradient space with LHS in each bounded domain. Last two components of the quintuple (I_1, I_2, I_3, I_4, I_5) are plotted.

(corresponding to increasing values of $\delta \in [0.15, 0.45]$ from Eq. (45)) and plotted the transversally isotropic pseudo-invariants (assuming a known referential vector $\mathbf{a}_0 = \frac{1}{\sqrt{6}}[1, 2, 1]^T$) for the generated samples. We can see that all of the pseudo-invariant values follow a clear pattern (not all points are reachable, i.e. the values $I_4 = I_5 = 2$) and furthermore that the convex-hull of the points would not be a good enough indicator as to where pseudo-invariant points can reside. Hence we need to come up with an additional sampling algorithm for the pseudo-invariant space, which does not change the positions of the primary invariants. That the samples in the pseudo-invariant space must also be evenly distributed can be seen from Figs. 6(a) and 6(b). In Fig. 6 we again sampled 10,000 points from the

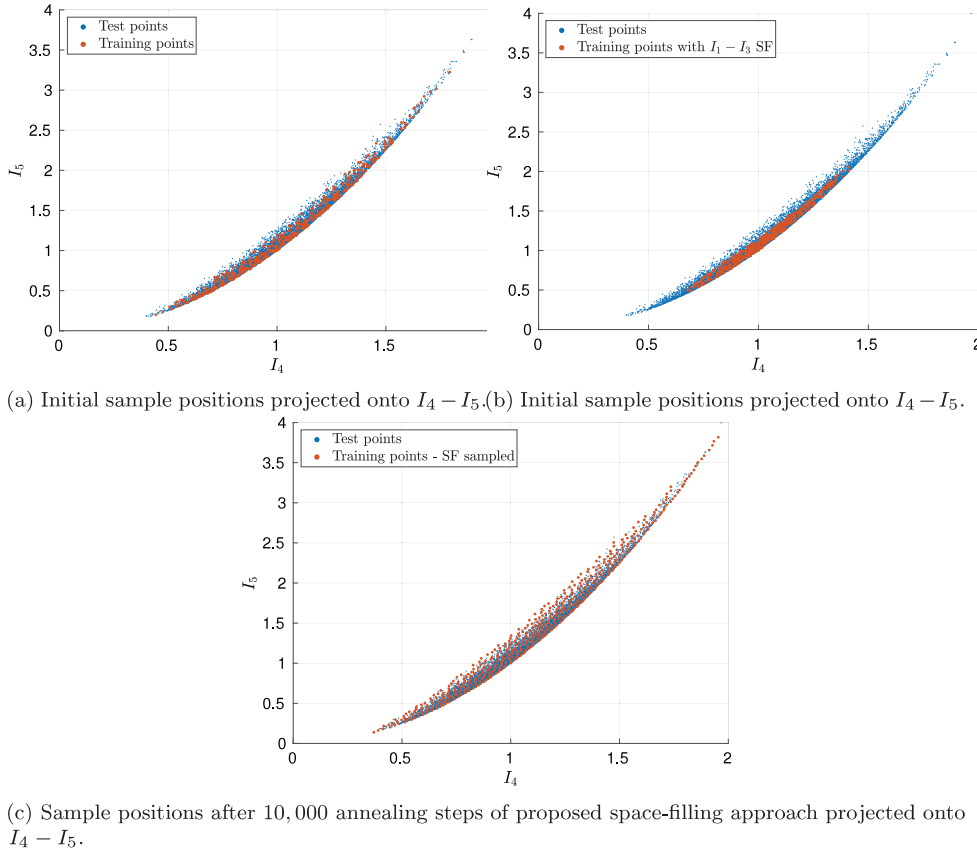


Fig. 6. Initial positions and positions after minimax approach with 10,000 iteration steps of 500 samples projected onto $I_4 - I_5$ plane in red ($\mathbf{a}_0 = \frac{1}{\sqrt{6}}[1, 2, 1]^T$). Positions of 10,000 test points generated in 17.5% training range in deformation gradient space with LHS and then projected onto $I_4 - I_5$ plane in blue. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

deformation gradient space with LHS in a 17.5% training domain (blue dots) as well as an additional 500 separate samples (red) dots. It can be seen that especially the outer edges of the envelope surrounding the blue dots are not well preserved by the 500 samples.

Next, we used algorithm 2 to sample 500 evenly spaced points in the primary invariant space. For each of these triplet of (I_1, I_2, I_3) we used Eqs. (48) and (49) to obtain corresponding right Cauchy–Green tensor. Since we know the referential vector \mathbf{a}_0 we are able to obtain the pseudo-invariant values (I_4, I_5) for all of the 500 samples which are plotted in Fig. 6(b). It can be seen that even though we have an excellent spread in the primary invariant space the obtained samples in pseudo-invariant space are very clustered and are not at all space-filling.

Hence, we propose an additional algorithm based on simulated annealing that is able to generate evenly spaced points in the pseudo-invariant space. The final points generated by this algorithm (using the same settings as described for Figs. 6(a) and 6(b)) are shown in Fig. 6(c) which preemptively highlights the performances of the following procedure.

The idea behind this approach, in the context of anisotropic material response, is based upon the fact that simple rotations of the right Cauchy Green tensor result in different pseudo-invariant values while leaving the principal invariants unchanged. To emphasize this we look at the following example. Consider a right Cauchy–Green tensor to be given by

$$\mathbf{C} = \begin{bmatrix} 1.4 & 0 & 0 \\ 0 & 1.1 & 0 \\ 0 & 0 & 0.8 \end{bmatrix} \quad (51)$$

which using $\mathbf{a}_0 = \frac{1}{\sqrt{6}}[1, 2, 1]^T$ has the invariant quintuple (3.3, 3.54, 1.232, 1.1, 1.24). Next consider this tensor to be rotated in $y - z$ plane around an angle of 0.1 rad, i.e.

$$\mathbf{C}_{rot} = \mathbf{R}_{\alpha=0.1}^T \mathbf{C} \mathbf{R}_{\alpha=0.1} \approx \begin{bmatrix} 1.4 & 0 & 0 \\ 0 & 1.097 & -0.029 \\ 0 & -0.029 & 0.803 \end{bmatrix} \quad (52)$$

which has the approximate invariant quintuple (3.3, 3.54, 1.232, 1.078, 1.199). Hence, this observation allows us to generate different pseudo-invariant values from known (and fixed) right Cauchy–Green tensors. We take advantage of this and build a simulated annealing algorithm around the three angles defining the rotation matrices acting on the three planes $x - y$, $x - z$ and $y - z$ in euclidean space. This has the advantage that the generated quintuple of values $(I_1, I_2, I_3, I_4, I_5)$ is always reachable from the primary invariants and the already obtained evenly spaced primary invariants do not need to be changed. This algorithm starts with the knowledge about N evenly spaced principal invariants $\mathbf{II}_{Iso} \in \mathbb{R}^{N \times 3}$ in some confined space with the requirement that one of the components of \mathbf{II} has to correspond to the invariants of the undeformed configuration. Then, using Eqs. (48) and (49) we obtain the right Cauchy–Green tensor \mathbf{C}_j , $j = 1, \dots, N$ in principal space for each of the N samples and the corresponding pseudo-invariants which are stored in $\mathbf{II}_{ani} \in \mathbb{R}^{N \times 2}$. For simplicity we consider the three rotation matrices

$$\begin{aligned} \mathbf{R}_x &= \begin{bmatrix} \cos \alpha_x & -\sin \alpha_x & 0 \\ \sin \alpha_x & \cos \alpha_x & 0 \\ 0 & 0 & 1 \end{bmatrix}, & \mathbf{R}_y &= \begin{bmatrix} \cos \alpha_y & 0 & \sin \alpha_y \\ 0 & 1 & 0 \\ -\sin \alpha_y & 0 & \cos \alpha_y \end{bmatrix}, \\ \mathbf{R}_z &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha_z & -\sin \alpha_z \\ 0 & \sin \alpha_z & \cos \alpha_z \end{bmatrix}. \end{aligned} \quad (53)$$

Each component of \mathbf{II}_{ani} gets assigned to one instance of the three rotation defining angles which are initially individually sampled in $\mathcal{U}[0, 2\pi]$ and stored in the angle matrix $\mathcal{A} \in \mathbb{R}^{N \times 3}$.

Then, similarly to Algorithm 2 the process iterates over N_T loops where each loop consists of the following procedure. Loop over all elements of \mathbf{II}_{ani} . For each component $\mathbf{II}_{j,ani} \subset \mathbf{II}_{ani}$ find the distance d to the current closest neighbor in the remaining dataset. Then using Box–Muller transform of Eq. (50) sample a random point on the three dimensional unit sphere which acts as the step direction \mathbf{n} of the current sample. Then, knowing the current step size T , sample $s = \mathcal{U}[0, T]$. The next possible position of the angle \mathcal{A}_j that is assigned to the current point can then be obtained with $\mathbf{p} = \mathcal{A}_j + s\mathbf{n}$. In order to check if this set of angles leads to a pseudo-invariant combination that is further away from the closest point than d , we obtain the rotated right Cauchy–Green tensor of the current point with

$$\mathbf{C}_{test} = \mathbf{R}_x^T(\mathbf{p}[1])\mathbf{R}_y^T(\mathbf{p}[2])\mathbf{R}_z^T(\mathbf{p}[3])\mathbf{C}_j\mathbf{R}_z(\mathbf{p}[3])\mathbf{R}_y(\mathbf{p}[2])\mathbf{R}_x(\mathbf{p}[1]). \quad (54)$$

From this tensor we can again obtain $I_{4,test}$ and $I_{5,test}$ and use these values to check the distance to the closest neighbor in the remaining dataset $\{\mathbf{II}_{ani} \setminus \mathbf{II}_{j,ani}\}$. If this distance is larger than d , set $\mathcal{A}_j = \mathcal{A}_j + s\mathbf{n}$ and $\mathbf{II}_{j,ani} = [I_{4,test}, I_{5,test}]$. If this is not the case, then leave all values unchanged. After each loop over all points of \mathbf{II}_{ani} reduce the step size with $T = \alpha T$ where $0 < \alpha < 1$. The full approach is summarized in algorithm box 3.

In order to obtain stress responses from some sample $(I_1, I_2, I_3, I_4, I_5)$, a corresponding right Cauchy–Green tensor needs to be available. As previously discussed, no analytical version analogue to the one for isotropic materials (Eqs. (48) and (49)) has been presented in the literature. For this reason we propose an approach based on numerical optimization. The right Cauchy–Green tensor is a solution of the constrained nonlinear equation system

$$\text{find } \mathbf{C}^* \text{ such that } \begin{bmatrix} \text{tr}(\mathbf{C}^*) - I_1 \\ 0.5(\text{tr}(\mathbf{C}^*)^2 - \text{tr}((\mathbf{C}^*)^2)) - I_2 \\ \det(\mathbf{C}^*) - I_3 \\ \text{tr}(\mathbf{A}\mathbf{C}^*) - I_4 \\ \text{tr}(\mathbf{A}(\mathbf{C}^*)^2) - I_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (55)$$

where $\mathbf{C}_{ij}^{lb} \leq \mathbf{C}_{ij}^* \leq \mathbf{C}_{ij}^{ub}$

Algorithm 3: Obtain evenly spread samples when pseudo-invariants are present. Values chosen by the authors are provided.

Result: $\Pi \in \mathbb{R}^{N \times 5}$: Matrix of evenly spread points in isotropic-invariant (I_1, I_2, I_3) and pseudo-invariant (I_4, I_5) space

Input: Target number of points in invariant space N , Number of annealing steps $N_T = 10,000$, Step size $T = 2\pi$, Step size factor $\alpha = 0.9995$, Evenly sampled points in isotropic invariant space $\Pi_{iso} \in \mathbb{R}^{N \times 3}$ based on Algorithm 2

for $j = 1 : N$ **do**

 Obtain C_j from $\Pi_{j,iso}$ using equations (48) and (49)

 Set $\Pi_{j,ani} = [I_4, I_5]$ obtained from C_j

end

Initialize angle matrix $\mathcal{A} \in \mathbb{R}^{N \times 3} \in \mathcal{U}[0, 2\pi]$

Set $R_x(\alpha_x) = \begin{bmatrix} \cos \alpha_x & -\sin \alpha_x & 0 \\ \sin \alpha_x & \cos \alpha_x & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $R_y(\alpha_y) = \begin{bmatrix} \cos \alpha_y & 0 & \sin \alpha_y \\ 0 & 1 & 0 \\ -\sin \alpha_y & 0 & \cos \alpha_y \end{bmatrix}$,

$R_z(\alpha_z) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha_z & -\sin \alpha_z \\ 0 & \sin \alpha_z & \cos \alpha_z \end{bmatrix}$

for $i = 1 : N_T$ **do**

for $j = 1 : N$ **do**

 Set $\Pi^{-j}_{ani} = \{\Pi_{ani} \setminus \Pi_{j,ani}\}$

 Set d as distance to closest neighbor of $\Pi_{j,ani}$ in Π^{-j}_{ani}

 Randomly sample 3-dimensional unit sphere point \mathbf{n} (eq. (50))

 Set $s \in \mathcal{U}[0, T]$

$\mathbf{p} = \mathcal{A}_j + s\mathbf{n}$

 Set $C_{test} = R_x^T(\mathbf{p}[1])R_y^T(\mathbf{p}[2])R_z^T(\mathbf{p}[3])C_jR_z(\mathbf{p}[3])R_y(\mathbf{p}[2])R_x(\mathbf{p}[1])$

 Set $\mathbf{p}_{ani} = [I_4, I_5]$ obtained from C_{test}

 Set d_{test} as distance to closest neighbor of \mathbf{p}_{ani} in Π^{-j}_{ani}

if $d_{test} > d$ **then**

$\mathcal{A}_j = \mathcal{A}_j + s\mathbf{n}$

$\Pi_{j,ani} = \mathbf{p}_{ani}$

end

end

$T = \alpha T$

end

return $\Pi = [\Pi_{iso}, \Pi_{ani}]$

where the upper and lower bounds C^{ub} and C^{lb} are known from the training domain of the deformation gradient. A real solution can be obtained with any nonlinear solver (such as the fsolve function of scipy [53]) when the input quintuple $(I_1, I_2, I_3, I_4, I_5)$ is a reachable value which is guaranteed when obtaining the samples from Algorithm 3.

Overall, the presented sampling approach is generally applicable and is not specific to certain constitutive responses. Hence, the introduced method can generate samples for a multitude of different problems.

5. Gaussian process regression

Gaussian process regression has recently gained more popularity for building surrogate models for constitutive laws. This is due to their convergence guarantees, deep stochastic background and excellent performance for out-of-sample model predictions [26]. They have also evolved into a common choice for building active learning models [54].

Consider a general dataset consisting of N data points to be given by

$$\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^N \quad (56)$$

where $\mathbf{x} \in \mathbb{R}^{n_i}$ and $\mathbf{y} \in \mathbb{R}^{n_o}$. The output vectors can be recombined to build the output vector $\mathbf{y}^{tp} \in \mathbb{R}^{n_o N}$ such that

$$\mathbf{y}^{tp} = [\mathbf{y}_1 \quad \dots \quad \mathbf{y}_N]^T. \quad (57)$$

Assume that the input output relationship can be approximated by a realization of a Gaussian process given by

$$\mathbf{Y}(\mathbf{x}) = \boldsymbol{\mu} + \mathbf{A}\mathbf{Z}, \quad (58)$$

with the output $\mathbf{Y} \in \mathbb{R}^{n_o}$, the mean $\boldsymbol{\mu} \in \mathbb{R}^{n_o}$, a positive-definite matrix $\mathbf{A} \in \mathbb{R}^{n_o \times n_o}$ (the first set of unknown parameters), and a vector of mutually independent Gaussian processes $\mathbf{Z} \in \mathbb{R}^{n_o}$ [55].

The correlation decay between two inputs \mathbf{x} and \mathbf{x}' is typically modeled by a user-defined autocorrelation function. Laurent et al. [56] identified the class of Matérn kernels as the most proficient autocorrelation formulation for computer experiments when no prior knowledge is available. In this, we restrict ourselves to the so-called Matérn 3/2 function [57] which reads

$$R(\mathbf{x}, \mathbf{x}', \boldsymbol{\theta}_i) = \prod_{k=1}^{n_i} \left(1 + \frac{\sqrt{3}|x_k - x'_k|}{\theta_{i,k}} \right) \exp \left(-\frac{\sqrt{3}|x_k - x'_k|}{\theta_{i,k}} \right), \quad (59)$$

where $\boldsymbol{\theta} = [\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_{n_o}]^T$ is a vector of unknown and trainable parameters

Consider the correlation matrix $\mathbf{R} \in \mathbb{R}^{n_o \times n_o}$ to be given by

$$\mathbf{R}(\mathbf{x}, \mathbf{x}') = \text{diag}\{R(\mathbf{x}, \mathbf{x}', \boldsymbol{\theta}_1), \dots, R(\mathbf{x}, \mathbf{x}', \boldsymbol{\theta}_{n_o})\}. \quad (60)$$

which allows us to write the covariance between two input values as

$$\text{Cov}(\mathbf{Y}(\mathbf{x}), \mathbf{Y}(\mathbf{x}')) = \mathbf{A}\mathbf{R}(\mathbf{x}, \mathbf{x}')\mathbf{A}^T \quad (61)$$

and the block-component-wise entries of the covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{n_o N \times n_o N}$ by

$$[\boldsymbol{\Sigma}]_{ij} = \text{Cov}(\mathbf{Y}(\mathbf{x}_i), \mathbf{Y}(\mathbf{x}_j)). \quad (62)$$

Then, a prediction with GPR at the input point \mathbf{x}_\star can be made with

$$\hat{\mathbf{y}}(\mathbf{x}_\star) = \hat{\boldsymbol{\mu}} + \boldsymbol{\Pi}(\mathbf{x}_\star)\boldsymbol{\Sigma}(\mathbf{y}^{tp} - \mathcal{F}\hat{\boldsymbol{\mu}}), \quad (63)$$

with $\mathcal{F} \in \mathbb{R}^{n_o N \times n_o} = \mathbf{1}_N \otimes \mathbf{I}_{n_o}$ where $\mathbf{1}_N \in \mathbb{R}^N$ is a vector of ones and $\mathbf{I}_{n_o} \in \mathbb{R}^{n_o \times n_o}$ is a unit matrix. Furthermore, $\boldsymbol{\Pi} \in \mathbb{R}^{n_o \times n_o N}$ and $\hat{\boldsymbol{\mu}} \in \mathbb{R}^{n_o}$ are defined as:

$$\boldsymbol{\Pi}(\mathbf{x}_\star) = [\text{Cov}(\mathbf{Y}(\mathbf{x}_\star), \mathbf{Y}(\mathbf{c}_1)) \quad \dots \quad \text{Cov}(\mathbf{Y}(\mathbf{x}_\star), \mathbf{Y}(\mathbf{c}_N))], \quad (64a)$$

$$\hat{\boldsymbol{\mu}} = (\mathcal{F}^T \boldsymbol{\Sigma}^{-1} \mathcal{F})^{-1} \mathcal{F}^T \boldsymbol{\Sigma}^{-1} \mathbf{y}^{tp}. \quad (64b)$$

The prediction output of Eq. (63) is dependent on the values of the unknown trainable parameters \mathbf{A} and $\boldsymbol{\theta}$. To simplify this procedure we assume in the following that the outputs are uncorrelated which means \mathbf{A} is a priori defined as a unit matrix. The remaining parameters $\boldsymbol{\theta}$ can be found using a restricted maximum likelihood approach [55]

$$\hat{\boldsymbol{\theta}} = \arg \max_{\boldsymbol{\theta}^\star} \left[-\frac{1}{4} \log(|\boldsymbol{\Sigma}|) \log(\mathcal{F}^T \boldsymbol{\Sigma}^{-1} \mathcal{F}) + \right. \\ \left. -\frac{1}{2} (\mathbf{y}^{tp} - \mathcal{F}\hat{\boldsymbol{\mu}})^T \boldsymbol{\Sigma}^{-1} (\mathbf{y}^{tp} - \mathcal{F}\hat{\boldsymbol{\mu}}) \right]. \quad (65)$$

After finding the best parameters, the GPR regression model is fully defined and predictions can be obtained by using Eq. (63).

Local approximate Gaussian process regression

Due to its setup as a nonparametric model, GPR suffers from computational intractability in the big data domain (>1000 data points) [26]. Based on the works of Gramacy and Apley [58], Kleijnen and van Beers [59], the authors in their recent work [29] introduced local approximate GPR (laGPR) to the field of data-driven constitutive models. This technique keeps the major advantages of the general GPR method but makes it tractable for larger datasets. The basic concept is build around the premise that points closer to an input of interest \mathbf{x}_* have more influence to its output prediction than points far away from it. This can be understood when observing the properties of the autocorrelation function of Eq. (59), if the elementwise difference between two points is significant the exponential will result in $R(\mathbf{x}_*, \mathbf{x}', \bullet) = 0$. Other autocorrelation functions show similar effects, see e.g. [60]. The idea in laGPR is to find a subset of cardinality $n \ll N$ of the whole dataset with N training points with which a locally accurate surrogate model can be obtained. These input points are called inducing points. Different techniques to find the inducing point set have been proposed and investigated in the literature. Different variations of formulations for the inducing point set \mathcal{X}_n have been explored and tested in the literature, see [61].

Specifically, Kleijnen and van Beers [59] simply use the n nearest neighbors of the point \mathbf{x}_* in the whole dataset as measured by the euclidean distance

$$d(\mathbf{x}_*, \mathbf{x}) = \sqrt{(\mathbf{x}_* - \mathbf{x})^T (\mathbf{x}_* - \mathbf{x})}. \quad (66)$$

Recently, it was shown by the authors that this formulation is able to accurately predict complex constitutive relationships based on data [29]. One negative side effect is that each output prediction requires a retraining of the unknown parameters of the local GPR model. However, as pointed out in [59], even with around 100 points in the local dataset the prediction can basically be done in real-time when choosing an efficient optimization algorithm.

6. Numerical tests

This section compares the presented physics-informed surrogate modeling approach for isotropic and anisotropic materials with the classical mapping approach. Furthermore, the efficiency of the proposed space-filling sampling approach is highlighted. For all the following numerical tests consider the training domain to be 17.5% which means

$$\bar{F}_{ij} \in [F_{ij}^L, F_{ij}^U] \text{ where } \begin{cases} 0.825 \leq 1 \leq 1.175, & \text{when } i = j \\ -0.175 \leq 0 \leq 0.175, & \text{when } i \neq j. \end{cases} \quad (67)$$

For testing, we randomly sample $N_t = 20,000$ points in this nine-dimensional training domain using LHS. With reference to these points we define the mean stress output error as

$$\mathcal{E}_S = \sqrt{\frac{1}{6N_t} \sum_{i=1}^6 \sum_{j=1}^{N_t} |\hat{s}_i^j - s_i^j|^2} \quad (68)$$

where $\mathbf{s} = [S_{11}, S_{12}, S_{13}, S_{22}, S_{23}, S_{33}]^T$.

All results were obtained with MATLAB [62].¹ The utilized local approximate GPR model is a modified version of the code provided by Lophaven et al. [63]. In order to showcase the efficiency of the presented approaches we deliberately choose the isotropic and anisotropic stress outputs to span different orders of magnitude. As described in Section 4 evenly-spread points in the deformation gradient space might have the same exact invariants and would therefore not yield any new information for the training process. Even more problematic is that the covariance matrix of GPR and therefore of laGPR gets ill conditioned when two points in the training dataset are too close to each other [60]. Hence, when obtaining input points in the invariant space from the deformation gradient we remove duplicate points from the dataset. We define two points $\mathbf{x}^1 \in \mathbb{R}^d$ and $\mathbf{x}^2 \in \mathbb{R}^d$ as being a duplicate if

$$|\mathbf{x}_i^1 - \mathbf{x}_i^2| < 0.01, \quad \forall i = 1, \dots, d. \quad (69)$$

This results in the fact that the size of the dataset used to train the classical mapping approach is always smaller than or equal to the one for the physics-informed mappings of Eqs. (26) and (38).

¹ After acceptance of the paper the codes of this manuscript will be released under https://github.com/FuhgJan/invariant_DD_CM. Python versions of the code can be made available under reasonable request.

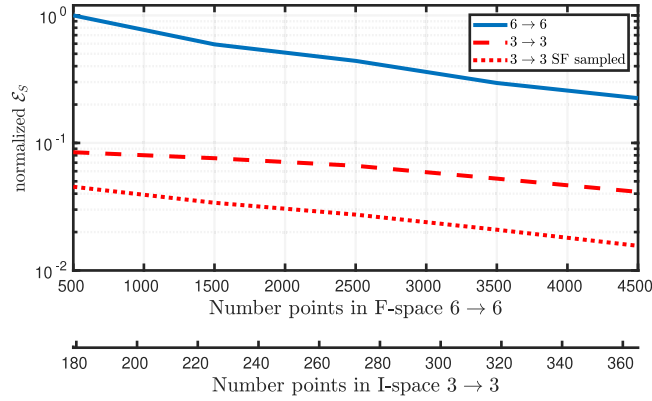


Fig. 7. Normalized error values for isotropic hyperelastic law using 20,000 test points in input range of 17.5% sampled with LHS. Three mapping approaches: the classical mapping approach (6 → 6), mapping in the invariant space (3 → 3) and mapping in the invariant space where the input points have been sampled with the proposed space-filling technique (3 → 3 SF sampled). Points have been sampled in deformation gradient space for 6 → 6 and then projected onto invariant space which reduced the number of points. Space-filling sampling has been done with the number of points in invariant space.

6.1. Isotropic example

Consider the compressible Mooney–Rivlin model of the form [31]

$$\Psi = \gamma_1(J - 1)^2 - 2(\gamma_2 + 2\gamma_3)\ln J + \gamma_2(I_1 - 3) + \gamma_3(I_2 - 3) \quad (70)$$

which yields the stress

$$\mathbf{S} = 2(\gamma_2 + \gamma_3 I_1)\mathbf{I} - 2\gamma_3 \mathbf{C} + (2\gamma_1 J(J - 1) - 2(\gamma_2 + 2\gamma_3))\mathbf{C}^{-1} \quad (71)$$

where we choose $\gamma_1 = \gamma_2 = 1.0$, and $\gamma_3 = 0.2$. In the following we use this analytical model to test the performance of the presented physics-informed surrogate modeling approach for the isotropic case with and without space-filling sampling and compare the results to the classical mapping approach as described in Eq. (23).

For all the following results the classical mapping was trained with 60 inducing points which is inline with the suggestion of Kleijnen and van Beers [59]. Since for all the investigated cases the number of points for the physical-mapping never exceeded 360 standard GPR was used for all of the following cases. Fig. 7 shows the error for the 20,000 randomly sampled points with LHS for the classical mapping approach (6 → 6), the physics-informed mapping approach (3 → 3) and the physics-informed mapping approach where the inputs are generated with the space-filling sampling technique of Algorithm 2 (3 → 3 SF sampled). For (6 → 6), between 500 and 4500 points are sampled in the nine-dimensional deformation gradient space of Eq. (67) with TPLHD. Using Eq. (69) the resulting invariant space datasets are then checked for duplicates, which results in datasets of sizes ranging between 180 and 360 samples. These samples are used to train the (3 → 3) approach. Using the same number of points as for (3 → 3) the Algorithm 2 is used to generate space-filling samples which were used to build the datasets for (3 → 3 SF sampled). From Fig. 7 it can be seen that even though the surrogate models in invariant space were trained on vastly lower number of points they perform better by around a factor 10 compared to the classical mapping approach. Additionally, the datasets that were created with the space-filling algorithm show large improvements in comparison to the plain (3 → 3) approach which uses samples that were generated in the deformation gradient space.

The differences between (3 → 3) and (3 → 3 SF sampled) are further highlighted in Fig. 8 which plots the normalized error values in invariant space for the 20,000 test points and 2500 deformation gradient training points (275 training points in invariant space) and which emphasizes the location of the largest errors. As expected, the areas of the largest errors of the (3 → 3) approach (Fig. 8(a)) are around the boundaries of the test data domain since the invariant is not sampled evenly. The errors of the space-filling dataset as shown in Fig. 8 are uniformly lower and especially the boundary regions are better represented. Here, the functional relationships that we intend

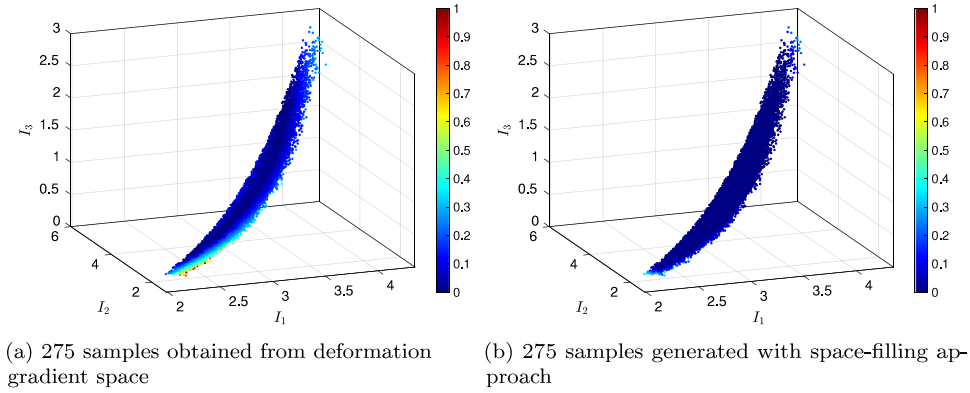


Fig. 8. Normalized error over 20,000 test points in 17.5% input range surrogate models trained with 275 sample points (a) 2500 points (sampled with TPLHD) in deformation gradient space and then projected onto invariant space which results in 275 samples when reducing duplicates (3 \rightarrow 3), (b) 278 sampled with the space-filling sampling approach of Algorithm 2 (3 \rightarrow 3 SF sampled).

to learn between the invariants and coefficients are rather simple. We expect the space-filling sampling scheme to provide an even more significant improvement in terms of error as these maps become more complex.

Consider the following applied deformation gradient range defining uniaxial tension/compression

$$\mathbf{F}_{app} = \mathbf{I} + F_{11,app} \mathbf{e}_1 \otimes \mathbf{E}_1, \quad \text{with } F_{11,app} \in [-0.8, 0.8] \quad (72)$$

not explicitly part of the training dataset, and which in fact applies loads far outside the input training domain of Eq. (67). The ground truth responses to this load in directions S_{11} and S_{22} as well as the predicted responses using 2500 TPLHD points for (6 \rightarrow 6) and 278 points for (3 \rightarrow 3) and (3 \rightarrow 3 SF sampled) after removing the duplicates are shown in Figs. 9(a) and 9(b). The corresponding absolute errors between the predicted and actual responses for these two cases are shown in Figs. 9(c) and 9(d) respectively. It can be seen that all of the investigated approaches are able to accurately capture the stress response within the training region (marked by the dotted lines in Figs. 9(a) and 9(b)). Notably, outside of the training domain the physics-informed approaches perform with significantly better accuracy compared to the classical mapping counterpart. Additionally, the space-filling approach shows better results than the normal (3 \rightarrow 3) mapping. This is the first example where physics-informed data-driven constitutive modeling in the context of hyperelasticity is shown to generalize proficiently. The initial training region corresponds to 17.5% but the results of Figs. 9(a) and 9(b) are shown in a testing region of 80%. Interestingly, it can be seen that the classical approach, which is solely data-driven provides non-physical results beyond the 17.5% training region.

6.2. Transversely isotropic example

Consider the following analytical transversely isotropic hyperelastic law suggested by Bonet and Burton [64] which is given as

$$\Psi = [\alpha + \beta \log J + \gamma(I_4 - 1)](I_4 - 1) - \frac{1}{2}\alpha(I_5 - 1), \quad (73)$$

yielding the corresponding second Piola–Kirchhoff stress tensor

$$\begin{aligned} \mathbf{S} = & 2\beta(I_4 - 1)\mathbf{C}^{-1} + 2[\alpha + 2\beta \log J + 2\gamma(I_4 - 1)]\mathbf{a}_0 \otimes \mathbf{a}_0 \\ & - \alpha(\mathbf{C}\mathbf{a}_0 \otimes \mathbf{a}_0 + \mathbf{a}_0 \otimes \mathbf{C}\mathbf{a}_0) \end{aligned} \quad (74)$$

where $\mathbf{a}_0 = \frac{1}{\sqrt{6}}[1, 2, 1]^T$ is a unit vector representing the direction of reinforcement, $I_4 = \mathbf{C} : \mathbf{a}_0 \otimes \mathbf{a}_0$ the fourth principal invariant, and normalized shear, bulk and fiber reinforcement moduli of $\alpha = 1.585e5$, $\beta = 5e4$ and $\gamma = 1.8e5$. This analytical model is used to test the performance of the presented physics-informed surrogate modeling approach for the anisotropic materials with and without space-filling sampling (here termed (5 \rightarrow 6) and (5 \rightarrow 6 SF sampled) respectively). These results are compared to the classical mapping approach (6 \rightarrow 6) as

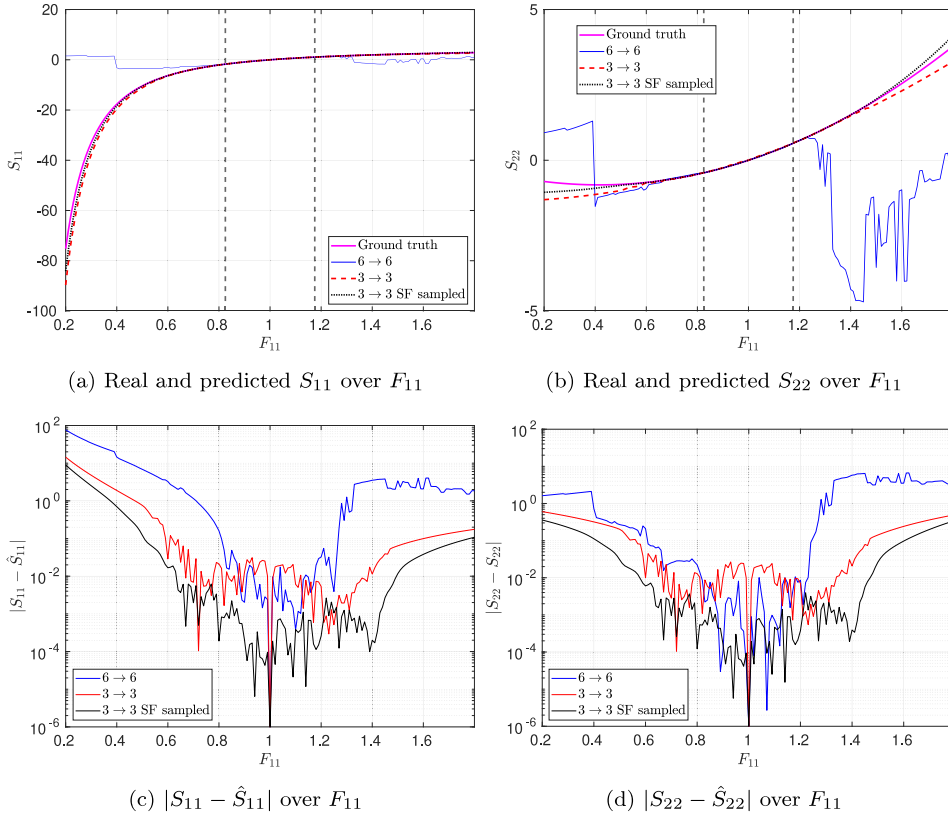


Fig. 9. Illustrative stress outputs for the isotropic case. Thicker horizontal dashed lines symbolize positions of 17.5% training domain.

described in Eq. (23). For all the following results all mappings were trained with laGPR and 60 inducing points. Fig. 10 shows the errors between the three cases for 20,000 test points in the training domain of Eq. (67) generated with LHS. Similarly to the isotropic case the mapping (6 \rightarrow 6) is trained on datasets consisting of 500 to 4500 points which are sampled in the nine-dimensional deformation gradient space of Eq. (67) with TPLHD. These datasets are then checked for duplicates using Eq. (69) which reduces the size of the datasets for training from the invariant space to a range between 500 and 3500 samples. These define datasets which are in turn used to train the (3 \rightarrow 3) approach. An equivalent number of samples is then respectively generated using the Algorithm 2 to obtain datasets consisting of space-filling samples in primary and pseudo-invariant space which are used to train (3 \rightarrow 3 SF sampled). It can be seen that training the surrogate in a physics-informed way decreases the error dramatically while at the same time less training samples are needed. Here again, the datasets that were generated in a space-filling manner outperform the datasets that were sampled in deformation gradient space.

Fig. 11 shows raw stress and error output values when the following deformation gradients yielding simple shear are applied

$$\mathbf{F}_{app} = \mathbf{I} + F_{12,app} \mathbf{e}_1 \otimes \mathbf{e}_2, \quad \text{with } F_{12,app} \in [-1, 1]. \quad (75)$$

This load path is not explicitly part of the training dataset, and even applies loads far beyond the input training domain of Eq. (67). Figs. 11(a) and 11(b) show the ground truth stress responses as well the predicted stresses in directions S_{11} and S_{12} . The respective absolute errors are shown in Figs. 11(c) and 11(b). The shown results are based on a (6 \rightarrow 6) model trained with 2000 training points and physics-informed models based on a corresponding 1480 points. Similarly to the isotropic case, it can be seen that all the surrogate models are able to capture the response inside the training domain (dotted vertical lines) in a proficient way. However outside the training domain the classical mapping approach becomes crucially unreliable while the physics-informed metamodels are able to follow the true response surprisingly far away from the main training domain once again confirming their ability to efficiently generalize. This indicates that training models that guarantee material frame indifference, material

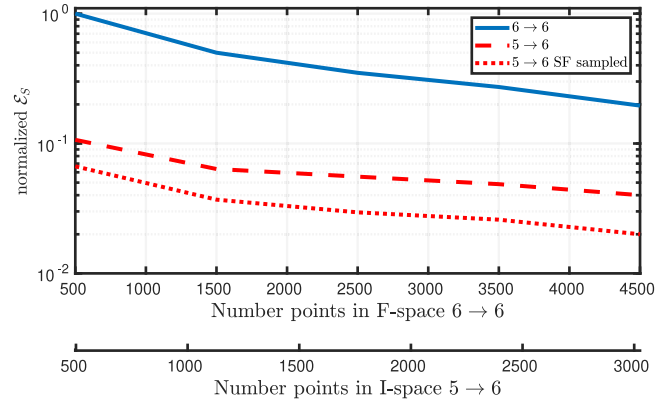


Fig. 10. Normalized error values for anisotropic hyperelastic law using 20,000 test points in input range of 17.5% sampled with LHS. Three mapping approaches: the classical mapping approach ($6 \rightarrow 6$), mapping in the invariant space ($5 \rightarrow 6$) and mapping in the invariant space where the input points have been sampled with the proposed space-filling technique ($5 \rightarrow 6$ SF sampled). Points have been sampled in deformation gradient space for $6 \rightarrow 6$ and then projected onto invariant space which reduced the number of points. Space-filling sampling has been done with the number of points in invariant space.

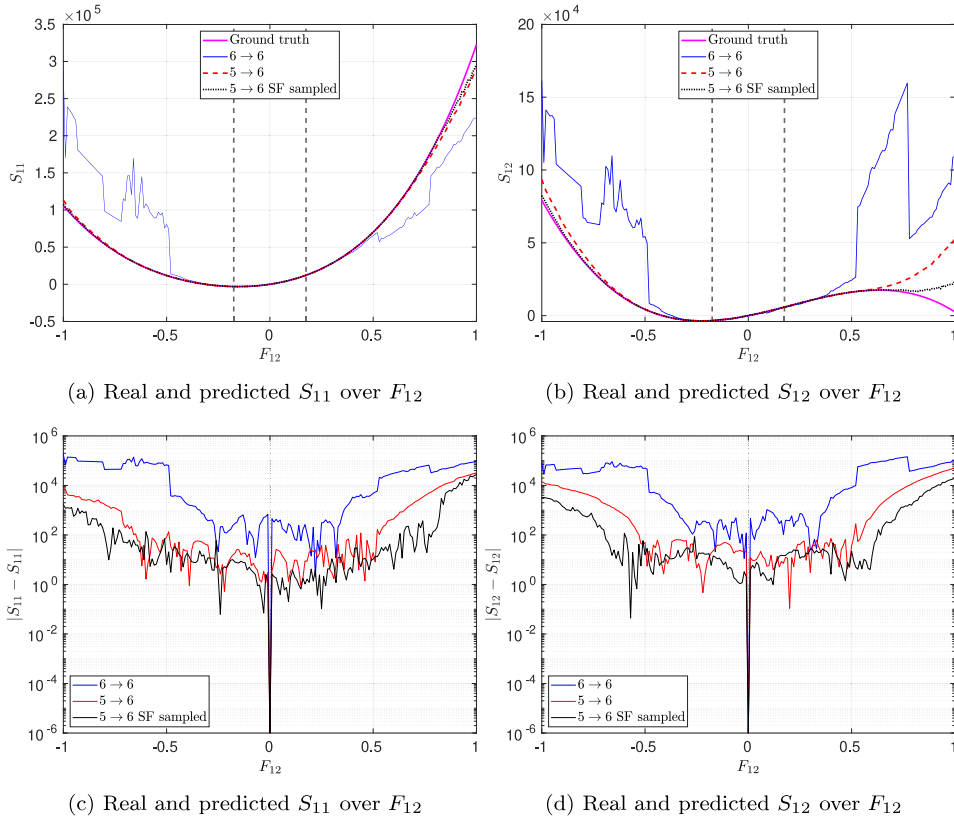


Fig. 11. Illustrative stress outputs for the anisotropic case. Thicker horizontal dashed lines symbolize positions of 17.5% training domain.

symmetry and thermodynamic consistency allows them to intrinsically learn the involved physics of the material law. Here again, the space-filling sampling approach shows better performances than the model that was trained from the dataset that was sampled in the deformation gradient space.

7. Discussion and outlook

This paper presents a technique to obtain physics-informed data-driven surrogates for hyperelastic material models based on training data from isotropic and anisotropic materials. The idea is based on writing the second Piola–Kirchhoff stress as a linear combination of an irreducible basis of stress generators. This allows us to build metamodels that map from the corresponding invariants of the input to the scalars of the linear combination. The trained models are then able to inherently capture five physical concepts: the preservation of the stress-free undeformed configuration, local balance of angular momentum, material frame indifference, material symmetry conditions and thermodynamic consistency. The surrogate modeling technique of choice is an approach called local approximate Gaussian process regression which in contrast to neural networks is a non-parametric model with convergence guarantees.

It is shown that the presented technique vastly outperforms the classical mapping approach which maps the symmetric right Cauchy–Green tensor components to the symmetric components of the second Piola–Kirchhoff stress tensor, without consideration of material frame indifference or material symmetry of the trained model. Surprisingly, the surrogates trained with the presented physics-informed concept were able to accurately capture stress paths which reached far outside the training domain showcasing their ability to generalize efficiently. Furthermore a space-filling sampling technique is proposed that is able to generate evenly spread samples in the invariant space, for isotropic and anisotropic materials, based on some bounded deformation gradient domain. The sampling technique was explained in detail and its effectiveness in comparison to randomly obtained samples from the deformation gradient space was highlighted for isotropic and anisotropic numerical examples. The major limitations of the current approach are that the degree of anisotropy as well as the preferred directions of the material have to be known a-priori. Hence, in future works we aim to approximate these generally unknown quantities and study how the presented framework performs using these approximations. Furthermore, we are interested in investigating the influence of very sparse datasets on the accuracy of the model.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors acknowledge the support by the National Science Foundation under grant no. CMMI-2129776.

Appendix

List of tensor gradient expressions

$$\begin{aligned}
 \frac{\partial I_1}{\partial \mathbf{C}} &= \mathbf{I} \\
 \frac{\partial I_2}{\partial \mathbf{C}} &= I_1 \mathbf{I} - \mathbf{C} \\
 \frac{\partial I_3}{\partial \mathbf{C}} &= I_3 \mathbf{C}^{-1} \\
 \frac{\partial I_4}{\partial \mathbf{C}} &= \mathbf{A} \\
 \frac{\partial I_5}{\partial \mathbf{C}} &= \mathbf{a}_0 \otimes \mathbf{C} \mathbf{a}_0 + \mathbf{a}_0 \mathbf{C} \otimes \mathbf{a}_0 \\
 \frac{\partial \mathbf{I}}{\partial \mathbf{C}} &= \mathbf{0} \otimes \mathbf{0} \\
 \frac{\partial \mathbf{C}}{\partial \mathbf{C}} &= \mathbf{I} \otimes \mathbf{I}
 \end{aligned} \tag{76}$$

$$\begin{aligned}
\frac{\partial \mathbf{A}}{\partial \mathbf{C}} &= \mathbf{0} \otimes \mathbf{0} \\
\left(\frac{\partial \mathbf{C}^{-1}}{\partial \mathbf{C}} \right)_{ijkl} &= -\frac{1}{2} \left(C_{ik}^{-1} C_{lj}^{-1} + A_{il}^{-1} A_{kj}^{-1} \right) \\
\left(\frac{\partial \mathbf{C}^2}{\partial \mathbf{C}} \right)_{ijkl} &= \delta_{ik} C_{lj} + C_{ik} \delta_{jl} \\
\left(\frac{\partial (\mathbf{A}\mathbf{C} + \mathbf{C}\mathbf{A})}{\partial \mathbf{C}} \right)_{ijkl} &= A_{ik} \delta_{jl} + \delta_{ik} A_{lj} \\
\left(\frac{\partial (\mathbf{A}\mathbf{C}^2 + \mathbf{C}^2\mathbf{A})}{\partial \mathbf{C}} \right)_{ijkl} &= A_{ik} C_{lj} + A_{im} C_{mk} \delta_{jl} + \delta_{ik} C_{lr} A_{rj} + C_{ik} A_{lj}
\end{aligned}$$

References

- [1] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* 378 (2019) 686–707.
- [2] T. Kadeethum, T.M. Jørgensen, H.M. Nick, Physics-informed neural networks for solving nonlinear diffusivity and Biot's equations, *PLoS One* 15 (5) (2020) e0232683.
- [3] J.N. Fuhg, N. Bouklas, The mixed deep energy method for resolving concentration features in finite strain hyperelasticity, *J. Comput. Phys.* 451 (2022) 110839.
- [4] J.N. Fuhg, I. Kalogeris, A. Fau, N. Bouklas, Interval and fuzzy physics-informed neural networks for uncertain fields, *Probab. Eng. Mech.* (2022) 103240.
- [5] T. Kadeethum, F. Ballarin, N. Bouklas, Non-intrusive reduced order modeling of poroelasticity of heterogeneous media based on a discontinuous Galerkin approximation, 2021, arXiv preprint arXiv:2101.11810.
- [6] Q. Hernandez, A. Badiás, D. González, F. Chinesta, E. Cueto, Deep learning of thermodynamics-aware reduced-order models from data, *Comput. Methods Appl. Mech. Engrg.* 379 (2021) 113763.
- [7] T. Kadeethum, D. O'Malley, J.N. Fuhg, Y. Choi, J. Lee, H.S. Viswanathan, N. Bouklas, A framework for data-driven solution and parameter estimation of PDEs using conditional generative adversarial networks, *Nat. Comput. Sci.* 1 (12) (2021) 819–829.
- [8] X. Wu, J. Ghaboussi, Representation of material behavior: neural network-based models, in: 1990 IJCNN International Joint Conference on Neural Networks, IEEE, 1990, pp. 229–234.
- [9] J. Ghaboussi, J.H. Garrett, X. Wu, Material modeling with neural networks, in: *Proc. Int. Conf. on Numerical Methods in Engineering: Theory and Applications*, 1990, pp. 701–717.
- [10] J. Ghaboussi, J. Garrett Jr., X. Wu, Knowledge-based modeling of material behavior with neural networks, *J. Eng. Mech.* 117 (1) (1991) 132–153.
- [11] M. Lefik, B.A. Schrefler, Artificial neural network as an incremental non-linear constitutive model for a finite element code, *Comput. Methods Appl. Mech. Engrg.* 192 (28–30) (2003) 3265–3283.
- [12] S. Jung, J. Ghaboussi, Characterizing rate-dependent material behaviors in self-learning simulation, *Comput. Methods Appl. Mech. Engrg.* 196 (1–3) (2006) 608–619.
- [13] D. Huang, J.N. Fuhg, C. Weissenfels, P. Wriggers, A machine learning based plasticity model using proper orthogonal decomposition, *Comput. Methods Appl. Mech. Engrg.* 365 (2020) 113008.
- [14] J.N. Fuhg, C. Böhm, N. Bouklas, A. Fau, P. Wriggers, M. Marino, Model-data-driven constitutive responses: Application to a multiscale computational framework, *Internat. J. Engrg. Sci.* 167 (2021) 103522, <http://dx.doi.org/10.1016/j.ijengsci.2021.103522>, URL: <https://www.sciencedirect.com/science/article/pii/S0020722521000690>.
- [15] X. Lu, J. Yvonnet, L. Papadopoulos, I. Kalogeris, V. Papadopoulos, A stochastic FE2 data-driven method for nonlinear multiscale modeling, *Materials* 14 (11) (2021) 2875.
- [16] H.J. Logarzo, G. Capuano, J.J. Rimoli, Smart constitutive laws: Inelastic homogenization through machine learning, *Comput. Methods Appl. Mech. Engrg.* 373 (2021) 113482.
- [17] Z. Liu, C. Wu, Exploring the 3D architectures of deep material network in data-driven multiscale mechanics, *J. Mech. Phys. Solids* 127 (2019) 20–46.
- [18] Y. Heider, K. Wang, W. Sun, SO (3)-invariance of informed-graph-based deep neural network for anisotropic elastoplastic materials, *Comput. Methods Appl. Mech. Engrg.* 363 (2020) 112875.
- [19] K. Xu, D.Z. Huang, E. Darve, Learning constitutive relations using symmetric positive definite neural networks, *J. Comput. Phys.* 428 (2021) 110072.
- [20] K. Linka, M. Hillgärtner, K.P. Abdolazizi, R.C. Aydin, M. Itskov, C.J. Cyron, Constitutive artificial neural networks: A fast and general approach to predictive data-driven constitutive modeling by deep learning, *J. Comput. Phys.* 429 (2021) 110010.
- [21] F. Masi, I. Stefanou, P. Vannucci, V. Maffi-Berthier, Thermodynamics-based artificial neural networks for constitutive modeling, *J. Mech. Phys. Solids* 147 (2021) 104277.

- [22] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, *Deep Learning*, Vol. 1, MIT Press Cambridge, 2016.
- [23] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, *Neural Netw.* 2 (5) (1989) 359–366.
- [24] Z. Lu, H. Pu, F. Wang, Z. Hu, L. Wang, The expressive power of neural networks: A view from the width, in: *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 6232–6240.
- [25] R.E. Jones, J.A. Templeton, C.M. Sanders, J.T. Ostien, Machine learning models of plastic flow based on representation theory, 2018, arXiv preprint [arXiv:1809.00267](https://arxiv.org/abs/1809.00267).
- [26] C.E. Rasmussen, Gaussian processes in machine learning, in: *Summer School on Machine Learning*, Springer, 2003, pp. 63–71.
- [27] I. Rocha, P. Kerfriden, F. van der Meer, On-the-fly construction of surrogate constitutive models for concurrent multiscale mechanical analysis through probabilistic machine learning, *J. Comput. Phys.* X 9 (2021) 100083.
- [28] J. Wang, T. Li, F. Cui, C.-Y. Hui, J. Yeo, A.T. Zehnder, Metamodeling of constitutive model using Gaussian process machine learning, *J. Mech. Phys. Solids* (2021) 104532.
- [29] J.N. Fuhg, M. Marino, N. Bouklas, Local approximate Gaussian process regression for data-driven constitutive models: development and comparison with neural networks, *Comput. Methods Appl. Mech. Engrg.* 388 (2022) 114217.
- [30] A.L. Frankel, R.E. Jones, L.P. Swiler, Tensor basis Gaussian process models of hyperelastic materials, *J. Mach. Learn. Model. Comput.* 1 (1) (2020).
- [31] A.G. Holzapfel, *Nonlinear Solid Mechanics II*, John Wiley & Sons, Inc., 2000.
- [32] A.E. Ehret, M. Itskov, A polyconvex hyperelastic model for fiber-reinforced materials in application to soft tissues, *J. Mater. Sci.* 42 (21) (2007) 8853–8863.
- [33] J. Zhang, J. Rychlewski, Structural tensors for anisotropic solids, *Arch. Mech.* 42 (3) (1990) 267–277.
- [34] M. Itskov, N. Aksel, A class of orthotropic and transversely isotropic hyperelastic constitutive models based on a polyconvex strain energy function, *Int. J. Solids Struct.* 41 (14) (2004) 3833–3848.
- [35] C.-S. Man, J.D. Goddard, Remarks on isotropic extension of anisotropic constitutive functions via structural tensors, *Math. Mech. Solids* 23 (4) (2018) 554–563.
- [36] Q.-S. Zheng, Theory of representations for tensor functions—a unified invariant approach to constitutive equations, 1994.
- [37] H. Xiao, General irreducible representations for constitutive equations of elastic crystals and transversely isotropic elastic solids, *J. Elasticity* 39 (1) (1995) 47–73.
- [38] J. Schröder, P. Neff, Invariant formulation of hyperelastic transverse isotropy based on polyconvex free energy functions, *Int. J. Solids Struct.* 40 (2) (2003) 401–445.
- [39] D. Balzani, P. Neff, J. Schröder, G.A. Holzapfel, A polyconvex framework for soft biological tissues. Adjustment to experimental data, *Int. J. Solids Struct.* 43 (20) (2006) 6052–6070.
- [40] J. Ghaboussi, D.A. Pecknold, M. Zhang, R.M. Haj-Ali, Autoprogressive training of neural network constitutive models, *Internat. J. Numer. Methods Engrg.* 42 (1) (1998) 105–126.
- [41] Y. Hashash, S. Jung, J. Ghaboussi, Numerical implementation of a neural network based material model in finite element analysis, *Internat. J. Numer. Methods Engrg.* 59 (7) (2004) 989–1005.
- [42] M. Lefik, D. Boso, B. Schrefler, Artificial neural networks in numerical modelling of composites, *Comput. Methods Appl. Mech. Engrg.* 198 (21–26) (2009) 1785–1804.
- [43] C. Miehe, Numerical computation of algorithmic (consistent) tangent moduli in large-strain computational inelasticity, *Comput. Methods Appl. Mech. Engrg.* 134 (3–4) (1996) 223–240.
- [44] A.G. Baydin, B.A. Pearlmutter, A.A. Radul, J.M. Siskind, Automatic differentiation in machine learning: a survey, *J. Mach. Learn. Res.* 18 (2018) 1–43.
- [45] M. Stein, Large sample properties of simulations using latin hypercube sampling, *Technometrics* 29 (2) (1987) 143–151.
- [46] F.A. Viana, G. Venter, V. Balabanov, An algorithm for fast optimal latin hypercube design of experiments, *Internat. J. Numer. Methods Engrg.* 82 (2) (2010) 135–156.
- [47] P.J. Van Laarhoven, E.H. Aarts, Simulated annealing, in: *Simulated Annealing: Theory and Applications*, Springer, 1987, pp. 7–15.
- [48] B. Chazelle, An optimal convex hull algorithm in any fixed dimension, *Discrete Comput. Geom.* 10 (4) (1993) 377–409.
- [49] X. Xu, K. Harada, Automatic surface reconstruction with alpha-shape method, *Vis. Comput.* 19 (7) (2003) 431–443.
- [50] P. Currie, The attainable region of strain-invariant space for elastic materials, *Int. J. Non-Linear Mech.* 39 (5) (2004) 833–842.
- [51] W.S. Burnside, A.W. Panton, *The Theory of Equations: With an Introduction to the Theory of Binary Algebraic Forms*, Hodges, Figgis, 1892.
- [52] G.E. Box, A note on the generation of random normal deviates, *Ann. Math. Stat.* 29 (1958) 610–611.
- [53] P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S.J. van der Walt, M. Brett, J. Wilson, K.J. Millman, N. Mayorov, A.R.J. Nelson, E. Jones, R. Kern, E. Larson, C.J. Carey, Í. Polat, Y. Feng, E.W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimman, I. Henriksen, E.A. Quintero, C.R. Harris, A.M. Archibald, A.H. Ribeiro, F. Pedregosa, P. van Mulbregt, SciPy 1.0 Contributors, *SciPy 1.0: Fundamental algorithms for scientific computing in python*, *Nature Methods* 17 (2020) 261–272, [http://dx.doi.org/10.1038/s41592-019-0686-2](https://doi.org/10.1038/s41592-019-0686-2).
- [54] J.N. Fuhg, A. Fau, U. Nackenhorst, State-of-the-art and comparative review of adaptive sampling methods for kriging, *Arch. Comput. Methods Eng.* (2020) 1–59.
- [55] J.D. Svenson, T.J. Santner, *Multiobjective Optimization of Expensive Black-Box Functions Via Expected Maximin Improvement*, Vol. 32, The Ohio State University, Columbus, Ohio, 2010.
- [56] L. Laurent, R. Le Riche, B. Soulier, P.-A. Boucard, An overview of gradient-enhanced metamodels with applications, *Arch. Comput. Methods Eng.* 26 (1) (2019) 61–106.

- [57] B. Matérn, Spatial Variation, in: Lecture Notes in Statistics, vol. 36, 1960.
- [58] R.B. Gramacy, D.W. Apley, Local Gaussian process approximation for large computer experiments, *J. Comput. Graph. Statist.* 24 (2) (2015) 561–578.
- [59] J.P. Kleijnen, W.C. van Beers, Prediction for big data through Kriging: small sequential and one-shot designs, *Amer. J. Math. Management Sci.* 39 (3) (2020) 199–213.
- [60] J.N. Fuhg, Adaptive surrogate models for parametric studies, 2019, arXiv preprint [arXiv:1905.05345](https://arxiv.org/abs/1905.05345).
- [61] R.B. Gramacy, laGP: large-scale spatial modeling via local approximate Gaussian processes in R, *J. Stat. Softw.* 72 (1) (2016) 1–46.
- [62] MATLAB Version 9.10.0 (R2021a), The MathWorks Inc., Natick, Massachusetts, 2021.
- [63] S.N. Lophaven, H.B. Nielsen, J. Søndergaard, et al., DACE: a Matlab Kriging Toolbox, Vol. 2, Citeseer, 2002.
- [64] J. Bonet, A. Burton, A simple orthotropic, transversely isotropic hyperelastic constitutive equation for large strain computations, *Comput. Methods Appl. Mech. Engrg.* 162 (1–4) (1998) 151–164.