



# A new texture descriptor for data-driven constitutive modeling of anisotropic plasticity

Jan Schmidt<sup>1,\*</sup> , and Alexander Hartmaier<sup>1</sup> 

<sup>1</sup> ICAMS, Ruhr-Universität Bochum, Universitätsstraße 150, 44801 Bochum, Germany

Received: 28 April 2023

Accepted: 8 August 2023

© The Author(s), 2023

## ABSTRACT

Constitutive modeling of anisotropic plastic material behavior traditionally follows a deductive scheme, relying on empirical observations that are cast into analytic equations, the so-called phenomenological yield functions. Recently, data-driven constitutive modeling has emerged as an alternative to phenomenological models as it offers a more general way to describe the material behavior with no or fewer assumptions. In data-driven constitutive modeling, methods of statistical learning are applied to infer the yield function directly from a data set generated by experiments or numerical simulations. Currently these data sets solely consist of stresses and strains, considering the microstructure only implicitly. Similar to the phenomenological approach, this limits the generality of the inferred material model, as it is only valid for the specific material employed in the virtual or physical experiments. In this work, we present a new generic descriptor for crystallographic texture that allows an explicit consideration of the microstructure in data-driven constitutive modeling. This descriptor compromises between generality and complexity and is based on an approximately equidistant discretization of the orientation space. We prove its ability to capture the structure–property relationships between a variety of cubic–orthorhombic textures and their anisotropic plastic behavior expressed by the yield function Yld2004-18p. Three different machine learning models trained with the descriptor can predict yield loci as well as  $r$ -values of unseen microstructures with sufficient accuracy. The descriptor allows an explicit consideration of crystallographic texture, providing a pathway to microstructure-sensitive data-driven constitutive modeling.

Handling Editor: Ghanshyam Pilania.

Address correspondence to E-mail: jan.schmidt-p2d@rub.de

E-mail Addresses: alexander.hartmaier@rub.de

<https://doi.org/10.1007/s10853-023-08852-2>

Published online: 07 September 2023

## Introduction

Machine learning (ML) or, more generally, data science-based approaches to solve boundary value problems in continuum mechanics have emerged as a popular research topic within materials science in recent years. In a comprehensive summary, Bock et al. [1] order various approaches along the process–structure–property–performance (PSPP) chain and assign them to various subproblems and application fields. In this work, we focus on the application field of constitutive modeling, which is the decisive link between stimulus and response, i.e., between stresses and strains, by a set of material laws.

The material model of interest in this work is the yield function of anisotropic, polycrystalline metals. In classic plasticity theory, the yield function is associated with a plastic potential. It describes whether a certain load leads to elastic or plastic material deformation and defines the direction of the plastic flow, in case of an associative flow rule. Considerable research has been directed toward the development of such phenomenological yield functions for anisotropic materials, where the yielding depends on the loading direction. To capture directional dependency, a linear transformation of the stress tensor has proved to be an essential step in formulating these advanced yield functions [2]. This transformation introduces a number of material-specific coefficients that have to be determined by experimental tests or numerical simulations. The number of required coefficients is determined by the particular formulation, which in turn depends on the degree of anisotropy in the material and the stress space applied to. It varies between six for less anisotropic materials [3] and plane stress formulation to 27 for profound anisotropy and full stress formulation [4]. As analytical material laws, phenomenological yield functions are numerically efficient and have proved to be accurate material models, which are widely employed in continuum-scale FEM simulations, e.g., in the sheet metal forming industry.

On the other hand, they rely on user experience and detailed knowledge of the material [5], show difficulties in the parametrization due to the non-uniqueness of the material coefficients [6] and face potential limitations to the class of materials they have been developed for. Data-driven constitutive modeling approaches to circumvent these shortcomings have recently become an active research topic. They can be ordered according to how profoundly the

data paradigm replaces the established relationships and concepts. On the one end of the spectrum stands the model-free approach by Kirchdoerfer and Ortiz [7] and the extensions by Eggersmann et al. [8]. Their approach, referred to as data-driven computing, completely bypasses the empirical constitutive model and determines the material response by a closest point search in a data set of prescribed experimental and/or simulated states. In a similar way, Chinesta et al. [9] describe the plastic material behavior by constitutive manifolds, also purely based on data. While their approaches arguably represent the most consistent use of the data paradigm to solve continuum mechanics problems, they also require an extremely large amount of mechanical data from experiments or simulations, which limits them in scenarios where only sparse data are available.

To reduce the amount of data needed and to recover the computational efficiency of analytical yield functions, researchers have started to use trained ML models as yield functions. To derive the constitutive behavior, the models are trained on data obtained from experiments or crystal plasticity (CP) simulations and are then used as predictive material models, for example inside the integration points of an FEM calculation. The use of CP simulations offers an efficient way for hierarchical materials modeling as the constitutive behavior on the continuum scale is coupled to CP calculations on the microscale. Lefik and Schrefler [10] present a neuronal network approach to describe the nonlinear stress–strain relationship in superconducting fibers and implemented it into an FE model. In a more recent approach, Huang et al. [11] employ a deep neuronal network (DNN) based on proper orthogonal decomposition to describe history-dependent plastic behavior. Vlassis and Sun [12] address the well-known interpretability issues of DNNs by proposing a level-set approach to cover also complex hardening phenomena. Nascimento et al. [13] automate the critical step of network design by applying Bayesian optimization to obtain optimal network architectures that learn convexity and iso-sensitivity. Although the majority of research focuses on DNN-based constitutive models, also other ML function classes have been employed recently to address the shortcomings of DNNs. Hartmaier [14], Shoghi and Hartmaier [15] formulate the yield function as binary classification problem and apply support vector machines (SVR) in combination with an effective data sampling strategy to reduce the amount of required training data.

Rocha et al. [16] apply Gaussian process regression (GPR) to derive constitutive relationships in a model-free approach. Fuhs and Bouklas [17] extend this concept and include physics-based principles to the model such as the preservation of the stress-free undeformed configuration, material frame indifference or thermodynamic consistency.

However, as with phenomenological yield functions, all of the approaches presented above treat the microstructure implicitly in a black-box fashion. The database required for parametrization or for training consists purely of stress–strain data, obtained from experimental tests or CP simulations where the microstructure is modeled by a representative volume element (RVE). Although this enables computationally efficient hierarchical materials modeling, no microstructural degrees of freedom are explicitly taken into account. The motivation for introducing microstructural degrees of freedom to data-driven constitutive modeling is twofold. On the one hand, this enables the description of microstructural evolution, and on the other hand, more general models can be trained that are able to describe the constitutive behavior of materials at different microstructural states. If only stresses and strains are used as degrees of freedom, changes in the material and the microstructure would require the generation of a completely new data set to infer the material model, which can be costly.

If one accepts the advantages of explicit microstructural description, the question arises which microstructural features should be considered and how they should be described. Of all microstructural features, texture, i.e., the orientation of grains within the polycrystalline microstructure, is the main source for anisotropic plastic behavior in metals [18]. Ali et al. [19] present a DNN approach for plane stress applications, where besides the strains also the individual orientations of all grains are incorporated into the input layer of the network by their Euler angle triplet. Fuhs et al. [5] use a component model similar to Luecke et al. [20], Pospiech et al. [21] in their neuronal network input space, where the texture is described by pairs of central orientation and spread. In both approaches, the descriptor lacks generality. In the prior, it directly depends on the number of grains used in the CP simulations. If larger RVEs are used, the input dimension and hence the architecture of the whole network change. In the latter, the descriptor is directly related to the number of peak orientations in the texture. A texture with

multiple maxima will require different, i.e., higher-dimensional descriptors than a texture with less maxima. Deformation-induced texture evolution cannot be described by this descriptor since the architecture of the network cannot be changed on the fly.

These concerns motivate the formulation of a different texture descriptor for data-driven constitutive modeling to allow a generic description of crystallographic texture. The necessary condition for such a descriptor is that it is able to capture the structure–property (s–p) relationship between texture and anisotropic plastic material behavior. It should further allow the description of a variety of different textures with a sufficient degree of accuracy. On the other hand, its dimensionality should not be too high to reduce the computational complexity in the data-driven models it is used in. In this work, we propose such a descriptor for cubic–orthorhombic textures, which are frequently observed in sheet metal forming industry. We prove the descriptor’s ability to capture the desired s–p relationship by relating the coefficients of the phenomenological yield function  $\text{Yld2004-18p}$  to the crystallographic texture. To infer the s–p relationship, we extend the scheme presented in an earlier work [22] where we determined the s–p relationship between the yield function and simple one-dimensional Goss and copper textures.

The paper is structured as follows: In Section “[Methods](#),” we introduce the data-driven framework that is employed to find the s–p relationship. The subsections introduce the texture descriptor (Section “[Texture descriptor](#)”), recall the phenomenological yield function (Section “[Anisotropic yield function](#)”), describe the parametrization procedure used to generate the data set (Section “[Parameterization of the yield function](#)”) and present the ML model in which the descriptor is used (Section “[Training ML models](#)”). Afterward, in Section “[Results](#),” we firstly introduce the different data sets used for training and testing of the ML models (Section “[Data set](#)”). In the following subsections, the training and validation results are presented, followed by a study of the models’ generalization property in terms of yield surface and  $r$ -values. In Section [Discussion](#), the results are discussed under the viewpoint of the required descriptor dimensionality, before the most important conclusions of this work are summarized in Section “[Conclusion](#).”

## Methods

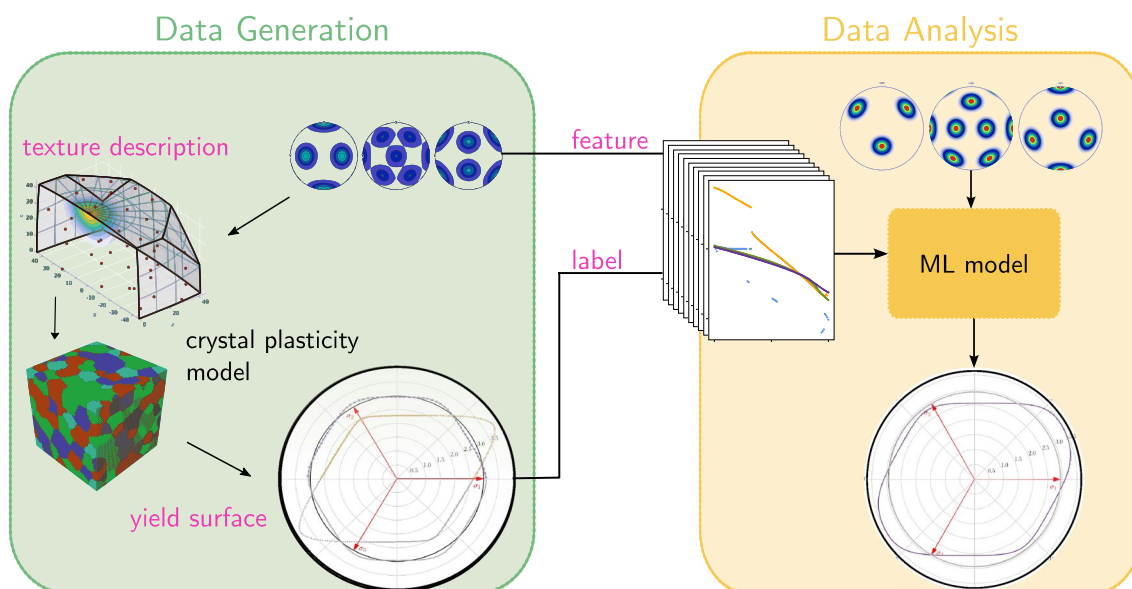
This section introduces the texture descriptor and the data-driven scheme in which it is applied. The latter is shown in Fig. 1; its goal is to determine a structure–property (s–p) relationship between the crystallographic texture and the anisotropic yield function to show the meaningfulness of the texture descriptor in data-driven constitutive modeling. The scheme extends the concept in [22] and will be briefly summarized in the following with a focus on the extensions. It consists of two blocks: the data generation process and the data analysis. As part of the data generation process, data points in terms of feature and label are generated. Firstly, the label, i.e., the texture descriptor  $\mathbf{x}$ , is determined for a given microstructure. Then, an RVE is generated for this microstructure, and a Taylor-based hybrid CP model is used to generate stress–strain curves in different loading directions. Subsequently, these data are used to parameterize the anisotropic yield function Yld2004-18p for the specific microstructure, resulting in the set of anisotropic coefficients, that are used as labels  $\mathbf{y}$ . Repeating these steps for various, different microstructures produces a data set, which is analyzed by employing supervised ML methods to determine the sought s–p relationship

between texture and yield surface. In the following, the methods used in each step are briefly explained.

### Texture descriptor

The descriptor we are looking for faces two fundamentally conflicting requirements arising from its use in a data-based process: On the one hand, it should be as general as possible to describe a wide range of textures in detail. On the other hand, it should be as compact as possible to train meaningful models with a limited number of data points and avoid the curse of dimensionality.

In their DNN constitutive model, Ali et al. [19] describe the texture by the set of all orientations  $g$  present in the microstructure. The orientations are characterized by the Bunge Euler angle triplets  $(\varphi_1, \Phi, \varphi_2)$ . The problem with this approach is the direct dependency of the descriptor on the number of crystallites. Microstructures with different numbers of grains cannot be described by the same descriptor, which limits the generality of the descriptor. The component model by Fuhs et al. [5] describes the texture by pairs of central orientation and spread. Here, the descriptor is directly related to the number of peak orientations in the texture, which allows a flexible description of different textures, but limits the generality in the same way as the prior descriptor. Furthermore,



**Figure 1** Data-driven scheme to identify s–p relationship between crystallographic texture (feature) and anisotropic yield surface (label) with a supervised ML approach. The methods

for the data generation as well as for the data analysis block are explained in Sections “[Texture descriptor–Training ML models](#)”.

deformation-induced texture evolution where the position and number of maxima changes due to the rotation of the grains are not considered by this approach.

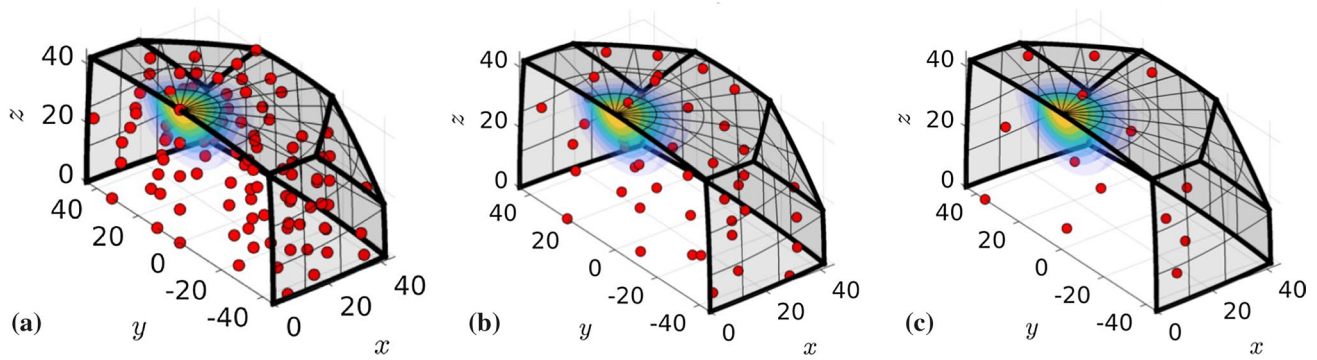
To avoid these shortcomings, we follow a more generic approach and focus on the common base of all textures, the orientation space. The starting point of our considerations is the orientation distribution function (ODF). It is defined as a mapping from the space of orientations  $SO(3)$  to the real numbers and assigns to each orientation  $g \in SO(3)$  the volume fraction of crystallites in a polycrystalline sample volume that are within  $dg$  around  $g$ . According to Bunge [23], the ODF can be represented by a series expansion based on generalized spherical harmonics (GSH). The Fourier coefficients of this expansion would meet the above requirements in such a way that they represent the most general and established method for texture description. However, in order to achieve this generality, the series expansion is truncated usually between 22 and 32 harmonics, yielding a potentially very high-dimensional input space [24]. Montes de Oca Zapiain et al. [25] use GSH coefficients to relate cubic–orthorhombic textures to a more simple anisotropic material behavior defined by Hill’s yield criterion [26]. Their analysis shows that the later the series is truncated, the better the accuracy of the trained models was. The maximum truncation point in their study is chosen after twelve harmonics, yielding a texture descriptor containing 32 distinct GSH coefficients. With this descriptor, the anisotropic coefficients of the yield function could be predicted quite accurately.

The descriptor we propose compromises between generality and complexity. It is based on evaluating the ODF over an approximately equidistant grid of resolution  $\theta$  in the orientation space  $SO(3)$ . Different methods have been proposed in literature to create an approximately equidistant grid in  $SO(3)$  [27–31]. The approach followed in this work is similar to the NED proposed by Helming [27, 28] and taken from the MATLAB toolbox MTEX [32]. It begins with sampling an approximately equidistant distribution of points on a sphere. Each point on the sphere can be described by a combination of azimuth and polar angle, associated here with the first two Euler angles  $\varphi_1$  and  $\Phi$ . The range of  $\Phi$ , bounded by the cubic–orthorhombic fundamental zone, is divided into  $n_\Phi$  sections of equal distance. These sections form latitudes on the sphere, separated by the grid resolution  $\theta$ . On each latitude,  $n_{\varphi_1}$  samples are distributed that again have

an angular distance of  $\theta$  to each other. As the angle  $\Phi$ , also  $\varphi_1$  is bounded by the fundamental zone, limiting the circular arc that is populated with samples. This approach results in an approximately equidistant grid of  $\varphi_1, \Phi$ -pairs on the sphere section, bounded by the cubic–orthorhombic fundamental zone. To complete the desired grid in  $SO(3)$ , the third Euler angle  $\varphi_2$  is divided into  $n_{\varphi_2}$  equal  $\theta$ -sections as well. The approximately equidistant grid within MTEX is then fully characterized by the tensor product between the approximately equidistant grid on the sphere for  $\Phi, \varphi_1$  and the equi-distribution w.r.t  $\varphi_2$

Different values of  $\theta$  lead to different values of  $n_{\varphi_1}, n_\Phi, n_{\varphi_2}$  and thus to a different number of grid nodes  $m$ . On each of these nodes, the ODF of the texture being described is evaluated. The obtained value is associated with the intensity of the ODF, expressed in multiples of a random distribution (MRD). Arranging all of the recorded intensities in vector notation yields the texture descriptor used in this work.

The concept of the descriptor is visualized in Fig. 2. Each sub-figure shows the same section of the orientation space discretized by a grid of different resolution. The Rodriguez–Frank representation is chosen here to visualize the approximately equidistant character of the grid. Since the focus in this work lies on rolling textures, the orientation space can be reduced to the cubic–orthorhombic fundamental zone, assuming crystal and specimen symmetry. The restriction to this section reduces the number of nodes and by that the complexity of the descriptor. However, the concepts and methods employed here also apply to textures with lower symmetries. The red dots in Fig. 2 correspond to the nodes of the equidistant grid on which the ODF is evaluated. The number of nodes decreases with decreasing grid resolution  $\theta$ , yielding a lower-dimensional texture descriptor. If the resolution of the grid is increased, the number of nodes increases and with this the dimensionality of the texture descriptor. In that sense, the descriptor embodies the two conflicting goals mentioned above and controls the trade-off between detailed texture description and low-dimensional representation. To find the optimum grid resolution,  $\theta$  is varied between  $11^\circ$  and  $30^\circ$ , resulting in nine different descriptors ranging from 7 to 111 dimensions. This range is motivated by the findings in [33] where the influence of grid resolution on ODF reconstruction has been studied. The grids for  $\theta \in [11^\circ, 15^\circ, 23^\circ]$  are exemplarily shown in Fig. 2. Table 1 introduces the acronyms



**Figure 2** Approx. equidistant grids in cubic–orthorhombic fundamental zone of Rodriguez–Frank space for different resolutions  $\theta = 11^\circ$  (a),  $\theta = 15^\circ$  (b) and  $\theta = 23^\circ$  (c). Red dots are grid nodes

**Table 1** Overview of the different descriptor variants studied in this paper

Acronym	$\theta$ [°]	# Grid points
7-d	30	7
9-d	27	9
12-d	24	12
16-d	23	16
25-d	21	25
36-d	18	36
48-d	15	48
72-d	13	72
111-d	11	111
111-d+PCA <sup>a</sup>	11	111

<sup>a</sup>Linear principal component analysis was applied to reduce dimensions (see Section “[Training and cross-validation](#)”)

for the different descriptors used in this work together with their properties.

### Anisotropic yield function

The material property for which the meaningfulness of the descriptor is to be shown is anisotropic plastic behavior. In this work, anisotropic plastic behavior is described by a variant of the Barlat Yld2004-18p yield function, a widely used phenomenological yield function in sheet metal forming [2]. The yield function is formulated as the generalized form of two isotropic Hosford yield criteria,

on which the example ODF centered at  $(0^\circ, 0^\circ, 40^\circ)$  is evaluated. The measured intensities of the ODF at the gridpoints are vectorized and used as texture descriptor.

$$\begin{aligned} \phi &= \phi(\Sigma) = \phi(\tilde{S}', \tilde{S}'') \\ &= |\tilde{S}'_1 - \tilde{S}''_1|^a + |\tilde{S}'_1 - \tilde{S}''_2|^a + |\tilde{S}'_1 - \tilde{S}''_3|^a + |\tilde{S}'_2 - \tilde{S}''_1|^a \\ &\quad + |\tilde{S}'_2 - \tilde{S}''_2|^a + |\tilde{S}'_2 - \tilde{S}''_3|^a + |\tilde{S}'_3 - \tilde{S}''_1|^a + |\tilde{S}'_3 - \tilde{S}''_2|^a \\ &\quad + |\tilde{S}'_3 - \tilde{S}''_3|^a = 4\sigma_y^a. \end{aligned} \tag{1}$$

The material-specific exponent  $a$  is set to 8 for fcc metals [2].  $S'_i$  and  $S''_i$  are the principal values of the transformed deviatoric stresses. They are obtained from solving the characteristic equation of the two linear transformed deviatoric stress tensors  $\tilde{s}', \tilde{s}''$

$$\begin{aligned} \tilde{s}' &= C' s \\ \tilde{s}'' &= C'' s, \end{aligned} \tag{2}$$

where the linear transformation is based on the two anisotropy tensors  $C'$  and  $C''$

$$\begin{aligned} C' &= \begin{bmatrix} 0 & -c'_{12} & -c'_{13} & 0 & 0 & 0 \\ -c'_{21} & 0 & -c'_{23} & 0 & 0 & 0 \\ -c'_{31} & -c'_{32} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c'_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & c'_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & c'_{66} \end{bmatrix}, \\ C'' &= \begin{bmatrix} 0 & -c''_{12} & -c''_{13} & 0 & 0 & 0 \\ -c''_{21} & 0 & -c''_{23} & 0 & 0 & 0 \\ -c''_{31} & -c''_{32} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & c''_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & c''_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & c''_{66} \end{bmatrix}. \end{aligned}$$

The anisotropy tensors are defined in the material reference frame that is aligned along the orthotropic symmetry axes. In the original formulation, the number of anisotropic coefficients is 18. However, it was shown by van den Boogaard et al. [6] that this formulation obeys a non-uniqueness of the anisotropic coefficients and that the same yield function can be described by different sets of anisotropic coefficients. To resolve this non-uniqueness, they propose to set the coefficients  $c'_{12}$  and  $c'_{13}$  to one.

Following the approach in [22], the number of coefficients is further reduced by limiting the yield function to the principal stress space, i.e., allowing only diagonal stresses in the basis of the principal directions of the orthotropic material reference frame. The main reason for this limitation is that it enables the application of the numerically efficient crystallographic yield locus method (CYL) by Biswas et al. [34]. This method replaces computational costly CPFEM simulations that are required to parameterize the yield loci for each texture. Therefore, the diagonal coefficients four to six in  $\mathbf{C}'$  and  $\mathbf{C}''$  can be set to zero, i.e.,  $c'_{44} = c'_{55} = c'_{66} = c''_{44} = c''_{55} = c''_{66} = 0$ , reducing the number of anisotropic coefficients and thus the dimensionality of the output space from 16 to 10.

### Parameterization of the yield function

After clarifying how structure and property are described, this section presents how exactly the properties for each structure are determined, i.e., how the set of anisotropic coefficients is identified for a given texture. This process is referred to as parameterization of the yield function. As mentioned in Section “Introduction,” it requires a set of stress states at yield onset. These stress states are obtained by applying the virtual lab approach introduced by Zhang et al. [35] where a CP model is used instead of experiments to determine these stress states for different loading directions.

The CP model applied in this work is a hybrid scheme, introduced by Biswas et al. [34]. It is based on the Taylor-type approach to crystallographic yield loci by Van Houtte et al. [36] and combines the accuracy of full-field CPFEM calculations with the efficiency of a simple Taylor model. Its central idea is to relax the constraints imposed by the Taylor assumption on the yield surface by calibrating the latter to results obtained from CPFEM. The calibration requires only two CPFEM calculations performed on an representative volume element (RVE)

for each texture, allowing the generation of large amounts of data in a short time. For details on the algorithm, the reader is referred to [34]. In this work, the two required CPFEM calculations are performed in the commercial FEM code Abaqus using a UMAT for the crystal plasticity. The crystal plasticity constitutive model and the used material parameters are given in Appendix. An RVE of 2197 linear eight-node brick elements with reduced integration (C3D8R) is generated to describe the crystallographic texture. Each element represents a cube-shaped grain with an edge length of 0.002 mm. It is noted here that these simulations are only evaluated up to the yield point of the model, for which the representation of each grain by only one element is justified. For simulations of the strain hardening behavior, it is recommended to use at least 8 elements per grain in such models. The required 2197 discrete orientations are sampled from the known ODF employing the texture reconstruction scheme by Biswas et al. [33] which is based on the integer approximation method. Periodic boundary conditions are applied on the RVE, and a homogenized macroscopic stress is imposed by applying Neumann boundary conditions at three representative nodes of the RVE [37].

From the hybrid scheme, 60 stress states at yield onset are obtained to which the yield function is fitted. This parameterization is done by minimizing the error function

$$E(\mathbf{c}) = \frac{\|\boldsymbol{\phi}(\mathbf{s}, \mathbf{c})\|_2}{\bar{\sigma}}. \quad (3)$$

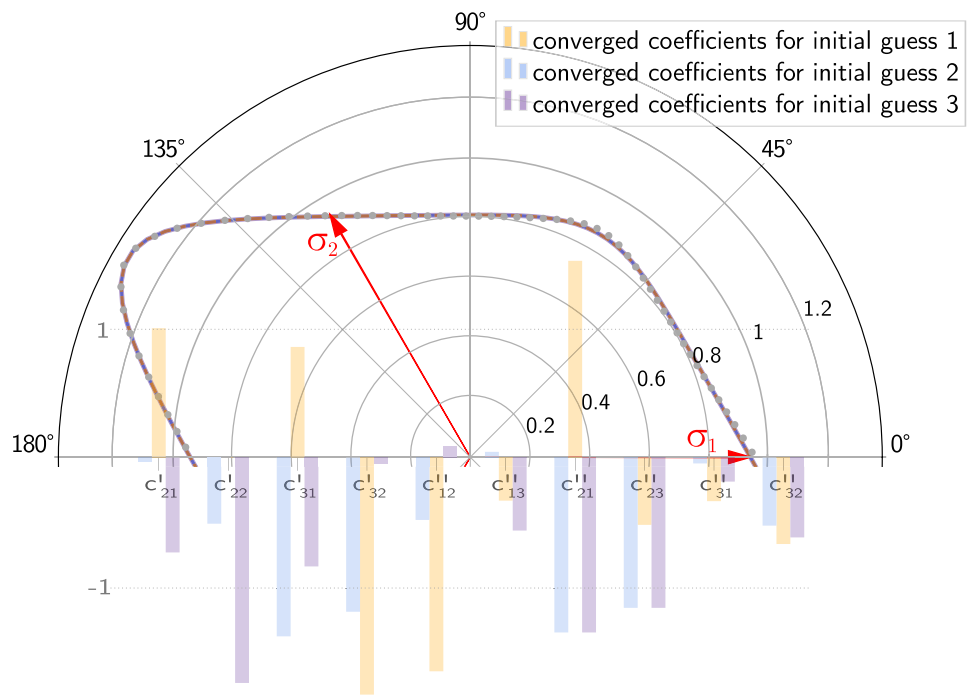
In Eq. (3),  $\boldsymbol{\phi}$  is a vector containing the yield function values for all 60 stress states from the CP model  $\mathbf{s}$ , and  $\bar{\sigma}$  is the isotropic yield strength equivalent. Each entry in  $\boldsymbol{\phi}$  is obtained by evaluating the yield function  $\phi$  (Eq. (1)) with the current set of anisotropic coefficients  $\mathbf{c}$  on the corresponding stress state at yield onset  $\mathbf{s}$  obtained from the CYL. If the yield function is accurately parameterized, this term will be close to zero since the yield function is zero at yield onset per definition. Thus, minimizing Eq. (3) with  $\mathbf{c}$  as independent variables results in the desired set of anisotropic coefficients and a parameterized yield function. For the minimization, the trust region algorithm implemented in the SciPy python library [38] is applied due to its advantages for non-convex optimization [39].

To address the well-known non-uniqueness between the anisotropic coefficients and the yield

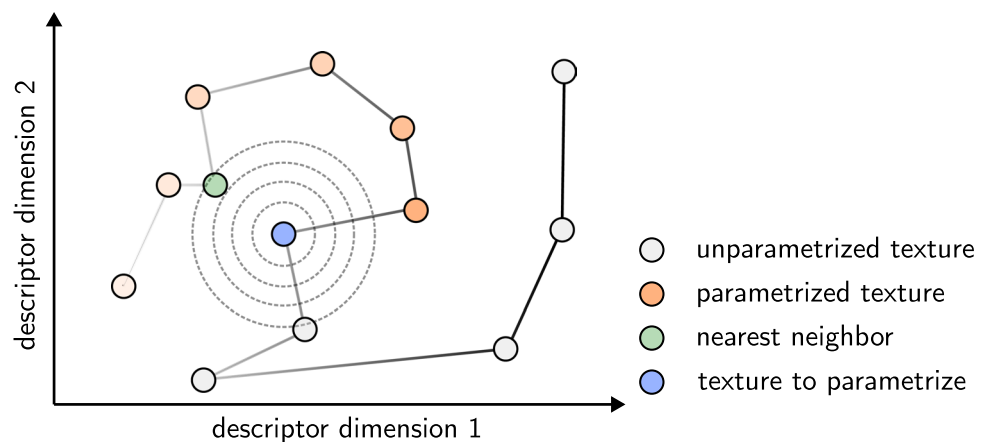
function, a regularization approach is required. The problem is visualized in Fig. 3. Initializing the minimization scheme mentioned above with three different initial guesses leads to three different converged sets of anisotropic coefficients, all with the same error score. The converged coefficients  $c'_{21}$  to  $c''_{32}$  are shown as bars in the figure with different colors for the three initial guesses. The point-symmetric yield loci corresponding to each set of coefficients are plotted above the bar chart in the cylindrical principal stress space together with the stress states at yield onset as gray dots. The definition of cylindrical principal stresses follows the derivations in Hartmaier [14]. It can be

seen that although the coefficients differ in all three converged sets, the yield loci are indistinguishable. If a s–p relationship is to be identified between the texture and the coefficients of the yield surface, the question arises which coefficients should be accepted for a certain texture. In [22], the authors argue that similar textures should have similar coefficients. In other words, small changes in the texture should not lead to large changes in coefficients. However, their approach was limited to the one-dimensional texture description. In this work, we extend the idea and adopt the concept to the new texture descriptor.

**Figure 3** Non-uniqueness between the set of anisotropic coefficients and the yield surface. Colored bars represent converged sets of anisotropic coefficients for three different initial guesses. The corresponding centro-symmetric yield loci are normalized by the average yield stress and plotted in the upper half of the  $\pi$ -plane. Although the coefficients are different, the yield functions they represent are indistinguishable.



**Figure 4** K-nearest-neighbor regularization approach to parameterize the yield function for each texture in the data set. The texture descriptor is simplified to two dimensions.





The regularization approach followed here is visualized in Fig. 4. It is based on the k-nearest-neighbor algorithm. For visualization purposes, the texture descriptor space is simplified to two dimensions and an example set of thirteen textures is shown as circles. For the blue texture  $x_1$  whose yield function is to be parameterized, first the 20 nearest neighbors in the data set of all textures are determined. To quantify the nearness of two textures  $x_1, x_2$ , the Euclidean distance in the texture descriptor space is used:

$$d = \|x_1 - x_2\|_2. \tag{4}$$

Then each of the 20 neighboring textures, starting with the closest, i.e., most similar, is queried to see whether it has already been parameterized. If yes, its converged coefficients are used as initial guess for the parameterization of the current texture  $x_1$ . If not, the next nearest neighbor is consulted. In Fig. 4, the nearest, parameterized neighbor is colored green. If none of the nearest neighbors is parameterized, the value one is chosen as initial guess for each coefficient. This method has the advantage that similar textures have similar coefficients, and thus, discontinuities due to large jumps in the parameter space of the searched function between texture and yield surface are reduced. On the other hand, the necessary resources are also reduced, since the initial guess is closer to the desired minimum than a completely random initial guess, and thus, the trust region algorithm converges faster. To increase the probability of finding already parameterized textures among the k-nearest neighbors, a certain order in which the textures in the data set are parameterized is specified in advance. Starting from a random initial texture, the distance metric in Eq. (4) is used to determine the next texture in the data set that is not yet parameterized. That texture is then parameterized next. Searching is repeated until all textures in the data set have been parameterized.

After all textures have been parameterized, the process of data generation is completed and the data set that is passed to the analysis block. Each data point in the set is composed of texture descriptor and the respective set of 10 anisotropic coefficients.

### Training ML models

In the data analysis block, three different supervised ML function classes are trained on the data set to identify the s–p relationship between texture and anisotropic

coefficients. Thus, they have to address the multivariate regression problem of finding the function

$$f(\text{texture}) = \text{anisotropic coefficients} \tag{5}$$

$$\mathbb{R}^m \rightarrow \mathbb{R}^{10},$$

which is mapping from the  $m$ -dimensional space of the texture descriptor to the 10-dimensional space of the coefficients. The value of  $m$  depends on the resolution of the grid in  $SO(3)$ . The function classes applied here are the  $\epsilon$  support vector regression ( $\epsilon$ -SVR) [40], multiple-output least-squares support vector regression (MLSSVR) [41] and the random forest algorithm (RFR) [42]. The latter two are so-called multi-output function classes. Here, a single model instance is trained that represents all output dimensions in one model. In contrast, the classical  $\epsilon$ -SVR trains one instance per output dimension. In the case of 10 anisotropic coefficients, 10 different models are created and trained independently of each other. The advantage of the multi-output function classes is that the correlations of the output dimensions are taken into account during training, which can lead to higher accuracy if there is a dependency between the coefficients [43].

To compare the different function classes and tune their hyperparameters, a nested ten–sevenfold cross-validation (CV) scheme is applied. The outer loop of the CV is used for model selection: The training data are split in ten folds. Nine of the ten folds are used to train one of the three function classes, whereas the remaining fold is used to validate the training performance. The nine training folds enter the inner CV loop, which is sevenfold. This loop is used for hyperparameter optimization, which is essential for successful training. We follow a grid search logic, where the number of hyperparameters to be determined depends on the respective function class. The grid points, from which the hyperparameter combinations arise, are listed in Table 2 for each function class.

To evaluate the models during training, validation and testing, the mean-squared error is calculated. It is generally formulated as

$$\text{MSE} = \frac{1}{n * 10} \sum_{i=1}^n \sum_{m=1}^{10} \left( y_m^{(i)} - \hat{y}_m^{(i)} \right)^2, \tag{6}$$

where  $n$  is the number of data points on which it is evaluated,  $y_m^{(i)}$  is the  $m$ th true anisotropic coefficient of data point  $i$  and  $\hat{y}_m^{(i)}$  is the prediction. After the three function classes are trained, their generalization

**Table 2** Hyperparameter grid points for each function class

SVR	MLSSVR	RFR
$C \in \{10^i, \text{ for } i \in [-4 \dots 4]\}$	$p \in \{2^i, \text{ for } i \in [-15 : 2 : 3]\}^a$	$n_{\text{trees}} \in \{10, 100, 500, 1000\}$
$\gamma \in \{10^i, \text{ for } i \in [-5 \dots 0]\}$	$\gamma \in \{2^i, \text{ for } i \in [-2 \dots 5]\}$	
$k(\mathbf{x}, \mathbf{x}) \in \{\text{rbf}, \text{linear}\}$	$k(\mathbf{x}, \mathbf{x}) \in \{\text{rbf}, \text{linear}\}$	
$\epsilon \in \{10^i, \text{ for } i \in [-5..0]\}$	$\lambda \in \{2^i, \text{ for } i \in [-10 : 2 : 10]\}^a$	

<sup>a</sup>[Start:steps:end]

capability is evaluated by applying them on a holdout test set.

## Results

To study the capabilities of the texture descriptor, a data set of 816 different cubic–orthorhombic textures is created. Details on construction and properties of that data set are given in Section “[Data Set](#).” In Section “[Training and cross-validation](#),” results from the training and validation process of the ML models are presented, followed by a study on the generalization properties of the trained models in Section “[Generalization on holdout test data set](#).”

## Data set

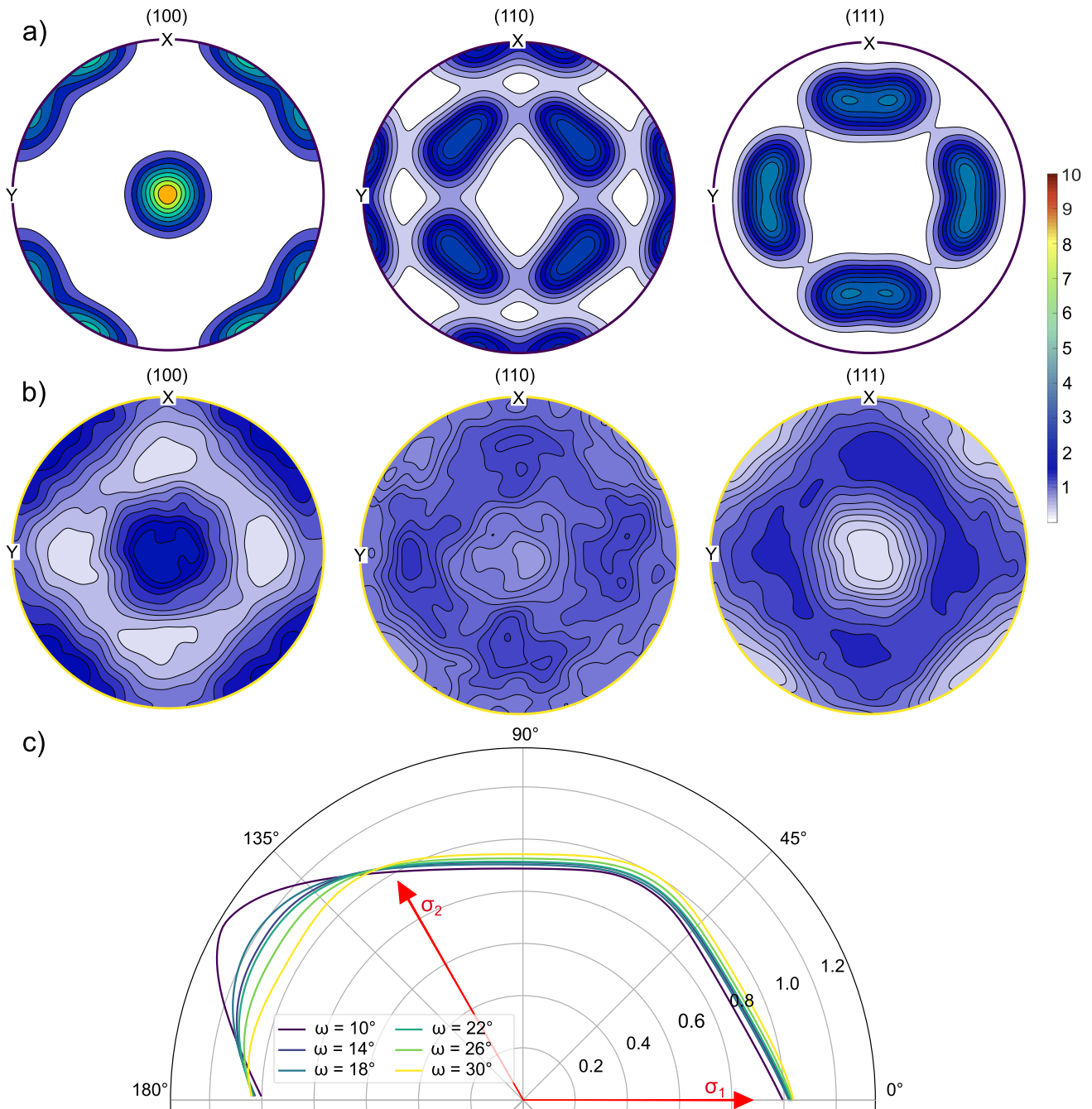
The data set constructed here consists of 816 different cubic–orthorhombic textures. Each texture is quantitatively defined by its ODF. To define the ODF, the kernel density estimation implemented in the MATLAB toolbox MTEX [32] is applied. Kernel density estimation approximates the ODF by superimposing a kernel function of specified half-width  $\omega$  over a given central orientation  $g$  in the space of orientations. The kernel function can be associated with a probability density function, centered at the orientation  $g$ . The half-width  $\omega$  then controls the spread of this distribution and its intensity, i.e., the height of the peak. For a small half-width, the distribution is narrow and only a small spread of orientations around the central orientation is sampled, yielding a highly anisotropic single crystal if  $\omega$  approximates zero. Vice versa, if  $\omega$  approaches  $90^\circ$ , the distribution becomes uniform and the resulting ODF corresponds to a random texture.

In this work, we choose 48 central orientations  $g_i, i \in [1 \dots 48]$  in the cubic–orthorhombic fundamental zone that are nearly equidistant to each other. For

each  $g_i$ , the de la Vallée Poussin kernel function is combined with 17 different half-widths  $\omega \in [10^\circ, 30^\circ]$ . This results in a number of 816 different unimodal, cubic–orthorhombic textures. To complete the training data set by the appropriate anisotropic coefficients, the yield function Yld2004-18p is parameterized for each texture using the method described in Section “[Parameterization of the yield function](#).”

Figure 5 gives an intuition of how changes in central orientation and half-width on the texture side affect the shape of the resulting yield surface on the property side. The pole figures correspond to a texture with central orientation  $g = (135^\circ, 7.5^\circ, 195^\circ)$ . In the top row, labeled (a), the texture is defined by the minimum kernel half-width  $\omega_{\min} = 10^\circ$ . This corresponds to the most narrow distribution sampled in this work. With a texture index of 6.69, the corresponding texture is severely anisotropic. The texture in the second row has the same central orientation  $g$ , but now the ODF is defined by the maximum kernel half-width  $\omega_{\max} = 30^\circ$  causing the largest spread in the orientation space. The resulting texture is much less anisotropic as shown by the pole figures in (b). The texture index is now 1.37, indicating an almost random texture.

The influence of the half-width  $\omega$  on the resulting yield loci can be seen in the bottom row of the figure. Similar to Fig. 3, the centro-symmetric yield loci are projected to the top half of the  $\pi$ -plane in the cylindrical principal deviatoric stress space. The yield surface corresponding to the texture in (a) is shown in dark purple and the yield surface of texture (b) in yellow. The half-width  $\omega$  is increased in steps of  $4^\circ$  between the two extremes (a) and (b). The figure clearly shows that the narrower the half-width of the kernel, the more anisotropic the resulting yield surface. This is plausible because the narrower the half-width, the more the behavior of the polycrystal approaches that of a single crystal, which is inherently anisotropic.



**Figure 5** Influence of the ODF kernel half-width  $\omega$  on the yield surface. Pole figures of a texture centered at  $g = (135^\circ, 7.5^\circ, 195^\circ)$  for minimum kernel half-width  $\omega_{\min} = 10^\circ$

(a) and maximum half-width  $\omega_{\max} = 30^\circ$  (b). Resulting yield loci for different kernel half-widths  $\in [\omega_{\min}, \omega_{\max}]$  are shown in (c).

### Training and validation set

The data set of 816 different textures is spread into a training and validation set and a holdout test set. The

training and validation set includes 652 randomly selected textures, representing 80% of the total data. The validation data sets are drawn from the training data set according to the cross-validation procedure described in Section “Training ML models.”

### Holdout test set

To evaluate the generalization properties of the trained models, a holdout test set of the remaining 164 textures is formed. Additionally to these, three prominent ideal texture components brass, copper and Goss, as well as three fiber textures  $\alpha$ ,  $\gamma$  and  $\theta$  are created. All of these textures are frequently observed in sheet metal forming and thus represent important test cases.

The ODFs of the ideal components follow the same logic as the unimodal textures described so far: a kernel of half-width  $\omega = 12^\circ$  is placed at the characteristic center orientation  $g$  of the ideal component. The key difference from the data described above, however, is that the central orientations of the ideal components differ from the 48 previously used. In that sense, this kind of test data requires a different kind of interpolation of the trained ML models.

The ODFs of the fiber components are even more different from the training validation data. Unlike the latter, they are not unimodal textures with a single peak, but have a skeleton line along which the maxima are distributed in the orientation space. Like the unimodal textures, the fiber textures are generated using MTEX. In Fig. 6, the skeleton lines of the  $\alpha$ ,  $\gamma$  and  $\theta$  fibers are plotted in the orientation space. The half-width  $\omega = 12^\circ$  was assigned to each fiber, describing a very narrow spread of orientations around the skeleton line. The pole figures for the  $\gamma$ -fiber are plotted as an example next to the skeleton plot. The different character of this strongly anisotropic texture with respect to the unimodal textures becomes immediately apparent when comparing the pole figures with those in Fig. 5. For the ML models that have been trained solely on

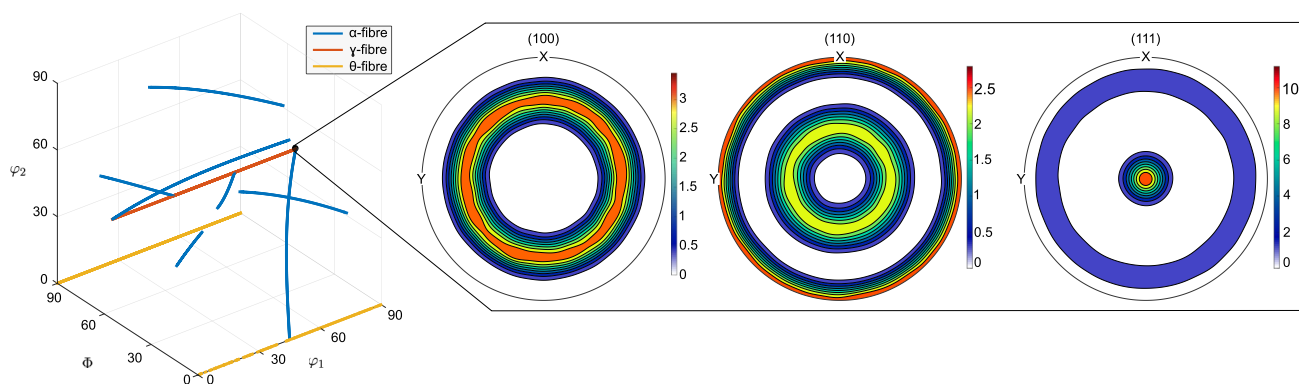
unimodal textures, predicting the anisotropic coefficients of fiber textures can be seen as an extrapolation task.

### Training and cross-validation

In this section, the results of training and validation process are presented. The three ML function classes introduced in Section “Training ML models” are compared based on the MSE score they achieve during cross-validation for various variants of the texture descriptor. The results should provide information about which function class and which variant of the descriptor is particularly suitable for determining the structure property relationship.

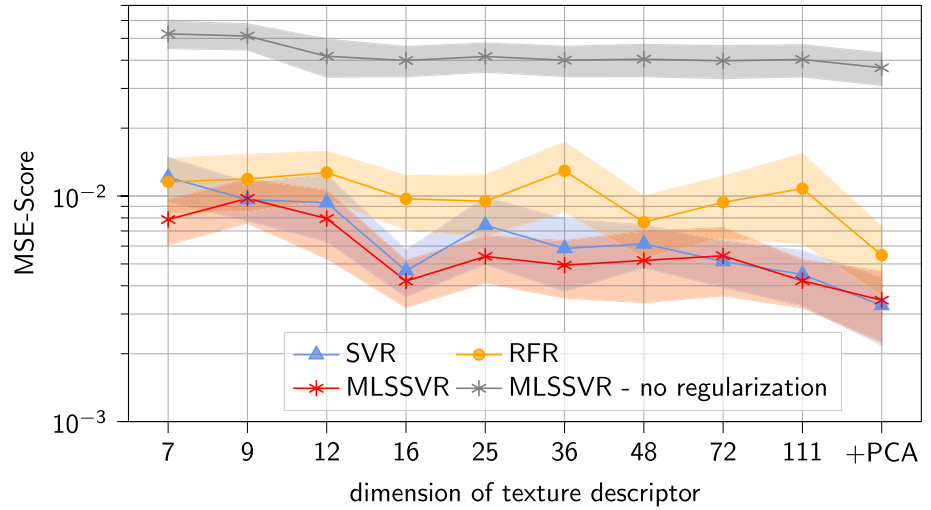
Figure 7 shows the results of the 10–7 nested CV. On the abscissa, the different variants of the texture descriptor are plotted in increasing dimension. The dimension of the respective descriptor corresponds to the number of grid nodes in the orientation space and arises from the resolution of the discretization as explained in Section “Texture descriptor.” On the ordinate the MSE is shown. The MSE is calculated according to Eq. (6) on the 10 different validation folds with  $n = 65$  data points, each. The colored lines in the figure correspond to the average MSE over the 10 validation folds, and the shadowed area around the lines marks the standard deviation.

In gray, the example of MLSSVR shows how the ML models behave when trained on a data set that is generated without the kNN regularization described in Section “Parameterization of the yield function.” With a mean validation MSE of around 0.04, the model is one order of magnitude less accurate than



**Figure 6** Skeleton lines of the three different fiber textures. Pole figures for  $\gamma$ -fiber texture with a  $12^\circ$  spread around the skeleton line are exemplary given on the right hand side of the figure.

**Figure 7** Mean-squared error (symbols and lines) and standard deviation (shaded regions) for the three different ML models during cross-validation.



the models trained on the regularized data set. If a regularized data set is used, the mean validation MSEs of SVR (blue) and RFR (orange) are  $1.2 \cdot 10^{-2}$  for the coarsest texture descriptor with only seven nodes. The MLSSVR (red) is already more accurate on this descriptor and achieves a smaller mean error with  $7.9 \cdot 10^{-3}$ . For the 16-d descriptor, all error curves reach a first relative minimum. The continuous function classes SVR and MLSSVR achieve an error of  $4.5 \cdot 10^{-3}$  and  $4.2 \cdot 10^{-3}$ , respectively, which is a 62% and 47% reduction from the 7-d description. With a mean validation MSE of  $1.1 \cdot 10^{-2}$ , the discontinuous RFR improves only slightly over the 7-d variant and is significantly less accurate than the other two function classes.

Further refinement of the grid resolution and the associated increase in the dimension of the texture descriptor up to 111 do not initially lead to any reduction in the error. The error of SVR and MLSSVR fluctuates slightly around  $5 \cdot 10^{-3}$ , with RFR fluctuating more sharply. When the dimension of the descriptor reaches 111, SVR and MLSSVR recover their relative minimum from the 16-d variant, whereas the RFR does not quite so. To check whether the stagnation of the error is related to the increasing complexity of the input space, a reduction using principal component analysis (PCA) is performed before training. Instead of the original 111 input dimensions, the  $m$  directions in which the variance of the data is largest are determined, and these are used for the texture description. The number of so-called principle components  $m$  enters the cross-validation as an additional hyperparameter and varies between 5 and 10 in our results,

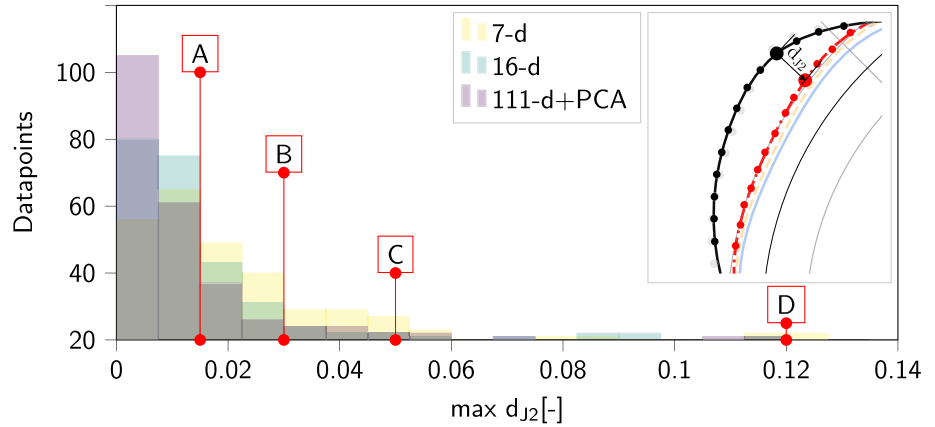
representing 76% and 91% of the cumulative variance of the data, respectively. The results show that the use of PCA in combination with the 111-d descriptor leads to an absolute minimum of the MSE with  $3.3 \cdot 10^{-3}$ ,  $3.5 \cdot 10^{-3}$  and  $5.5 \cdot 10^{-3}$  for SVR, MLSSVR and RFR, respectively.

### Generalization on holdout test data set

To assess the quality of descriptor and ML models, the generalization properties on the holdout test set are evaluated. Rather than comparing the predicted with the known coefficients in terms of MSE as was done previously, we now directly compare the resulting yield loci and their gradients.

In Fig. 8, the yield loci predicted by the MLSSVR are compared to the true reference yield loci for each of the 164 textures in the holdout test set. The histogram describes the distribution of the maximum relative difference between the reference and the predicted yield surface  $d_{J_2}$ . The concept behind this metric is illustrated in the small graphic in the upper right corner of Fig. 8. The graphic shows a section of the  $\pi$ -plane in the cylindrical principle stress space with the true reference yield surface in black and the MLSSVR prediction in red. In order to calculate  $d_{J_2}$ , firstly the  $J_2$ -equivalent stress is evaluated for the same 60 stress states at yield onset in the upper half of the  $\pi$ -plane (plotted as dots) for the reference and the predicted yield function, respectively. Then  $d_{J_2}$  is calculated as the difference between the two equivalent stresses, normalized by the true reference  $J_2$ -equivalent stress. The largest  $d_{J_2}$  among all of the 60 stress states is reported

**Figure 8** Distribution of the maximum relative distance  $d_{J_2}$  between true and MLSSVR-predicted yield surface for all texture in the holdout test set for the texture descriptors 7-d, 16-d and 111-d+PCA. The labels A, B, C, and D refer to the error levels used in following figures.

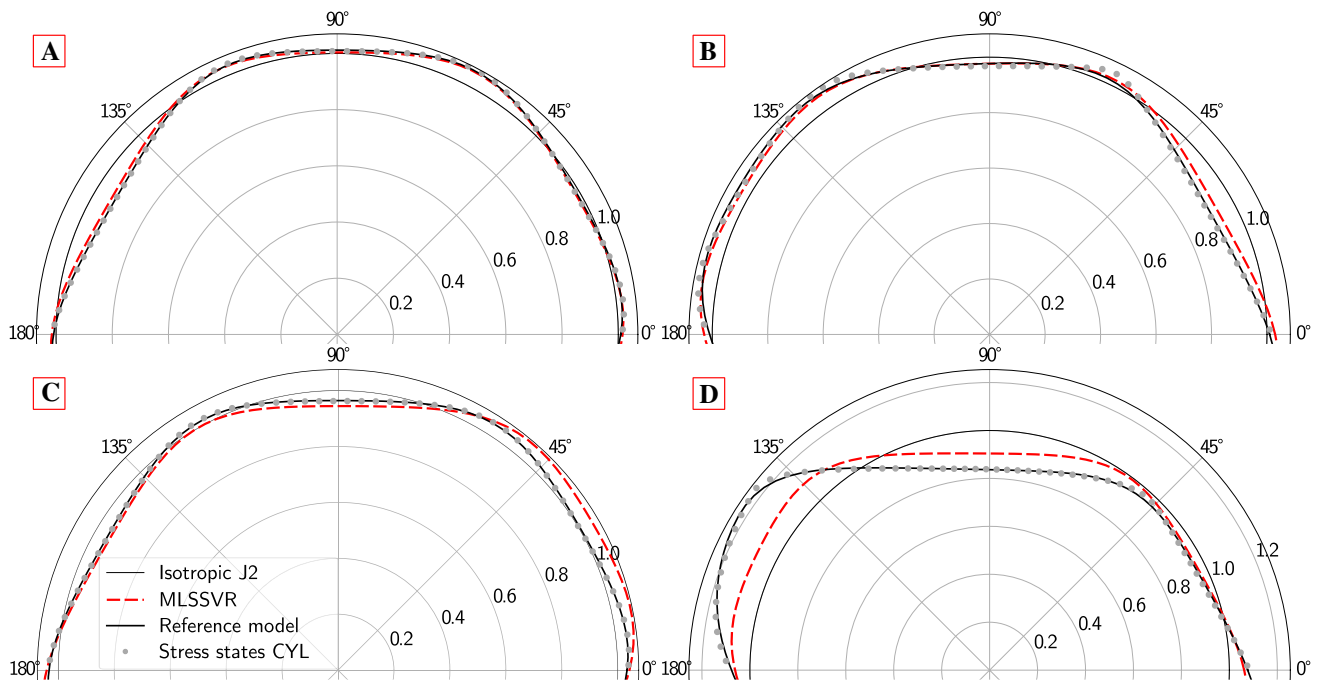


as  $\max d_{J_2}$  and added to the histogram. Repeating this calculation for all the 164 data points in the holdout test set results in the histogram displayed in Fig. 8. To get a sense of what this difference means qualitatively in terms of the shape of the entire yield surface, the yield loci at four characteristic error levels 1.5%, 3%, 5% and 12% (A–D) are shown as examples in Fig. 9

The colored bars in the histogram represent the  $d_{J_2}$ -error distributions of the MLSSVR trained on three different descriptor variants 7-d, 16-d and 111-d + PCA. For the coarsest 7-d descriptor in yellow, the error

follows a rather flat distribution with 50% of the test data below a maximum difference of 1.5% (point A). The second characteristic error level of 3%  $\max d_{J_2}$  is not exceeded by 80% of the test data (point B). Only in 6% of the test data, the observed deviation was larger than 5% w.r.t the reference yield surface.

If the 16-d descriptor is used for training of the MLSSVR, the error distribution shifts to the upper left direction of the histogram: The fraction of data points that do not exceed the first critical error level of 1.5% increases by 20% and now amounts 70%. Furthermore,



**Figure 9** Centro-symmetric yield loci for error levels A–D defined in Fig. 8 of the test data set plotted in upper half of  $\pi$ -plane. All stresses are normalized by the average yield stress of the corresponding microstructure.

the number of data points below a maximum difference of 3% increases to 90%, and the fraction of outliers above 5% error decreases to 4%.

The trend toward a narrower error distribution is continued when the 111-d descriptor combined with PCA is used for training. In this case, 77% of the yield loci in the test data set are predicted with a maximum deviation of less than 1.5%. The amount of data points below error level B does not change w.r.t the 16-d descriptor while the amount of outliers above 5% is slightly reduced to 3%.

In addition to the contours of the yield surface, the  $r$ -values of the predicted and reference yield functions are compared in the following. The  $r$ -value is a measure for plastic anisotropy widely used in sheet metal forming industry. For uniaxial tension, it is defined as the ratio of width strain to thickness strain or their rates. We follow the derivations in [44] and calculate the  $r$ -value  $r_\varphi$  by

$$\begin{aligned}
 r_\varphi &= \frac{\dot{\epsilon}_{2'2'}^p}{\dot{\epsilon}_{3'3'}^p} \Big|_{\sigma_\varphi} \\
 &= - \frac{\dot{\epsilon}_{2'2'}^p}{\dot{\epsilon}_{1'1'}^p + \dot{\epsilon}_{2'2'}^p} \Big|_{\sigma_\varphi} \\
 &= - \frac{\sin^2 \varphi \cdot \frac{\partial \bar{\sigma}}{\partial \sigma_{11}} - \frac{1}{2} \sin 2\varphi \cdot \frac{\partial \bar{\sigma}}{\partial \sigma_{12}} + \cos^2 \varphi \cdot \frac{\partial \bar{\sigma}}{\partial \sigma_{22}}}{\frac{\partial \bar{\sigma}}{\partial \sigma_{11}} + \frac{\partial \bar{\sigma}}{\partial \sigma_{22}}} \Big|_{\sigma_\varphi} .
 \end{aligned} \tag{7}$$

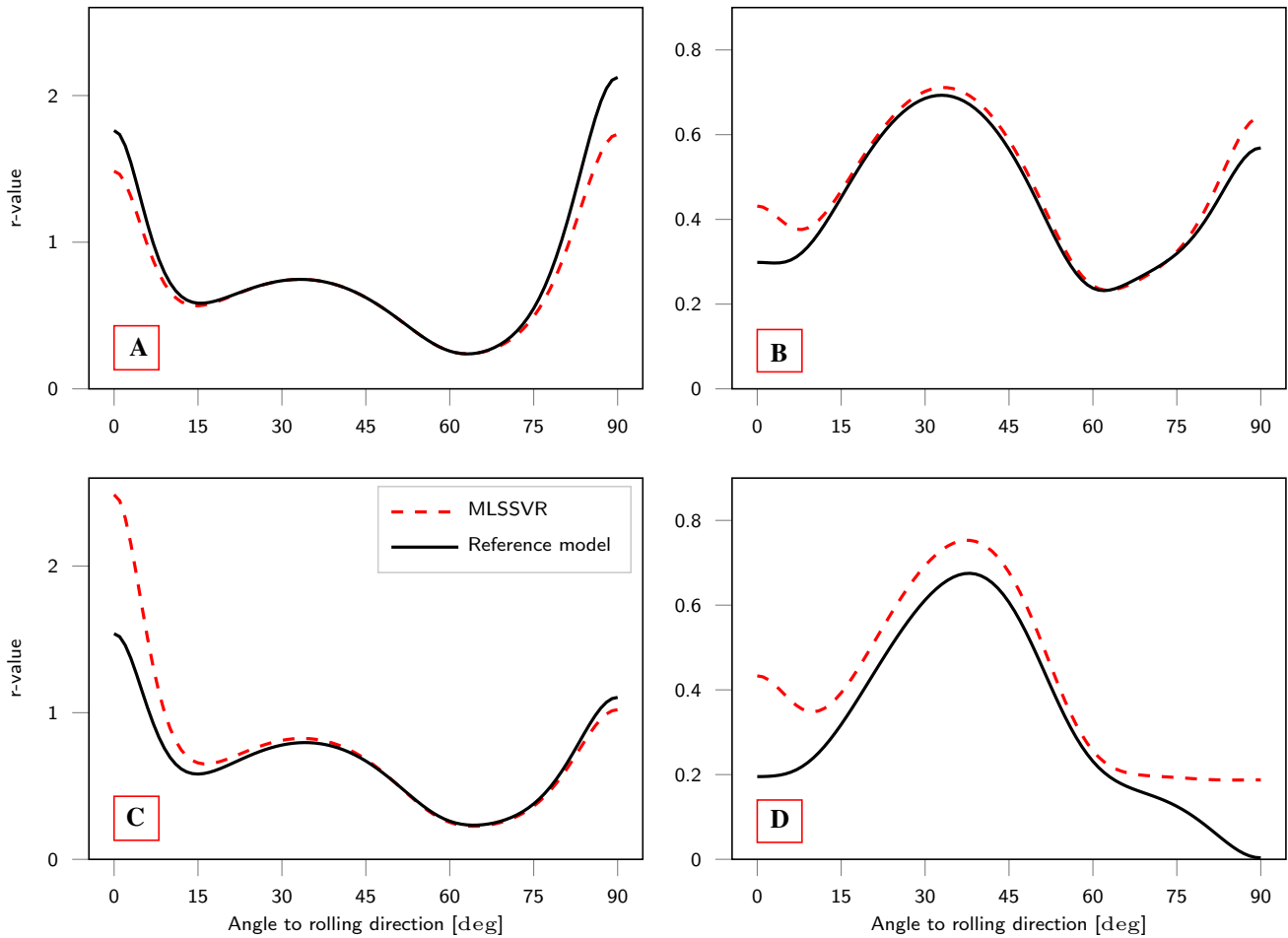
In Eq. (7), the tensor components with dashed indices are defined in the reference frame of uniaxial loading. This reference frame is related to the orthotropic material reference frame by a rotation of  $\varphi$  around the normal direction. Setting  $\varphi$  for example to 15°, 45° or 90° gives the three standard  $r$ -values in  $\varphi$  degree to rolling direction (direction 1 of material reference frame). The partial derivatives  $\frac{\partial \bar{\sigma}}{\partial \sigma_{ij}}$  are the gradients of the yield function (Eq. (1)), evaluated at the rotated uniaxial stress tensor at yield onset. As described in Section “Anisotropic yield function,” the anisotropic coefficients corresponding to the shear stress components have been set to zero, as only normal stresses are used for the parameterization. This would lead to vanishing gradients  $\frac{\partial \bar{\sigma}}{\partial \sigma_{12}}$  in the  $r$ -value calculation. In order to avoid this, the material is treated to be isotropic with respect to these directions and the anisotropic

coefficients in reference and predicted model where enforced to be 1, consequently.

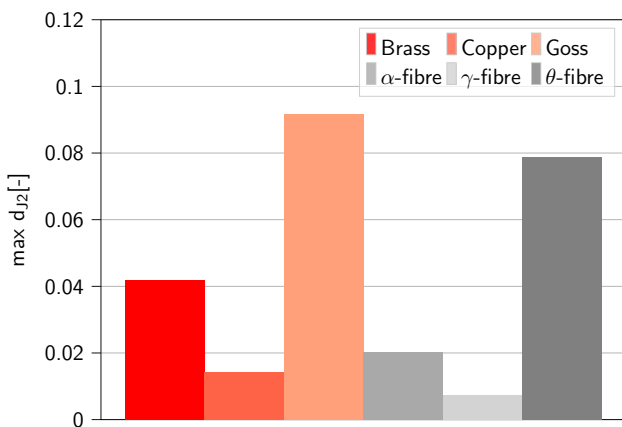
Figure 10 shows the  $r$ -values in case of uniaxial tension with tensile axis oriented by  $\varphi$  between 0° and 90° to rolling direction for four different textures. The chosen textures are the same as in Fig. 8 and correspond to the characteristic error levels A–D. The  $r$ -values for the MLSSVR-predicted yield functions are shown in red and the reference models in black. For the smallest error level A, the  $r$ -values between 10° and 75° are in almost perfect agreement with the reference model. With increasing  $d_{J_2}$  error, also the deviation between predicted and reference  $r$ -values increases. For error level B in the second sub-figure, in addition to the inaccuracies at the edges, small deviations in the middle part of the  $r$ -value curve appear. For the most inaccurately predicted yield function in the test data set in the last sub-figure, significant quantitative deviations of the  $r$ -values are observed. However, the qualitative course of the reference curve can be approximately reproduced.

In a final consideration, the generalization properties of the trained MLSSVR are evaluated on the prominent texture components brass, copper and Goss, as well as of three fiber textures  $\alpha$ ,  $\gamma$  and  $\theta$  introduced in Section “Data set.” The former, like the textures in the holdout test data set, address the interpolation properties of MLSSVR, since they are also unimodal textures but with different central orientation. The application of the trained MLSSVR to fiber textures is an extrapolation task, since the training data set does not contain any fiber textures but only unimodal ones.

The maximum  $d_{J_2}$ -error for the six textures is shown in Fig. 11. The red-colored bars correspond to the unimodal textures, while the gray bars represent the fiber textures. All yield functions are predicted in the limits discussed above for the holdout data set. For the relatively isotropic copper texture, the predicted yield function is quite accurate with a maximum deviation of 1.4%. For the more anisotropic brass texture the deviation is 4.2%. The least accurate prediction is made by MLSSVR for the strongly anisotropic Goss texture with a maximum deviation of 9%. In general, however, the shape can also be reproduced well here. It is noteworthy that even the fiber textures can be predicted with good accuracy, although the ML model has only been trained on unimodal textures.



**Figure 10**  $r$ -values for error levels A–D defined in Fig. 8 of the test data set.



**Figure 11** Maximum relative distance  $d_{J_2}$  between true and MLSSVR-predicted yield surface for prominent texture components (red) and fibers (gray).

## Discussion

In the following, the results from the previous section will be used to assess whether and in which variant the descriptor presented here is suitable for determining the  $s$ - $p$  relationship between texture and anisotropic yield function.

Looking at the CV results, the first noticeable aspect is the importance of kNN regularization in data generation. If regularization is not applied, the MSE in the CV is consistently an order of magnitude higher than for a regularized data set, resulting in very inaccurate predictions of the yield function. With the help of the regularization approach, it is possible to use the non-uniqueness of the anisotropic coefficients, which is often described as a disadvantage of phenomenological models in the literature [13, 17], to generate a distribution of the coefficients that is relatively continuous. The observations confirm the considerations in [22]



that regularization partially resolves discontinuities in the data set for the case of a higher-dimensional texture description.

If the results of the CV are considered from the point of view of model selection, the continuous models SVR and MLSSVR are found to be more appropriate functional classes than the discontinuous RFR, which is a common choice for solving regression problems. The results further suggest that the description of the intercorrelation of the anisotropic coefficients, which the MLSSVR does through its hierarchical Bayes approach but which the SVR completely ignores, has only little effect on the accuracy of the results. However, earlier studies on simplified, one-dimensional texture descriptions suggest that modeling intercorrelation is advantageous in the case of a sparser data set, and MLSSVR is significantly more accurate in such situations than single-target function classes such as SVR, which disregard correlation [22].

Looking at the evolution of the MSE during CV over the different descriptor variants, it is noticeable that a minimum number of 16 nodes in the orientation space leads to a first relative error minimum. A coarser mesh with 7, 9 or 12 nodes is not able to capture the decisive features of the texture required to explain the anisotropic coefficients. Furthermore, it can be observed that with further mesh refinement of the orientation space, the error for all three function classes stagnates and even slightly increases up to the finest discretization with 111 nodes. A possible explanation for this would be that although the finer discretization adds additional nodes to the descriptor, these do not represent any additional information gain in terms of the  $s$ - $p$  relationship sought between texture and yield function. This thesis is contradicted by the fact that when PCA is applied to the 111-d descriptor, the error decreases compared to the 16-d variant. This suggests that the finer mesh with 111 nodes does add relevant information to the descriptor and makes a second explanation more likely, which is referred to as the bias-variance trade-off in statistical learning [45]. Adding more nodes increases the dimension of the input space for the ML models and thus their parameter spaces. In these spaces, the correct parameters have to be found during training in order to approximate the sought unknown objective function as accurate as possible. If the space is larger due to an information-rich, higher-dimensional descriptor, it is possible to find a better approximation (lower bias). However, the search for the appropriate parameters is much more

difficult than in a low-dimensional space (higher variance). Since the number of data points available for the search is the same in both spaces, it is possible that the simple approximation in 16-d space is more accurate than the high-variance approximation in 111-d space. If PCA is used to systematically reduce the 111-d input space before training, the information content of a high-dimensional descriptor can still be used during training without having to pay the price of the high-dimensional parameter space. For this reason, the MSE values for all function classes drop below the relative minimum of the 16-d descriptor.

Summarizing the CV results, the combination of 111-d descriptor with PCA using MLSSVR achieves the best results to determine the sought  $s$ - $p$  relationship between texture and anisotropic yield function. Comparing the number of required nodes in the orientation space with similar approaches, such as the histogram approach by Dornheim et al. [46], it turns out to be slightly lower. For the prediction of elastic constants, the authors achieve good results with 512 bins in the cubic fundamental zone. Generalizing the 111-d descriptor presented here for the cubic-orthorhombic fundamental zone to triclinic specimen symmetry yields a number of 435 nodes.

To study the generalization properties of the trained MLSSVR, it is applied to the holdout test data sets. The first data set analyzed contains the 164 unimodal textures that were separated from the overall data set before training. The error distribution of the maximum deviation in yield onset, shown as a histogram in Fig. 8, emphasizes the findings from the CV with respect to the optimal descriptor variant. While the distribution is rather flat, it becomes narrower in case the 111-d+PCA variant is used during training. In that case, 90% of the yield functions in the test set are predicted with a maximum deviation in yield onset below 3%. An outlier fraction of 3% deviates by more than 5%. Also qualitatively, the predicted yield loci agree well with the reference ones, as shown in Fig. 9.

Comparing the  $r$ -values between predicted and reference yield function (Fig. 10), a slightly larger discrepancy is observed than for the plain shape. One explanation for this could be the high sensitivity of the  $r$ -values with respect to the stress gradients as already pointed out by Nascimento et al. [13] and Zhang et al. [35]. Overall, however, the  $r$ -value profiles can be predicted very well for a large proportion of the test data, especially if the angle to the rolling direction  $\varphi$  is between  $10^\circ$  and  $75^\circ$ . Close to rolling and transverse direction ( $\varphi = 0^\circ$  and

$90^\circ$ ), the  $r$ -values systematically deviate from the reference values. This is partially explained by the ratio of shear to normal stresses and the enforced condition on the shear coefficients described in Section “[Generalization on holdout test data set.](#)” For  $\varphi = 0^\circ$  and  $90^\circ$ , the shear stress is equal to zero and the fraction vanishes from the numerator in Eq. (7). With increasing angle  $\varphi$ , the fraction of the shear component in the Cauchy stress tensor increases up to a maximum at  $\varphi = 45^\circ$  and then decreases again until it gets zero as  $\varphi$  approaches  $90^\circ$ . As the shear coefficients of reference and ML yield function are enforced to be the same, increasing the fraction of the stress component weighs the error in a beneficial way. Nevertheless, as shown for error level D, the anisotropic coefficients of the normal stress components still play a role even in case of maximum shear fractions and can lead to substantial deviations if predicted inaccurately.

The second data set consists of six characteristic rolling textures: the three unimodal textures Goss, copper and brass, and the three fiber textures  $\alpha$ ,  $\gamma$  and  $\theta$ . The MLSSVR predicts the yield loci of the unimodal textures with a similar accuracy as the textures in the first data set, emphasizing the good interpolation properties of the model for even more distinct textures. However, it is noticeable that also the fiber textures are predicted very accurately, although only unimodal textures were used during the training process. This suggests that the proposed descriptor could also be suitable for fiber textures and that the MLSSVR even extrapolates to this texture type to some extent.

Compared to the results of the related work by Montes de Oca Zapiain et al. [25], the descriptor presented here is shown to be an accurate and efficient alternative to their GSH-based approach. The predictions of the advanced Yld2004-18p yield criterion show a similar accuracy as the neuronal network predictions of the simpler Hill model studied in [25]. Instead of the parameter-intensive neural network, ML models were used here, which are also suitable for smaller data sets. This will play a more important role in particular when more full-field CP computations have to be performed to generate training data in the full stress space.

## Conclusion

In this work, we present a new approach for the description of crystallographic texture in data-driven constitutive modeling of polycrystalline metals. The

description is based on a discretization of the orientation space by an equidistant grid of user-specific resolution. We show that the approach is suitable to determine the structure–property (s–p) relationship between cubic–orthorhombic textures and the material parameters of the anisotropic Yld2004-18p yield function [2]. For this purpose, we train three different supervised machine learning (ML) methods on a data set of 816 textures that have been described by the novel descriptors with different grid resolutions.

We find that a grid with 16 nodes already captures the essential texture information needed to determine the s–p relationship. Refining the grid to 111 nodes and successively applying dimension reduction by principal component analysis (PCA) to 5 or 10 components, further improves the accuracy of the models. Independent of the grid resolution, the use of the k-nearest-neighbor (kNN) regularization introduced here, proofed to be a prerequisite for successful training. By requiring that neighboring textures should have similar coefficients, the well-known non-uniqueness of the Yld2004-18p material parameters, where different parameter sets describe the same yield function [6], could be bypassed.

The trained models show good generalization properties on a holdout test data set and can accurately predict the yield surface and also the  $r$ -values for most textures in the test set. Final investigations suggest that interpolation to other unimodal textures is possible as well as extrapolation to fiber textures. The latter, however, requires further evaluation.

The new descriptor allows the incorporation of meaningful microstructural degrees of freedom into data-driven constitutive modeling of anisotropic plasticity, providing a pathway to model texture evolution and to train more general material models. In upcoming work, we plan to evaluate the suitability of the descriptor for fiber textures and to extend the method to the entire stress space. Since this extension requires more computationally intensive crystal plasticity simulations, active learning methods could be of particular interest, as shown for example in [47].

## Acknowledgements

Not applicable. Only authors mentioned in the author list contributed to this paper. There is no funding to acknowledge.

### Author contributions

JS was involved in conceptualization, methodology, software, formal analysis, investigation, writing—original draft and visualization. AH was responsible for conceptualization, methodology, writing—review and editing, supervision, project administration and funding acquisition.

### Funding

Open Access funding enabled and organized by Projekt DEAL.

### Data availability

Data and code will be made available on request.

### Declarations

**Conflict of 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.

**Ethical approval** Not applicable.

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

### Appendix A: CP Model

As explained in Section “Parameterization of the yield function,” the hybrid crystal plasticity scheme employed in this work requires two full-field CPFEM calculations on an RVE for each texture. Details on the crystal plasticity constitutive model as well as on the material parameters are given in the following.

According to Lee and Liu [48], the total deformation gradient  $F$  is multiplicatively decomposed into an elastic and a plastic part:

$$F = F_e F_p. \tag{8}$$

The plastic part  $F_p$  maps the reference configuration to the stress-free intermediate configuration. All of the following calculation steps required to find the stress at the next increment are performed in this intermediate configuration. From there,  $F_e$  maps to the current configuration taking lattice rotations and elastic stretches into account. Following Roters et al. [49], the multiplicative decomposition of the deformation gradient yields an additive decomposition of the velocity gradient  $L$

$$\begin{aligned} L &= \dot{F}F^{-1} = \dot{F}_e F_e^{-1} \\ &\quad + F_e \dot{F}_p F_p^{-1} F_e^{-1} \\ &= L_e F_e L_p F_e^{-1}. \end{aligned} \tag{9}$$

The evolution of plastic flow is then given by

$$\dot{F} = L_p F_p^{-1}. \tag{10}$$

The formulation of the plastic velocity gradient  $L_p$  requires an additional kinematic equation as well as a constitutive model that relates stimulus and response on grain level. Assuming that dislocation slip is the sole cause for plastic flow, the kinematic equation is given by

$$L_p = \sum_{\alpha=1}^{N_s} \dot{\gamma}^{\alpha} \tilde{m}^{\alpha} \otimes \tilde{n}^{\alpha} = \sum_{\alpha=1}^{N_s} \dot{\gamma}^{\alpha} \tilde{M}^{\alpha}. \tag{11}$$

In Eq. (11), the Schmid tensor in the intermediate configuration  $\tilde{M}$  for slip system  $\alpha$  is composed of the slip plane normal  $\tilde{n}^{\alpha}$  and the slip direction  $\tilde{m}^{\alpha}$ . For the fcc structures analyzed in this work, four unique slip planes of type {111} with three slip directions of type <110> have to be considered.

The rate-dependent phenomenological constitutive model proposed by Rice [50] and Hutchinson and Hill [51] is employed to relate the slip rates  $\dot{\gamma}^\alpha$  on the slip system  $\alpha$  to the respective resolved shear stress on that system  $\tau^\alpha$

$$\dot{\gamma}^\alpha = \dot{\gamma}_0 \left| \frac{\tau^\alpha}{\tau_c^\alpha} \right|^m \operatorname{sgn}(\tau^\alpha), \quad (12)$$

where  $\dot{\gamma}_0$  and  $m$  are material parameters for the reference shear rate and the strain rate sensitivity exponent, respectively;  $\tau_c^\alpha$  is the slip resistance or critical resolved shear stress on slip system  $\alpha$ . The evolution of this material parameter due to hardening effects is captured by the rate equation

$$\dot{\tau}_c^\alpha = h_{\alpha\beta} \left| \dot{\gamma}^\beta \right|, \quad (13)$$

where  $h_{\alpha\beta}$  expresses the hardening matrix,

$$h_{\alpha\beta} = q_{\alpha\beta} \left[ h_0 \left( 1 - \frac{\tau_c^\beta}{\tau_s} \right)^a \right], \quad (14)$$

which empirically describes the interaction between different slip systems in a grain [49]. The material parameters for slip hardening  $h_0$ ,  $a$  and  $\tau_s$  are assumed to be identical for all slip systems in the fcc crystal. Latent hardening effects are captured by  $q_{\alpha\beta}$ .

All material parameters that appear in the above equations (plus the elastic constants) are summarized in Table 3. Their values are taken from the work of Anand [52] on the single-crystal elasto-viscoplastic behavior of fcc copper.

**Table 3** Material parameters of crystal plasticity constitutive model

Parameter	Value	Unit
$C_{11}$	170	GPa
$C_{12}$	124	GPa
$C_{44}$	75	GPa
$\dot{\gamma}_0$	0.001	s <sup>-1</sup>
$m$	83	–
$a$	2.5	–
$h_0$	250	MPa
$\tau_s$	148	MPa
$q_{\alpha\beta}$	$\alpha, \beta$ Coplanar: 1.0 Non-coplanar: 1.4	– –

## References

- [1] Bock FE, Aydin RC, Cyron CJ, Huber N, Kalidindi SR, Klusemann B (2019) A review of the application of machine learning and data mining approaches in continuum materials mechanics. *Front Mater* 6:110–23. <https://doi.org/10.3389/fmats.2019.00110>
- [2] Barlat F, Aretz H, Yoon JW, Karabin ME, Brem JC, Dick RE (2005) Linear transformation-based anisotropic yield functions. *Int J Plast* 21(5):1009–1039. <https://doi.org/10.1016/j.ijplas.2004.06.004>
- [3] Barlat F, Lege DJ, Brem JC (1991) A six-component yield function for anisotropic materials. *Int J Plast* 7(7):693–712. [https://doi.org/10.1016/0749-6419\(91\)90052-Z](https://doi.org/10.1016/0749-6419(91)90052-Z)
- [4] Aretz H, Aegerter J, Engler O (2010) Analysis of earing in deep drawn cups. *AIP Conf Proc* 1252(1):417–424. <https://doi.org/10.1063/1.3457585>
- [5] Fuhg JN, van Wees L, Obstalecki M, Shade P, Bouklas N, Kasemer M (2022) Machine-learning convex and texture-dependent macroscopic yield from crystal plasticity simulations. *Materialia* 23:101446–23. <https://doi.org/10.1016/j.mtla.2022.101446>
- [6] van den Boogaard T, Havinga J, Belin A, Barlat F (2016) Parameter reduction for the Yld 2004-18p yield criterion. *Int J Mater Form* 9(2):175–178. <https://doi.org/10.1007/s12289-015-1221-3>
- [7] Kirchdoerfer T, Ortiz M (2016) Data-driven computational mechanics. *Comput Methods Appl Mech Eng* 304:81–101. <https://doi.org/10.1016/j.cma.2016.02.001>
- [8] Eggersmann R, Kirchdoerfer T, Reese S, Stainier L, Ortiz M (2019) Model-free data-driven inelasticity. *Comput Methods Appl Mech Eng* 350:81–99 (2019) [arXiv:1808.10859](https://arxiv.org/abs/1808.10859) [physics]. <https://doi.org/10.1016/j.cma.2019.02.016>
- [9] Chinesta F, Ladeveze P, Ibanez R, Aguado JV, Abisset-Chavanne E, Cueto E (2017) Data-driven computational plasticity. *Proc Eng* 207:209–214. <https://doi.org/10.1016/j.proeng.2017.10.763>
- [10] Lefik M, Schrefler BA (2003) Artificial neural network as an incremental non-linear constitutive model for a finite element code. *Comput Methods Appl Mech Eng* 192(28):3265–3283. [https://doi.org/10.1016/S0045-7825\(03\)00350-5](https://doi.org/10.1016/S0045-7825(03)00350-5)
- [11] Huang D, Fuhg JN, Weißenfels C, Wriggers P (2020) A machine learning based plasticity model using proper orthogonal decomposition. *Comput Methods Appl Mech Eng* 365:113008–26. <https://doi.org/10.1016/j.cma.2020.113008>
- [12] Vlassis NN, Sun W (2021) Sobolev training of thermo-dynamic-informed neural networks for interpretable elasto-plasticity models with level set hardening. *Comput*

- Methods Appl Mech Eng 377:113695–37. <https://doi.org/10.1016/j.cma.2021.113695>
- [13] Nascimento A, Roongta S, Diehl M, Beyerlein IJ (2023) A machine learning model to predict yield surfaces from crystal plasticity simulations. *Int J Plast* 161:103507–20. <https://doi.org/10.1016/j.ijplas.2022.103507>
- [14] Hartmaier A (2020) Data-oriented constitutive modeling of plasticity in metals. *Materials* 13(7):1600–18. <https://doi.org/10.3390/ma13071600>
- [15] Shoghi R, Hartmaier A (2022) Optimal data-generation strategy for machine learning yield functions in anisotropic plasticity. *Front Mater* 9:868248. <https://doi.org/10.3389/fmats.2022.868248>
- [16] Rocha IBCM, Kerfriden P, van der Meer FP (2021) On-the-fly construction of surrogate constitutive models for concurrent multiscale mechanical analysis through probabilistic machine learning. *J Comput Phys X* 9:100083. <https://doi.org/10.1016/j.jcpx.2020.100083>
- [17] Fuhg JN, Bouklas N (2022) On physics-informed data-driven isotropic and anisotropic constitutive models through probabilistic machine learning and space-filling sampling. *Comput Methods Appl Mech Eng* 394:114915–26. <https://doi.org/10.1016/j.cma.2022.114915>
- [18] Raabe D, Roters F, Barlat F, Chen L-Q (2006) Continuum scale simulation of engineering materials: fundamentals—microstructures—process applications. Wiley, Weinheim
- [19] Ali U, Muhammad W, Brahme A, Skiba O, Inal K (2019) Application of artificial neural networks in micromechanics for polycrystalline metals. *Int J Plast* 120:205–219. <https://doi.org/10.1016/j.ijplas.2019.05.001>
- [20] Luecke K, Pospiech J, Jura J, Hirsch J (1986) On the Presentation of Orientation Distribution Functions by Model Functions. *Z Metall* 77:312–321
- [21] Pospiech J, Lücke K, Sztwiertnia K (1993) Orientation distribution and orientation correlation functions for description of microstructures. *Acta Metall Mater* 41(1):305–321. [https://doi.org/10.1016/0956-7151\(93\)90361-U](https://doi.org/10.1016/0956-7151(93)90361-U)
- [22] Schmidt J, Biswas A, Vajragupta N, Hartmaier A (2022) Data-oriented description of texture-dependent anisotropic material behavior. *Modell Simul Mater Sci Eng* 30:065001–28. <https://doi.org/10.1088/1361-651X/ac7739>
- [23] Bunge H-J (2013) Texture analysis in materials science: mathematical methods. Elsevier, Berlin
- [24] Wenk H-R, Houtte PV (2004) Texture and anisotropy. *Rep Prog Phys* 67(8):1367–1428. <https://doi.org/10.1088/0034-4885/67/8/R02>
- [25] de Oca-Zapiain DM, Lim H, Park T, Pourboghrat F (2022) Predicting plastic anisotropy using crystal plasticity and Bayesian neural network surrogate models. *Mater Sci Eng A* 833:142472. <https://doi.org/10.1016/j.msea.2021.142472>
- [26] Hill R (1948) A theory of the yielding and plastic flow of anisotropic metals. *Proc R Soc Lond Ser A Math Phys Sci*. <https://doi.org/10.1098/rspa.1948.0045>
- [27] Helming K (1995) Texturapproximation durch modellkomponenten. PhD Thesis, Technische Universität Clausthal, Clausthal
- [28] Helming K (1997) A nearly equal distant grid of orientations for quantitative texture analysis. *Texture Stress Microstruct* 28(3–4):219–230. <https://doi.org/10.1155/TSM.28.219>
- [29] Górski KM, Hivon E, Banday AJ, Wandelt BD, Hansen FK, Reinecke M, Bartelmann M (2005) HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *Astrophys J* 622(2):759–771. <https://doi.org/10.1086/427976>
- [30] Yershova A, Jain S, LaValle SM, Mitchell JC (2010) Generating uniform incremental grids on  $SO(3)$  using the hopf fibration. *Int J Robot Res* 29(7):801–812. <https://doi.org/10.1177/0278364909352700>
- [31] Roşca D, Morawiec A, Graef MD (2014) A new method of constructing a grid in the space of 3d rotations and its applications to texture analysis. *Modell Simul Mater Sci Eng* 22(7):075013. <https://doi.org/10.1088/0965-0393/22/7/075013>
- [32] Bachmann F, Hielscher R, Schaeben H (2010) Texture analysis with MTEX—free and open source software toolbox. *Solid State Phenom* 160:63–68. <https://doi.org/10.4028/www.scientific.net/SSP.160.63>
- [33] Biswas A, Vajragupta N, Hielscher R, Hartmaier A (2020) Optimized reconstruction of the crystallographic orientation density function based on a reduced set of orientations. *J Appl Crystallogr* 53(1):178–187. <https://doi.org/10.1107/S1600576719017138>
- [34] Biswas A, Kalidindi SR, Hartmaier A (2022) A hybrid approach for the efficient computation of polycrystalline yield loci with the accuracy of the crystal plasticity finite element method. *Modell Simul Mater Sci Eng* 30(2):025015–23. <https://doi.org/10.1088/1361-651X/ac4a24>
- [35] Zhang H, Diehl M, Roters F, Raabe D (2016) A virtual laboratory using high resolution crystal plasticity simulations to determine the initial yield surface for sheet metal forming operations. *Int J Plast* 80:111–138. <https://doi.org/10.1016/j.ijplas.2016.01.002>
- [36] Van Houtte P, Cauwenberg G, Aernoudt E (1987) Analysis of the earing behaviour of aluminium 3004 alloys by means of a single model based on yield loci calculated from orientation distribution functions. *Mater Sci Eng* 95:115–124. [https://doi.org/10.1016/0025-5416\(87\)90503-9](https://doi.org/10.1016/0025-5416(87)90503-9)

- [37] Boeff M (December 2016) Micromechanical modelling of fatigue crack initiation and growth. PhD Thesis, Ruhr-Universität Bochum, Bochum
- [38] Jones E, Oliphant T, Peterson P (2001) SciPy: Open Source Scientific Tools for Python. <https://docs.scipy.org/doc/scipy/reference/optimize.minimize-trustconstr.html> (2001)
- [39] Yuan, Y.-X.: A review of trust region algorithms for optimization. In: ICM99: Proceedings of the fourth international congress on industrial and applied mathematics, pp 271–282 (1999)
- [40] Vapnik V (2000) The nature of statistical learning theory, 2nd edn. In: Information Science and Statistics. Springer, New York. <https://doi.org/10.1007/978-1-4757-3264-1>
- [41] Xu S, An X, Qiao X, Zhu L, Li L (2013) Multi-output least-squares support vector regression machines. *Pattern Recogn Lett* 34(9):1078–1084. <https://doi.org/10.1016/j.patrec.2013.01.015>
- [42] De'ath G (2001) Multivariate regression trees: a new technique for modeling species-environment relationships. *Ecology* 83:1105–1117. <https://doi.org/10.2307/3071917>
- [43] Borchani H, Varando G, Bielza C, Larrañaga P (2015) A survey on multi-output regression. *WIREs Data Min Knowl Discov* 5(5):216–233. <https://doi.org/10.1002/widm.1157>
- [44] Aretz H, Barlat F (2013) New convex yield functions for orthotropic metal plasticity. *Int J Non-Linear Mech* 51:97–111. <https://doi.org/10.1016/j.ijnonlinmec.2012.12.007>
- [45] James G, Witten D, Hastie T, Tibshirani R (2021) An introduction to statistical learning: with applications in R. Springer, New York
- [46] Dornheim J, Morand L, Zeitvogel S, Iraki T, Link N, Helm D (2022) Deep reinforcement learning methods for structure-guided processing path optimization. *J Intell Manuf* 33(1):333–352. <https://doi.org/10.1007/s10845-021-01805-z>
- [47] Morand L, Link N, Iraki T, Dornheim J, Helm D (2022) Efficient exploration of microstructure-property spaces via active learning. *Front Mater* 8:824441. <https://doi.org/10.3389/fmats.2021.824441>
- [48] Lee EH, Liu DT (1967) Finite-strain elastic plastic theory with application to plane-wave analysis. *J Appl Phys* 38(1):19–27. <https://doi.org/10.1063/1.1708953>
- [49] Roters F, Eisenlohr P, Bieler TR, Raabe D (2011) Crystal plasticity finite element methods: in materials science and engineering. Wiley
- [50] Rice JR (1971) Inelastic constitutive relations for solids: an internal-variable theory and its application to metal plasticity. *J Mech Phys Solids* 19(6):433–455. [https://doi.org/10.1016/0022-5096\(71\)90010-X](https://doi.org/10.1016/0022-5096(71)90010-X)
- [51] Hutchinson JW, Hill R (1976) Bounds and self-consistent estimates for creep of polycrystalline materials. *Proc R Soc Lond A Math Phys Sci* 348(1652):101–127 (1976). <https://doi.org/10.1098/rspa.1976.0027>
- [52] Anand L (2004) Single-crystal elasto-viscoplasticity: application to texture evolution in polycrystalline metals at large strains. *Comput Methods Appl Mech Eng* 193(48):5359–5383. <https://doi.org/10.1016/j.cma.2003.12.068>

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.