Machine learning for structure-guided materials and process design

Lukas Morand [a,*], Tarek Iraki [b,c], Johannes Dornheim [c,d], Stefan Sandfeld [b], Norbert Link [c], Dirk Helm [a]

[a] Fraunhofer Institute for Mechanics of Materials IWM, Freiburg, Germany
[b] Institute for Advanced Simulations - Materials Data Science and Informatics (IAS-9), Forschungszentrum Jülich GmbH, 52425, Jülich, Germany
[c] Intelligent Systems Research Group ISRG, University of Applied Sciences, Karlsruhe, Germany
[d] Institute for Applied Materials - Computational Materials Sciences IAM-CMS, Karlsruhe Institute of Technology, Karlsruhe, Germany

ARTICLE INFO

ABSTRACT

In recent years, there has been a growing interest in accelerated materials innovation in the context of the process-structure-property chain. In this regard, it is essential to take into account manufacturing processes and tailor materials design approaches to support downstream process design approaches. As a major step into this direction, we present a holistic and generic optimization approach that covers the entire process-structure-property chain in materials engineering. Our approach specifically employs machine learning to address two critical identification problems: a materials design problem, which involves identifying near-optimal material microstructures that exhibit desired properties, and a process design problem that is to find an optimal processing path to manufacture these microstructures. Both identification problems are typically ill-posed, which presents a significant challenge for solution approaches. However, the non-unique nature of these problems offers an important advantage for processing: By having several target microstructures that perform similarly well, processes can be efficiently guided towards manufacturing the best reachable microstructure. The functionality of the approach is demonstrated at manufacturing crystallographic textures with desired properties in a simulated metal forming process.

1. Introduction

1.1. Motivation

Accelerated materials innovation has become a core research field in integrated computational materials engineering (ICME) and is pushed forward strongly in materials science and engineering (cf. materials genome initiative [1], European advanced materials initiative [2]). Essentially, the properties of such new sustainable, resilient, and high-performance materials depend on the microstructure of the material, which, in turn, depends on the manufacturing process. Consequently, designing new materials without taking into account the manufacturing process does not add significant value [3]. For an application in industry, groundbreaking technologies for modeling and optimization that cover the entire process-structure-property chain are required.

The process-structure-property chain, as depicted in Fig. 1, was originally introduced in [4] and describes fundamental relations in materials processing. A basic characteristic of this chain is its modularity: Each individual link represents a specific identification problem, which includes identifying microstructures for given desired properties (materials design) and finding optimal processing paths for targeted microstructures (process design). These identification problems are typically not well-posed in the sense of Hadamard [5], which presents a significant challenge for solution approaches [6]. However, the non-unique nature of these problems offers an important advantage for processing: It enables a more flexible production as processes can be efficiently guided to manufacture the best reachable microstructure from a set of equivalent microstructures with respect to their properties.

Leveraging this advantage presents certain challenges that require specialized yet generic materials and process design approaches. These approaches must be capable of identifying multiple solutions to their individual design problems and must work in conjunction with each other. In an effort to address this challenge, the present paper proposes a solution by combining two recently developed machine learning ap-
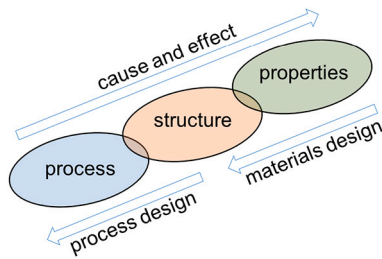
Fig. 1. Process-structure-property chain following [4].

proaches, as described in [7] and [8]. The motivation behind using these approaches is elaborated in the following.

To handle the complexity and dimensionality of the identification problems, the use of machine learning has shown to be suitable for materials and process design applications [9,10]. In this work, a novel machine learning framework is introduced that combines the reinforcement learning-based process design approach developed in [7]: multi-equivalent goal structure-guided processing path optimization (MEG-SGGPO), with the materials design approach developed in [8]: Siamese multi-task learning-based optimization (SMTLO). The framework is tailored to the specifics of process-structure-property optimization problems and, therefore, constitutes a significant advancement towards accelerated process and materials design.

Although each of the approaches has been shown to work well for their individual design problems, the combined usage of both is not investigated yet. This is particularly the aim of the present work, while the approaches are enhanced by using a recently developed novel distance measure for one-point statistics microstructure representations: the Sinkhorn distance $\mathscr{D}_{\mathrm{sh}}$ [12]. Specifically, the Sinkhorn distance is highly suitable as it takes into account neighborhood information encoded in histogram-based microstructure representations. In this work, we demonstrate the approach at manufacturing metallic materials with desired elastic and anisotropy properties, which are affected by the crystallographic texture that evolves during forming.

1.2. Related work

Pioneering work has been done in the field of materials and process design, however, typically either focusing on solving materials design problems (without taking into account processing) or focusing on solving process design problems for given desired properties (not taking into account the microstructure of materials) [13–17]. In the following, we discuss works that describe approaches for solving materials and process design problems with an emphasize on the application of machine learning.

A widely-used approach for designing materials is the microstructure sensitive design (MSD) approach [16]. This approach primarily focuses on identifying microstructures that exhibit desired properties, with six out of the seven MSD steps being dedicated to this task. The processing of microstructures is only briefly addressed by the final MSD step. To the authors knowledge, there are only few works that show how to set up process-structure-property linkages for crystallographic texture optimization within the framework of MSD. One approach involves modeling crystallographic texture evolution as fluid flow in the orientation space. On this basis, so-called processing streamlines are used to guide from one point to another [15,16].

For example, in the case of optimizing the crystallographic texture of an orthotropic plate, these streamlines were calculated for the processing operations of tension and compression, and used to guide from a random crystallographic texture to required crystallographic textures [17]. These required crystallographic textures have been identified in beforehand based on the MSD approach [13]. Alternatively, so-called texture evolution networks have been developed within the context of MSD, using a priory sampled processing paths to create a directed tree

graph where microstructures are represented by the nodes on the graph [18]. Graph search algorithms are then used to find optimal processing paths from an initial to a targeted microstructure.

Besides the MSD approach, other approaches exist that solve crystallographic texture optimization problems in terms of materials design, however, without solving a corresponding optimal processing problem. For instance, Kuroda and Ikawa [19] used a genetic algorithm to identify optimal combinations of typical fcc rolling (crystallographic) texture components for given desired properties. Also, Liu et al. [20] used optimization algorithms, but incorporated machine learning techniques to efficiently identify significant features and regions of the orientation space. Surrogate-based optimization has also been explored in several studies, such as for handling uncertainties in materials design [21]. Alternatively, probabilistic modeling approaches can be used to directly to solve the inverse identification problem [22].

Regarding process design, machine learning-based and data mining-based approaches have been proposed by several works, however, without solving the corresponding materials design problem in beforehand. One approach involves the use of principle component analysis to represent one-step and two- to three-step deformation processes and to identify the deformation sequences required for reaching target crystallographic textures [23,24]. Alternatively, a database approach can be used that stores microstructure representations and corresponding processing paths [14]. The database can be searched for desired crystallographic textures yielding optimal process paths. These process paths can then be fine-tuned using gradient-based optimization. Another option is to store microstructure representations in a lower dimensional feature space generated by a variational autoencoder [25]. In this lower dimensional feature space, optimal processing paths can be identified using a suitable distance measure. A more recent work for process design in terms of crystallographic texture, although optimizing process-property linkages, is described in [26]. Therein, neural networks are used to learn texture evolution in a deformation process, while an optimizer is used to drive the process to produce a material with desired properties. In this framework, however, crystallographic texture is varied only indirectly and it cannot be guaranteed that an optimal texture is found in terms of material properties and reachability by the process.

Recent research directions in the field of materials and process design are the usage of probabilistic methods such as Bayesian approaches to tackle the ill-posedness of the inverse problem. In [27] a Bayesian methodology was used to estimate a posterior distribution of microstructures that are conditioned by a user-defined target property given a prior distribution. A similar approach was used in [28], in which optimal process parameters were estimated for manufacturing a material with given desired material properties (process-property linkage). In terms of process-structure optimization problems, in [29], a Bayesian active learning approach is proposed that is based on Gaussian processes. While Bayesian methods generally suit well for solving such inverse problems, it can be hard to train these when the amount of data increases. Therefore, active learning approaches were developed that do not use Gaussian processes but instead make use of a committee-based active learning approach [30]. Other novel approaches in materials engineering make use of knowledge graphs for microstructure representation and the application of graph neural networks for property prediction, as demonstrated in [31]. Such approaches are rather descriptor-based approaches, such as the ones presented in [32] and [33] for meta materials design.

Other recent approaches for designing materials make use of generative models, such as in [11]. Therein, a generative adversarial network (GAN) is modified to generate microstructures based on continuous variables, which is an important prerequisite for designing materials. In [34], a further developed continuous conditioned GAN for diverse outputs was introduced and applied to a synthetic and real-world airfoil design task. Alternatively, in [35] a diffusion-based machine learning model is used for microstructure reconstruction with respect to given material properties. An alternative, descriptor-based approach for mi-

crostructure reconstruction can be found in [36]. Furthermore, since the work by [25], variational auto-encoder-based approaches gain popularity, such as in [37] for the design of quantum-mechanical properties in small organic molecules. Recent developments in the design of molecules investigate the potential of using large language models, such as in [38].

In the future, it is envisioned that design approaches will be implemented as applications on data platforms, as described in [39]. Such platforms, like the one described in [40], enable the storage of heterogeneous material data in a FAIR manner, ensuring that the data is findable, accessible, interoperable, and reusable [41]. This, in turn, allows for (automated) data processing using applications, such as based on simulation or machine learning. The utilization of these platforms opens up the possibility of employing design approaches for experimentation, which has the potential to lead to autonomous materials discovery and manufacturing [42].

1.3. Structure of this work

In the following Section 2, we introduce the structure-guided materials and process design approach as well as the metal forming process simulation at which the functionality of the approach is demonstrated and the used representation of crystallographic texture. Afterwards, in Section 3, the results for optimizing the crystallographic texture in the metal forming process for given desired properties are shown. The results are discussed in Section 4 and an outlook is given in Section 5.

2. Methods

The methods section is split into two parts: First, we present the basics for the structure-guided materials and process design approach studied in this paper, and, second, we present the domain specific application case at which we evaluate the presented approach.

2.1. Structure-guided materials and process design

2.1.1. General concept

The approach presented in this paper combines two machine learning approaches for materials design, namely Siamese multi-task learning-based optimization (SMTLO), and for process design, namely multi-equivalent goal structure guided processing path optimization (MEG-SGGPO), see Fig. 2. The approach starts in the properties space, where a target region of desired properties is defined. Then, the SMTLO approach is applied to identify a set of diverse microstructures that yield material properties inside the target region (Step 1). When having identified this set, MEG-SGGPO identifies the process path that leads to the best reachable microstructure (Step 2). The microstructure space is linking process and properties, and, therefore, is of overall importance for our approach. A suitable distance measure is one of the core ingredients for the approach to work. The proposed approach is generic in nature, making it applicable to any kind of process-structure-property relations. Furthermore, it is not limited to particular input/output data formats, and can even be applied to image data.

2.1.2. Siamese multi-task learning-based optimization (SMTLO)

The SMTLO approach described in [8] consists of a Siamese neural networks-based multi-task learning model and an optimizer. The latter uses the predictions of the learned neural network model to generate candidate microstructures that are supposed to (i) yield properties inside a defined target region and (ii) are reachable by the underlying manufacturing process, see Fig. 3. The details of the SMTLO approach is described in the following. Alternative approaches, that focus only the mapping of microstructures to properties can be found, for example in [43].

The multi-task learning model is grounded on an encoder f_{enc} that transforms a microstructure representation x into a lower dimensional latent feature space representation z

$$z = f_{\text{enc}}(x, \theta_{\text{enc}}), \tag{1}$$

with the trainable parameters θ_{enc}. The encoder is equipped with three heads, each of which is designed to solve a specific learning task. These are

1. a decoder head that reconstructs the microstructure representations x from z

$$x' = f_{\text{dec}}(z, \theta_{\text{dec}}), \tag{2}$$

with the trainable parameters θ_{dec}. The encoder-decoder part is trained using a loss term that minimizes the Sinkhorn distance between the original microstructure x and its reconstruction x' [12]

$$\mathcal{L}_{\text{recon}} = \mathcal{D}_{\text{sh}}(x, x'). \tag{3}$$

2. a prediction head f_{regr} to infer material properties $\hat{p}$

$$\hat{p} = f_{\text{regr}}(z, \theta_{\text{regr}}), \tag{4}$$

with the trainable parameters θ_{regr}. The sample-wise loss term is defined by the mean squared error between predicted properties $\hat{p}$ and true properties p:

$$\mathcal{L}_{\text{regr}} = \frac{1}{n_{\text{p}}} \sum_{i}^{n_{\text{p}}} (p_i - \hat{p}_i)^2, \tag{5}$$

with the number of properties n_{p}.

3. an auto-encoder head that serves as an anomaly detector to estimate whether a microstructure in its latent representation belongs to the set of known microstructures (defined by the training data) or not:

$$z' = f_{\text{valid}}(z, \theta_{\text{valid}}), \tag{6}$$

with the trainable parameters θ_{valid}. For this auto-encoder, a mean squared error loss function is used:

$$\mathcal{L}_{\text{valid}} = \frac{1}{n_{\text{z}}} \sum_{i}^{n_{\text{z}}} (z_i - z'_i)^2, \tag{7}$$

with n_{z} dimensions of the latent feature space.

The neural networks-based multi-task learning model that is used to solve the above mentioned learning tasks is trained using a combined loss function

$$\mathcal{L}_{\text{MTL}} = \mathcal{W}_{\text{recon}} \mathcal{L}_{\text{recon}} + \mathcal{W}_{\text{regr}} \mathcal{L}_{\text{regr}} + \mathcal{W}_{\text{valid}} \mathcal{L}_{\text{valid}}, \tag{8}$$

with individually weighted loss terms using the parameters $\mathcal{W}_{\text{recon}}$, $\mathcal{W}_{\text{regr}}$ and $\mathcal{W}_{\text{valid}}$.

The objective of the materials design step is to enable the optimizer to identify a diverse set of microstructures for given desired material properties, each microstructure being reachable by the underlying manufacturing process. In order to quantify diversity, a distance measure is required in the latent feature space. To ensure that the original microstructure distance is preserved in the lower dimensional latent feature space, the multi-task learning model is embedded in a Siamese neural network model [44]. This involves training two twin models simultaneously, with shared weights in the encoder, and adding a further loss term that applies to the latent feature space. Note that the two twin models are trained using different input and output vectors, denoted with the subscripts $_L$ and $_R$ in the following. The preservation loss leads to multidimensional scaling (see [45] and [46]) and writes

$$\mathcal{L}_{\text{pres}} = (\mathcal{D}_{\text{sh}}(x_L, x_R) - \text{dist}(z_L, z_R))^2, \tag{9}$$

with the absolute distance in the latent feature space

$$\text{dist}(z_L, z_R) = \frac{1}{n_{\text{z}}} \sum_{i}^{n_{\text{z}}} |z_{L,i} - z_{R,i}|. \tag{10}$$
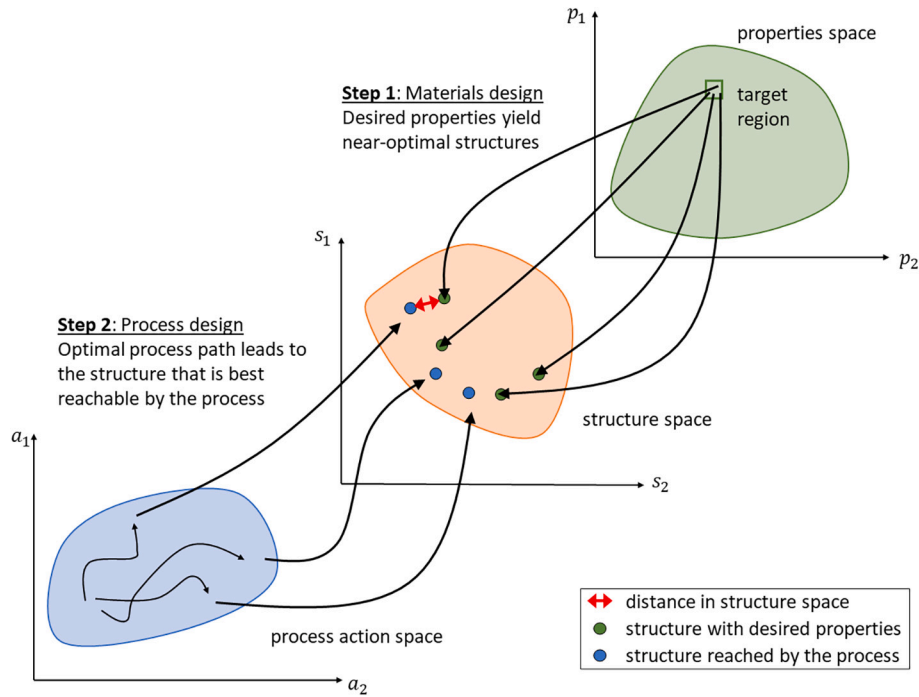
Fig. 2. General concept for structure-guided materials and process design. The first step (materials design) is addressed by the SMTLO approach, while the second step (process design) is addressed by the MEG-SGGPO approach.
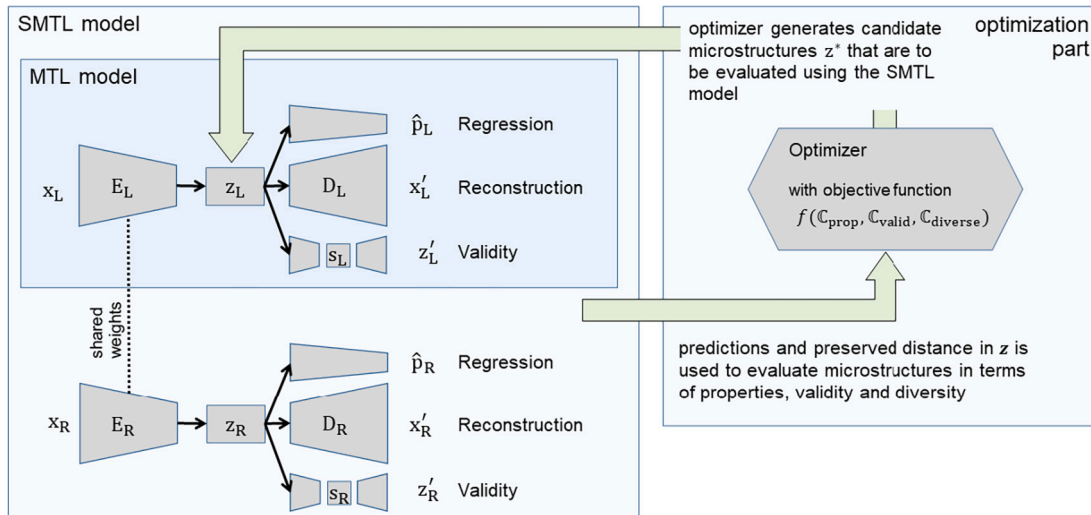

Fig. 3. The neural networks-based Siamese multi-task learning model (left) and the optimization part (right). Together, both parts build the SMTLO approach to solve materials design problems.

The overall loss function writes

$$\mathcal{L} = \mathscr{W}_{\text{recon}} \mathscr{L}_{\text{recon}} + \mathscr{W}_{\text{regr}} \mathscr{L}_{\text{regr}}$$
$$+ \mathscr{W}_{\text{valid}} \mathscr{L}_{\text{valid}} + \mathscr{W}_{\text{pres}} \mathscr{L}_{\text{pres}} + \lambda \Omega(\boldsymbol{\theta}), \quad (11)$$

with the regularization term $\Omega(\boldsymbol{\theta})$, and the weight $\mathscr{W}_{\text{pres}}$ for the preservation loss.

After training the Siamese multi-task learning model, we use a genetic optimizer [47] to search for candidate microstructures in the obtained latent feature space. The optimizer minimizes three cost terms: $\mathscr{C}_{\text{prop}}, \mathscr{C}_{\text{valid}}, \mathscr{C}_{\text{divers}}$, which are explaining in detail in [8]. Briefly described, $\mathscr{C}_{\text{prop}}$ describes the distance between the predicted properties $\hat{p}$ for a candidate microstructure z^* to the target region. $\mathscr{C}_{\text{valid}}$ aims to penalize candidate microstructures that are not inside the region of known

microstructures. $\mathscr{C}_{\text{divers}}$ aims to enforce candidate microstructures to be as far away as possible from each other in the actual population (in other words as diverse as possible). The overall objective function writes

$$\mathscr{C} = \mathscr{V}_{\text{prop}} \mathscr{C}_{\text{prop}} + \mathscr{V}_{\text{valid}} \mathscr{C}_{\text{valid}} + \mathscr{V}_{\text{divers}} (1 + \mathscr{C}_{\text{divers}}), \quad (12)$$

with individual weights $\mathscr{V}_{\text{prop}}, \mathscr{V}_{\text{valid}}$, and $\mathscr{V}_{\text{divers}}$. The property cost term directs the candidate microstructures to align with the desired target properties. The validity cost ensures that the optimizer stays within the region of valid microstructures, while the diversity cost guarantees sufficient variation among the candidate microstructures. By optimizing the objective function in Equation (12), only those candidate microstructures that meet the necessary criteria are selected for the process design step.

2.1.3. Multi-equivalent goal structure-guided processing path optimization (MEG-SGGPO)

In reinforcement learning, an agent learns to take optimal decisions by interacting with its environment in a sequence of discrete time steps t. The MEG-SGGPO approach [7] is tailored for efficient learning in the case of multiple equivalent target microstructures. Given a set $\mathcal{G}$ of near-optimal microstructures $\breve{g}^i \in \mathcal{G}$, an initial microstructure g_0, and a microstructure generating process $p(g_t, a_t) = g_{t+1}$, the go of the approach is to find a sequence of processing parameters $a_t \in A$ that guide the process towards manufacturing the best reachable microstructure $g^* \in \mathcal{G}$.

MEG-SGGPO is based on deep Q-networks [48] and its extensions [49–51]. Deep Q-networks are derived from Q-learning, where the so-called Q-function is approximated from data gathered by the reinforcement learning agent while following a policy $a = \pi(g)$. The Q-function models the expected sum of future reward signals for state action pairs (g_t, a_t). In MEG-SGGPO the Q-function is generalized to also take the goal microstructure into account and is, in its recursive form, defined as

$$Q(g_t, a_t, \breve{g}^i) = \mathbb{E}_{P,\pi}\left[R(g_t, g_{t+1}, \breve{g}^i) + \max_{a_{t+1} \in A} Q(g_{t+1}, a_{t+1}, \breve{g}^i) \right], \quad (13)$$

where $R(g_t, g_{t+1}, \breve{g}^i)$ is a per goal microstructure pseudo reward function

$$R(g_t, g_{t+1}, \breve{g}^i) = \frac{1}{\mathscr{D}(g_{t+1}, \breve{g}^i)} - \frac{1}{\mathscr{D}(g_t, \breve{g}^i)}, \quad (14)$$

which is based on a microstructure distance function $\mathscr{D}$.

The process design task is a two-fold optimization problem to (i) identify the best reachable goal microstructure $\breve{g}^{i*}$ and (ii) learn the optimal policy π^*. MEG-SGGPO addresses this by using the generalized functions defined above in a nested reinforcement learning procedure. At the beginning of each episode (i.e. an execution of the process during learning), the agent uses the generalized Q-function to identify the targeted microstructure $\breve{g}^{i'}$ from $\mathcal{G}$ and determines the processing parameters during the episode. In a first step, an estimation of the best reachable goal microstructure $\breve{g}^{i*}$ is extracted from the current Q estimation by

$$\breve{g}^{i*} = \arg\max_{\breve{g}^i \in \mathcal{G}} \left[\mathcal{V}(g_0, \breve{g}^i) + \frac{1}{\mathscr{D}(g_0, \breve{g}^i)} \right], \quad (15)$$

where the generalized state-value function $\mathcal{V}$ is defined as

$$\mathcal{V}(g, \breve{g}^i) = \max_{a \in A} Q(g, a, \breve{g}^i). \quad (16)$$

The targeted microstructure is chosen in an ϵ-greedy approach [52], where the agent chooses random targets and processing parameters in an ϵ fraction of the cases ($0 \le \epsilon \le 1$) and optimizes targets and processing parameters in the remaining cases. The optimal processing path is identified simultaneously in an inner loop by using the Q-learning approach.

2.2. Application case

2.2.1. Crystallographic texture optimization

To study it's functionality, the structure-guided materials and process design approach is applied to a crystallographic texture optimization problem in a metal forming process. The process-structure-property chain adapts to cold forming operations that change the crystallographic texture of a material, which significantly affects elastic and plastic properties. The underlying metal forming process simulation is described in the following section, as well as the determination of the properties that we are going to target.

We have chosen crystallographic texture optimization as the focal point of this paper for the following reasons: Crystallographic texture of polycrystals and their corresponding properties is one of the core topics of materials engineering. In particular, understanding how microstructural features affect the elastic and anisotropic properties of metals plays an important role in manufacturing processes such as sheet metal forming. At the same time, crystallographic texture is a complex and high-dimensional microstructural feature, rendering the materials and process design problems inherently challenging. Therefore, it serves as an ideal benchmark problem to demonstrate the effectiveness of our proposed approach.

2.2.2. Metal forming process simulation

In this study, we use the metal forming process simulation as is described in [7]. The simulation applies a deformation $\hat{F}$

$$\hat{F} = R \widetilde{F} R^\top, \quad (17)$$

with R being a rotation matrix that describes one out of 25 possible loading directions. The deformation $\widetilde{F}$ is defined using orthogonal basis vectors e_i and the operator $\otimes$ for the dyadic product

$$\widetilde{F} = \widetilde{F}_{11} e_1 \otimes e_1 + \widetilde{F}_{22} e_2 \otimes e_2 + \widetilde{F}_{33} e_3 \otimes e_3 \quad (18)$$

with $\widetilde{F}_{11}$ corresponding to 10% strain increments. $\widetilde{F}_{22}, \widetilde{F}_{33}$ are adjusted such that the stresses are in balance. The metal forming process consists of seven subsequent loading steps, each in a separate loading direction. For this study, we conducted 76980 random process paths to generate the training data for the SMTLO approach.

The underlying material model is a crystal plasticity model of Taylor-type [53]. The volume averaged stress for n_{oris} crystals with different orientations is calculated by

$$\overline{T} = \frac{1}{V} \sum_i^{n_{\mathrm{oris}}} T^{(i)} V^{(i)}, \quad (19)$$

with the total volume V, the individual volume of each crystal $V^{(i)}$, and the Cauchy stress tensor $T^{(i)}$.

With the multiplicative decomposition of the deformation gradient in its elastic and plastic part

$$F = F_{\mathrm{e}} \cdot F_{\mathrm{p}}, \quad (20)$$

and the conversion formula for the stress tensor in the intermediate configuration T^*

$$T^* = F_{\mathrm{e}}^{-1} \cdot (\det(F_{\mathrm{e}})T) \cdot F_{\mathrm{e}}^{-\top}, \quad (21)$$

the Cauchy stress tensor is derived using

$$T^* = \frac{1}{2} \mathbb{C} : (F_{\mathrm{e}}^\top \cdot F_{\mathrm{e}} - I), \quad (22)$$

where I denotes the second order identity tensor and $\mathbb{C}$ the fourth order elastic stiffness tensor.

The evolution of the plastic deformation is described using the plastic part of the velocity gradient

$$L_{\mathrm{p}} = \dot{F}_{\mathrm{p}} \cdot F_{\mathrm{p}}^{-1} = \sum_\alpha \dot{\gamma}^{(\alpha)} m^{(\alpha)} \otimes n^{(\alpha)}, \quad (23)$$

with the slip rates $\dot{\gamma}^{(\alpha)}$ on slip system α that is defined by the slip plane normal $n^{(\alpha)}$ and the slip direction $m^{(\alpha)}$. The slip rates are calculated by a phenomenological power law. The crystal reorientation is calculated by applying a rigid body rotation derived from the polar decomposition of F_{e} to the original orientation, see [54,12]. For the metal forming process simulation, the same material model parameters have been used as in [7].

For the purpose of this study, we use the above described material model to evaluate the Young's modulus E and an anisotropic property $\widetilde{R}$ in three orthogonal directions after a process run. The Young's modulus is calculated using the slope of the stress-strain curve in the elastic regime, that results after applying uniaxial tension. $\widetilde{R}$ is inspired by the Lankford coefficients in sheet metal forming and is calculated as the ratio between transverse strain and the strain in loading direction in the plastic regime, also after applying uniaxial loading.
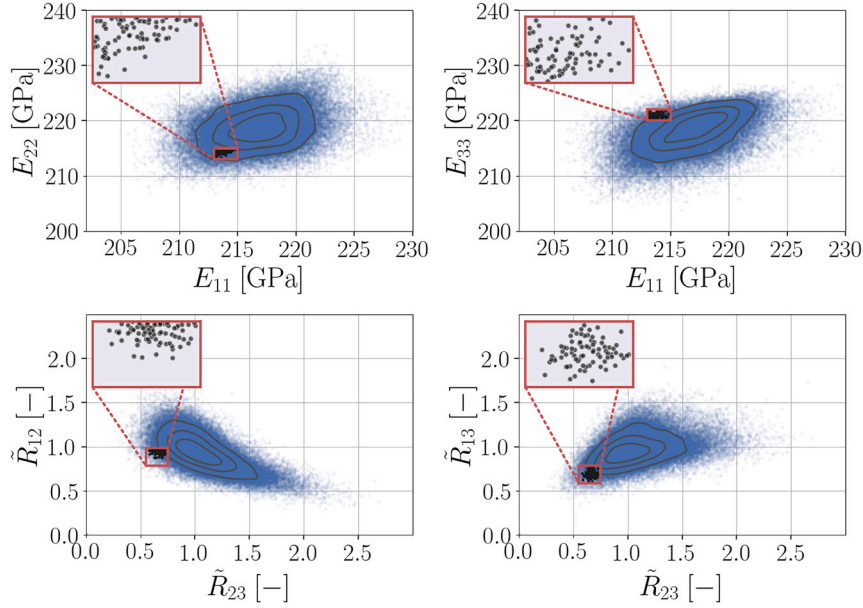
Fig. 4. Projections of the property space showing the data distribution and the target region. The distribution of the underlying point cloud is displayed in blue, while the gray dots mark data points that are already located inside the target region. The black isolines indicate regions with the same point cloud density.

2.2.3. Crystallographic texture representation and texture distance measure

Crystallographic texture is represented by using the histogram-based description introduced by Dornheim et al. [7]. The orientation histogram (here, we used a soft-assignment factor of 3) consists of 512 orientation bins that are nearly uniformly distributed in the cubic-triclinic fundamental zone. The bin orientations were created using the software *neper* [55,56]. In contrast to the original MEG-SGGPO and SMTLO approach, in this work, crystallographic texture distances are measured using the Sinkhorn distance that is applied to the histogramm representations [12]. Specifically, the Sinkhorn distance is an efficient implementation of the Earth Movers distance that measures the least amount of work necessary to transform one histogram into the other [57,58]. Therefore, local orientation distances [59] encoded in the histograms are taken into account, in contrast to the originally proposed Chi-Squared distance, which is basically a bin-wise comparison of two histograms.

3. Results

3.1. Solving the materials and process design tasks

The following demonstrates how the proposed approach can be implemented to solve a coupled materials and process design problem, specifically in relation to processing crystallographic textures with desired properties. First, it is shown how near-optimal crystallographic textures are identified for given elastic and anisotropy properties using SMTLO. The properties are in particular the Young's moduli E_i and anisotropy measures $\widetilde{R}_i$, both in three orthogonal directions. Once near-optimal crystallographic textures are identified, in a second step, MEG-SGGPO is used to guide the underlying manufacturing process to produce the best reachable crystallographic texture out of the set of identified near-optimal textures.

The basis for applying the SMTLO approach is a data set of 76980 samples, composed of crystallographic textures and corresponding properties. 2D projections of the training data distribution, generated by metal forming process simulation, are shown in Fig. 4 including the region delineating the desired properties (target region). In this study, the target region is centered at $E_{11}, E_{22}, E_{33} = 214, 214, 221$ GPa and $\widetilde{R}_{23}, \widetilde{R}_{12}, \widetilde{R}_{13} = 0.65, 0.685, 0.885$. Its width equals 2 GPa for E_i and 0.2 for $\widetilde{R}_i$. Both E_i and $\widetilde{R}_i$ are determined in three orthogonal directions at the material point at the end of the process. E_i is given by the slope in

the elastic regime, while $\widetilde{R}_i$ is calculated by the ratio between transverse and tensile strain in the plastic regime when applying uniaxial tension. In Fig. 4, we highlighted training data points that are already located inside the target region. We use this set of 80 crystallographic textures as benchmark set.

3.2. Materials design using SMTLO

The Siamese multi-task learning model is realized via feedforward neural networks with *tanh* activation functions and are implemented based on the Pytorch API [60]. For hyperparameter optimization the random search method [61] is utilized using 5-fold cross-validation. The training data set comprises 80% of the total data, while the test data set accounts for the remaining 20%. The following hyperparameters have been chosen: The Glorot Normal method [62] is used for weight initialization and the Adam optimizer [63] is used with the following parameters: learning rate = 0.001, weight decay = 10^{-6}, batch size = 128.

The Siamese multi-task learning model is trained for 100 epochs, where the best intermediate result of the test set is retained to prevent over fitting and to apply early stopping [64]. Before the model is trained, the loss terms are scaled to values between 0 and 1 in order to make them comparable. The following weights for the scaled loss terms in Eq. (11) were based on hyper parameter optimization: $\mathscr{W}_{\text{recon}} = 0.2$, $\mathscr{W}_{\text{regr}} = 0.06$, and $\mathscr{W}_{\text{valid}} = 0.04$, and $\mathscr{W}_{\text{pres}} = 0.70$. The loss curves resulting from the model training are collected for both, the test and the training dataset, for each individual epoch and are shown for each task separately in Fig. 5. It can be seen that for the validity, reconstruction, and regression tasks, the loss is minimized up to epoch 50 with no signs of overfitting. For the distance preservation task, the loss continues to decrease beyond epoch 50, also without any overfitting observed. For the cost function of the optimizer (defined in Eq. (12)) that we need after having a trained Siamese multi-task learning model, we use the weights $\mathscr{V}_{\text{prop}} = 0.75$, $\mathscr{V}_{\text{valid}} = 0.05$, and $\mathscr{V}_{\text{divers}} = 0.2$.

The results for the properties prediction are given by the mean absolute error (MAE) between the true and predicted Young's moduli and $\widetilde{R}$-values: $\text{MAE}_E = 0.368$ [GPa] and $\text{MAE}_{\widetilde{R}} = 0.032$ [-]. The result of the distance preservation is measured by the coefficient of determination R^2, between the Sinkhorn distance of two input crystallographic textures and the l_1 distance of their corresponding latent feature vectors: $R^2(\mathscr{D}_{\text{sh}}(\boldsymbol{x}_L, \boldsymbol{x}_R), l_1(\boldsymbol{z}_L, \boldsymbol{z}_R)) = 90.22[\%]$.
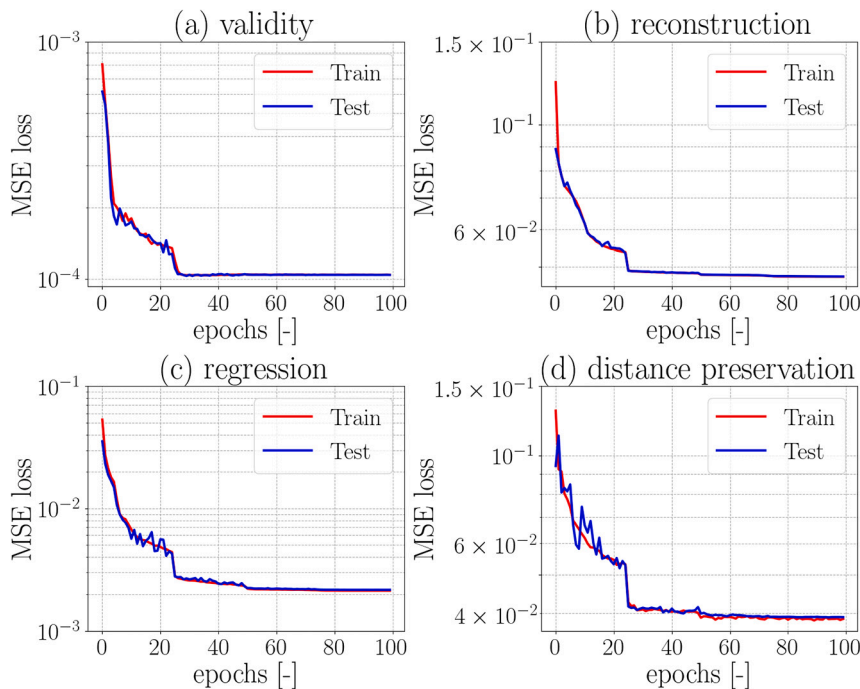
Fig. 5. Loss curves of the Siamese multi-task learning model for the test dataset (blue) and the training dataset (red) across tasks: (a) validity, (b) reconstruction, (c) regression, and (d) distance preservation, plotted on a logarithmic scale.
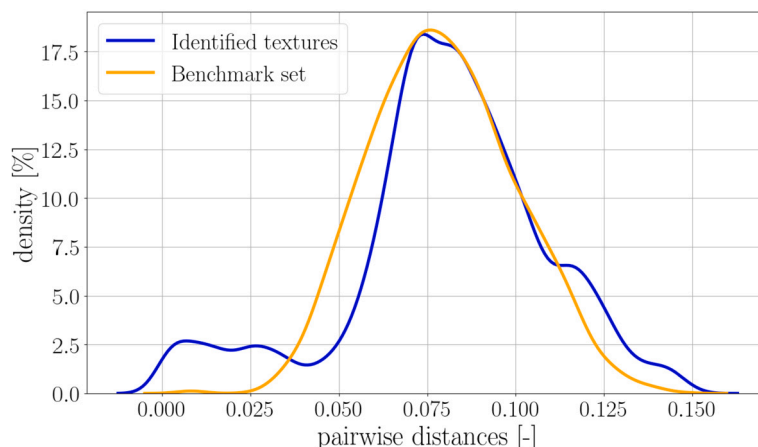

Fig. 6. Density of pairwise distances between crystallographic textures belonging to the set of identified crystallographic textures (red) and between crystallographic textures belonging to the benchmark set (green), evaluated in the latent space of the Siamese multi-task learning model.

Using the SMTLO approach (with the trained multi-task learning model as described above), we were able to identify a set of 175 near-optimal crystallographic textures. As we aim to identify a preferably diverse set of crystallographic textures, we depict their pairwise distances in Fig. 6 and compare them to the benchmark set. As one can easily see, the SMTLO approach is able to find a set of crystallographic textures that differ more to each other than the benchmark set.

For the subsequent process design step, we chose ten goal textures from the set of identified crystallographic textures that differ strongly from each other [65]. The distribution of the chosen goal textures in properties space is shown in Fig. 7. The properties of the goal textures are mainly located inside the target region, with some exceptions that we tolerate for now and explain later in Section 4. To show the differences between the goal textures, in Fig. 8, four crystallographic textures are depicted exemplary as pole figure plots. While the pole figure intensities are similar for all crystallographic textures, the shape of the represented orientation distribution differs strongly.

The artificial neural networks and the reinforcement learning approach were trained a workstation equipped with 20 2.2 GHz CPU cores and a GeForce RTX 3090 GPU with 24GB RAM by using the Linux Ubuntu operating system.

3.3. Process design using MEG-SGGPO

Experiments are conducted with the following hyperparameters: Deep Q-learning as described in Section 2.1.3 as basic algorithm where the target-network is updated every $n_\theta = 50$ time-steps. Q-networks with hidden layer sizes of [128, 256, 256, 128], layer normalization and ReLU activation functions. The learning process starts after 100 control-steps. The networks are trained after each control-step with a mini-batch of size 32. The Adam optimizer [63] is used for neural network training, with a learning rate of 5×10^{-4}. An ϵ-greedy policy is applied, with an initial exploration rate $\epsilon_0 = 0.5$ and the final exploration-rate $\epsilon_f = 0.0$, with $n_\epsilon = 390$.
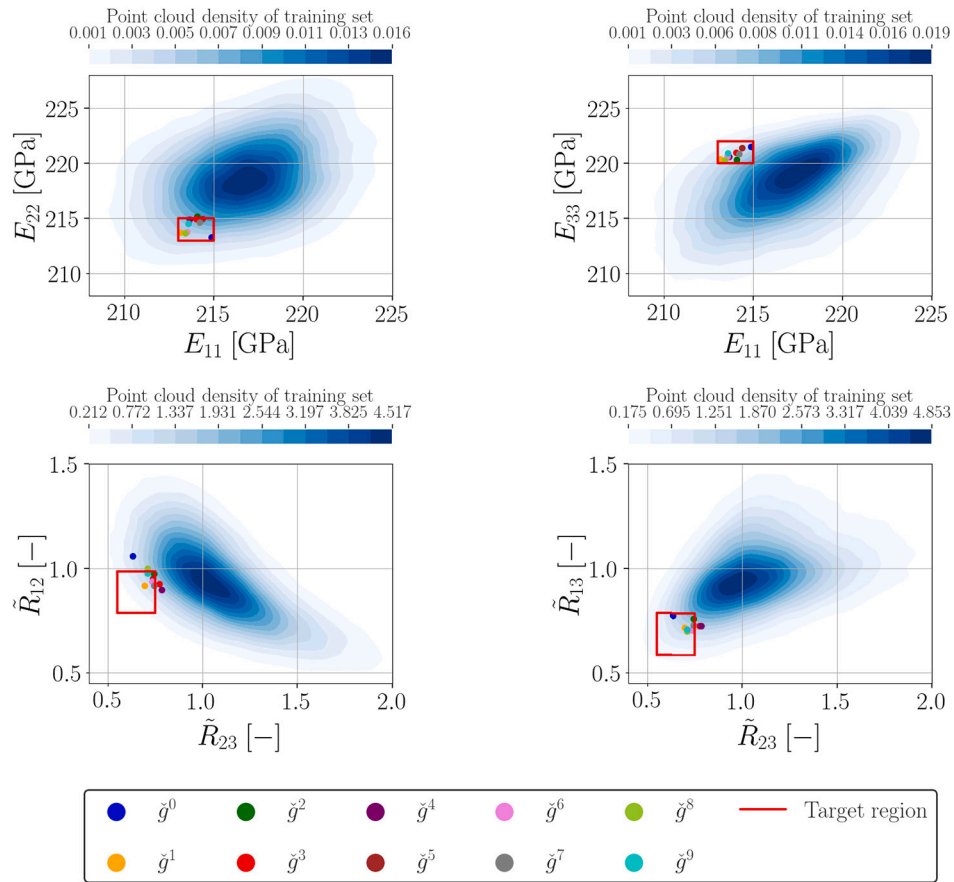
Fig. 7. Projections of the properties space showing the target region and the calculated properties of the ten chosen goal textures $\breve{g}^0$ to $\breve{g}^9$.
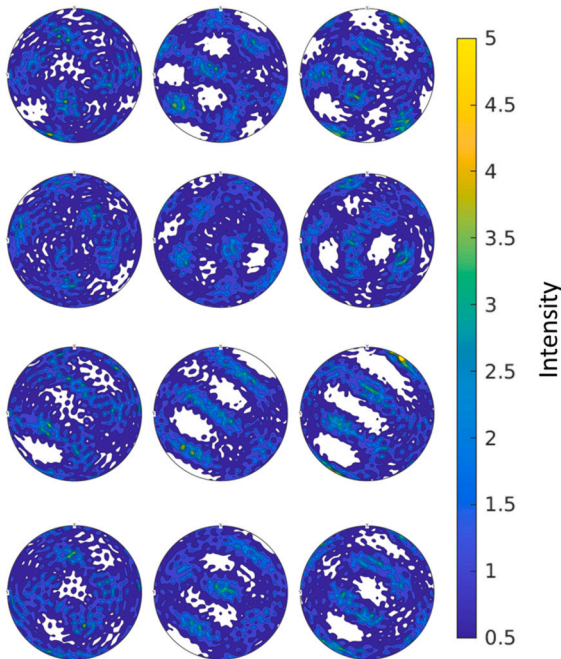

Fig. 8. Four exemplary crystallographic textures of the ten chosen goal textures depicted as pole figure plots: (001), (110), (111).

Once a diverse set of crystallographic textures has been identified, MEG-SGGPO is used to guide the metal forming process to the best reachable crystallographic texture. For solving the identification problem, the MEG-SGGPO approach is allowed to conduct 400 process runs,

Table 1

Properties of the produced crystallographic texture and the distance to the target region.

Target property	Value	Distance to target region	Unit
E_{11}	217.5	2.5	GPa
E_{22}	216.6	1.6	GPa
E_{33}	222.2	0.3	GPa
$\widetilde{R}_{23}$	0.744	0	-
$\widetilde{R}_{12}$	0.831	0.046	-
$\widetilde{R}_{13}$	1.058	0.073	-

so-called episodes. The evolution of the distance between the produced and the chosen goal texture is depicted over the episodes in Fig. 9. Initially, the reinforcement learning agent attempts to identify processing paths that target randomly selected goal textures. As the learning process progresses, the agent focuses increasingly on producing goal texture $\breve{g}^2$.

For comparison, Fig. 10 presents pole figure plots of the goal texture $\breve{g}^2$ and the produced crystallographic texture $\breve{g}^{2'}$ after 400 episodes. Visually, the produced crystallographic texture is highly similar to the goal texture with intensity peaks at the same positions and of a similar magnitude. This also holds for areas not covered by orientations at all. To quantify the effectiveness of the approach, we compute the properties resulting from the produced crystallographic texture and compare them with the desired ones. The results, listed in Table 1, show that five of the properties (E_{11}, E_{22}, E_{33}, $\widetilde{R}_{12}$ and $\widetilde{R}_{13}$) are very close to the target region, while $\widetilde{R}_{23}$ lies inside.

In summary, by sequentially applying the SMTLO and the MEG-SGGPO approach, we were able to successfully produce a crystallo-
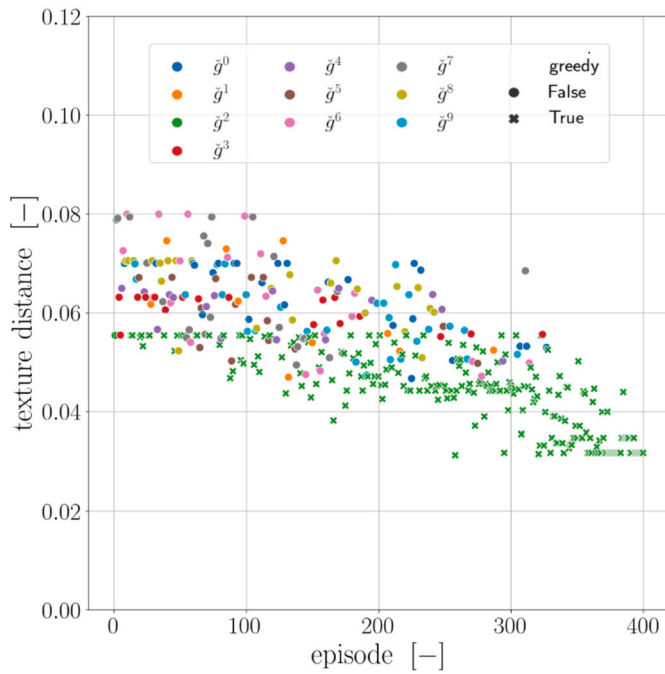
Fig. 9. Sinkhorn texture distance $\mathscr{D}_{sh}$ between the produced and the goal texture over the episodes of the MEG-SGGPO approach. The color indicates the targeted crystallographic texture.
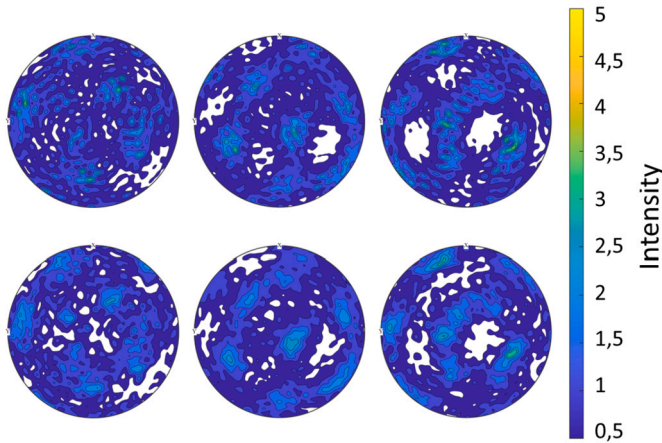

Fig. 10. Pole figure plots (001), (110), (111) of the targeted goal texture $\check{g}^2$ (top) and the produced crystallographic texture $\check{g}^{2'}$ (bottom).

graphic texture with desired properties in a simulated metal forming process. The resulting properties fall within the acceptable range from an engineering point of view, albeit at the border of the target region defined in property space.

4. Discussion

The results presented demonstrate the effectiveness of the SMTLO approach in identifying sets of near-optimal crystallographic textures that exhibit desired properties in a given target region. The set of crystallographic textures identified by the approach is shown to be more diverse than the benchmark set obtained from the training data set. However, the properties of the identified crystallographic textures are not completely inside the target region, as can be seen in Fig. 7.

There are two reasons for this: First, the formulation of the optimization objective in the SMTLO approach allows for a trade-off between (i) finding crystallographic textures with properties inside the target region, (ii) identifying possibly diverse crystallographic textures

and, (iii) guaranteeing that crystallographic textures can be produced by the process under consideration. Second, due to prediction errors of the underlying machine learning model, the SMTLO approach identifies crystallographic textures whose predicted properties lie inside the target region, while some of the true properties (calculated by the numerical simulation on the basis of the reconstructed crystallographic textures) slightly lie outside. Both of these issues can be addressed by modifying the objective function of the SMTLO optimizer, defined by Equation (12). For instance, targeting the center of the properties region, instead of its bounds can mitigate both issues. Additionally, the second issue can be addressed by enhancing the machine learning model, for example, by increasing the amount of training data.

Taking a diverse subset of the identified crystallographic textures as goal textures, the MEG-SGGPO approach guides the metal forming process closely to one of the chosen goal textures. Although the reinforcement learning agent was not able to reproduce one of the goal textures exactly (which is challenging due to the many possible processing paths), the measured distance between the targeted crystallographic texture $\check{g}^2$ and the produced one $\check{g}^{2'}$ is sufficiently low ($\mathscr{D}_{sh} = 0.031$) when relating it to the distance to the nearest neighbor in the training data set ($\mathscr{D}_{sh} = 0.033$) and to the furthest data point ($\mathscr{D}_{sh} = 0.141$). This can be seen in Fig. 9, which depicts the reduction in Sinkhorn distance from the initial texture g_0 to the final produced texture $\check{g}^{2'}$ guided by the reinforcement learning agent.

We want to remark here, that it is generally difficult to evaluate distances between crystallographic textures, even for experts. Therefore, the quality of the produced crystallographic texture is determined by two considerations: (i) the distance of the textures in the microstructure space (as illustrated above), which is statistically valid, and (ii) the distance in properties, which can be assessed based on the results shown in Table 1. It can be seen that for the produced texture, one property lies within the target region, while the remaining five properties are in close proximity. It is remarkable that it took the reinforcement learning agent only 400 episodes to achieve this result compared to the baseline set that is grounded on the generation of 76980 random samples. The reinforcement learning algorithm can therefore be seen as being data efficient.

Nevertheless, in this study, it seems that a lower bound is existing that is difficult to overcome by the MEG-SGGPO approach. In general, this can have two reasons: First, the forming process is unable to produce crystallographic textures that are closer to the goal texture identified by the SMTLO approach. Yet, the SMTLO approach already addresses this issue by enforcing microstructures to remain within the region delineated by the known microstructures from the training data set via the term $\mathscr{V}_{valid}\mathscr{C}_{valid}$ in Equation (12). We expected that a stronger enforcement leads to crystallographic textures that are better reachable by the process. This, however, can lead to a lower accuracy in terms of obtaining the desired material properties, due to the change in the other weights in the objective function of the SMTLO optimizer. Second, the MEG-SGGPO approach identified a local optimum and got stuck. This can be mitigated by longer MEG-SGGPO runs with optimized hyperparameters. Longer runs, however, have not shown to yield significant improvements in the presented study. As future work, it is desirable to incorporate the knowledge contained in the generated training samples for SMTLO into MEG-SGGPO a priori to enhance its performance.

5. Conclusion

In summary, the machine learning-based approach presented in this study enables accelerated materials development by taking into account the manufacturing process. This was particularly challenging as for the considered metal forming process a total of 25^7 different processing paths would be possible; we found that

- 76980 samples were sufficient for the SMTLO approach to successfully solve the materials design problem. It allowed us to identify

a diverse set of near-optimal crystallographic textures with desired properties. The crystallographic textures are considered near-optimal in the sense that all of them bear different but satisfactory macroscopic properties and, at the time point of the identification, it is not know which one is best reachable by the process.

- after solving the materials design problem, a total of 400 episodes was sufficient for the MEG-SGGPO approach to successfully guide the forming process to the best reachable crystallographic texture and to solve the process design problem.

This shows that the applied approach is highly data efficient and capable of effectively optimizing process-structure-property relations end-to-end in manufacturing processes. Moreover, we emphasize that the approach leverages the non-uniqueness of the materials and process design problems, i.e. a diverse set of microstructures is identified for given desired properties and subsequently, the best reachable microstructure is produced. This is a significant advantage when transferring the approach to real manufacturing systems, where often constraints exist, which may exclude microstructures from being producible.

Nonetheless, the application still needs to be validated in real manufacturing systems. In our specific case, the amount of required training data is relatively high as compared to what is available in typical production workflows. Thus, for application in real-world manufacturing, a future scenario could involve pre-training the machine learning models with a numerical simulation that serves as a digital twin of the process and the material.

CRediT authorship contribution statement

Lukas Morand: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Data curation, Conceptualization. **Tarek Iraki:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Conceptualization. **Johannes Dornheim:** Writing – review & editing, Software, Conceptualization. **Stefan Sandfeld:** Writing – review & editing, Visualization, Methodology. **Norbert Link:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization. **Dirk Helm:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT in order to improve language and readability. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

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.

Acknowledgements

Data availability

Data and code has been published in the following repositories:

- Training data for the SMTLO approach is made available via Fraunhofer Fordatis repository at https://fordatis.fraunhofer.de/handle/fordatis/319 [66].
- The process simulation is made available via Fraunhofer Fordatis repository at https://fordatis.fraunhofer.de/handle/fordatis/201.2 [67].
- The source code for the Siamese multi-task learning framework is made available via gitlab at https://gitlab.com/tarekiraki/the_smtl_framework.
- The reinforcement learning environment is made available via github at https://github.com/Intelligent-Systems-Research-Group/RL4MicrostructureEvolution.

References

[1] J.J. de Pablo, N.E. Jackson, M.A. Webb, L.-Q. Chen, J.E. Moore, D. Morgan, R. Jacobs, T. Pollock, D.G. Schlom, E.S. Toberer, et al., New frontiers for the materials genome initiative, npj Comput. Mater. 5 (1) (2019) 41.

[2] Advanced Materials Initiative 2030, The materials roadmap 2030, Tech. Rep., 2022.

[3] P. Grant, Foresight – new and advanced materials, Tech. Rep., 2013.

[4] G.B. Olson, Computational design of hierarchically structured materials, Science 277 (5330) (1997) 1237–1242.

[5] J. Hadamard, Sur les problèmes aux dérivées partielles et leur signification physique, Princeton University Bulletin, 1902, pp. 49–52.

[6] A. Agrawal, A. Choudhary, Perspective: materials informatics and big data: realization of the "fourth paradigm" of science in materials science, APL Mater. 4 (5) (2016) 053208.

[7] J. Dornheim, L. Morand, S. Zeitvogel, T. Iraki, N. Link, D. Helm, Deep reinforcement learning methods for structure-guided processing path optimization, J. Intell. Manuf. 33 (2021) 333–352.

[8] T. Iraki, L. Morand, J. Dornheim, N. Link, D. Helm, A multi-task learning-based optimization approach for finding diverse sets of material microstructures with desired properties and its application to texture optimization, J. Intell. Manuf. (2023).

[9] Y. Liu, T. Zhao, W. Ju, S. Shi, Materials discovery and design using machine learning, J. Mater. 3 (3) (2017) 159–177.

[10] M. Mozaffar, S. Liao, X. Xie, S. Saha, C. Park, J. Cao, W.K. Liu, Z. Gan, Mechanistic artificial intelligence (mechanistic-ai) for modeling, design, and control of advanced manufacturing processes: current state and perspectives, J. Mater. Process. Technol. 302 (2022) 117485.

[11] S. Bompas, S. Sandfeld, A generative model for accelerated inverse modelling using a novel embedding for continuous variables, arXiv:2311.11343, 2024.

[12] T. Iraki, L. Morand, N. Link, S. Sandfeld, D. Helm, Accurate distances measures and machine learning of the texture-property relation for crystallographic textures represented by one-point statistics, Model. Simul. Mater. Sci. Eng. 32 (5) (2024) 055016.

[13] S.R. Kalidindi, J.R. Houskamp, M. Lyons, B.L. Adams, Microstructure sensitive design of an orthotropic plate subjected to tensile load, Int. J. Plast. 20 (8–9) (2004) 1561–1575.

[14] V. Sundararaghavan, N. Zabaras, On the synergy between texture classification and deformation process sequence selection for the control of texture-dependent properties, Acta Mater. 53 (4) (2005) 1015–1027.

[15] D. Li, H. Garmestani, S. Ahzi, Processing path optimization to achieve desired texture in polycrystalline materials, Acta Mater. 55 (2) (2007) 647–654.

[16] D.T. Fullwood, S.R. Niezgoda, B.L. Adams, S.R. Kalidindi, Microstructure sensitive design for performance optimization, Prog. Mater. Sci. 55 (6) (2010) 477–562.

[17] B.L. Adams, S. Kalidindi, D.T. Fullwood, D. Fullwood, Microstructure Sensitive Design for Performance Optimization, Butterworth-Heinemann, 2012.

[18] J.B. Shaffer, M. Knezevic, S.R. Kalidindi, Building texture evolution networks for deformation processing of polycrystalline fcc metals using spectral approaches: applications to design for targeted performance, Int. J. Plast. 26 (8) (2010) 1183–1194.

[19] M. Kuroda, S. Ikawa, Texture optimization of rolled aluminum alloy sheets using a genetic algorithm, Mater. Sci. Eng. A 385 (1–2) (2004) 235–244.

[20] R. Liu, A. Kumar, Z. Chen, A. Agrawal, V. Sundararaghavan, A. Choudhary, A predictive machine learning approach for microstructure optimization and materials design, Sci. Rep. 5 (1) (2015) 1–12.

[21] P.V. Balachandran, D. Xue, J. Theiler, J. Hogden, T. Lookman, Adaptive strategies for materials design using uncertainties, Sci. Rep. 6 (1) (2016) 1–9.

[22] A. Tran, T. Wildey, Solving stochastic inverse problems for property–structure linkages using data-consistent inversion and machine learning, JOM 73 (1) (2021) 72–89.

[23] P. Acar, V. Sundararaghavan, Linear solution scheme for microstructure design with process constraints, AIAA J. 54 (12) (2016) 4022–4031.

[24] P. Acar, V. Sundararaghavan, Reduced-order modeling approach for materials design with a sequence of processes, AIAA J. 56 (12) (2018) 5041–5044.

[25] S. Sundar, V. Sundararaghavan, Database development and exploration of process–microstructure relationships using variational autoencoders, Mater. Today Commun. 25 (2020) 101201.

[26] J. Lin, M. Hasan, P. Acar, J. Blanchet, V. Tarokh, Neural network accelerated process design of polycrystalline microstructures, Mater. Today Commun. 36 (2023) 106884.

[27] A.P. Generale, A.E. Robertson, C. Kelly, S.R. Kalidindi, Inverse stochastic microstructure design, Acta Mater. 271 (2024) 119877.

[28] A.P. Generale, C. Kelly, G. Harrington, A.E. Robertson, M. Buzzy, S. Kalidindi, A bayesian approach to designing microstructures and processing pathways for tailored material properties, in: AI for Accelerated Materials Design-NeurIPS 2023 Workshop, 2023.

[29] A. Tran, J.A. Mitchell, L.P. Swiler, T. Wildey, An active learning high-throughput microstructure calibration framework for solving inverse structure–process problems in materials informatics, Acta Mater. 194 (2020) 80–92.

[30] L. Morand, N. Link, T. Iraki, J. Dornheim, D. Helm, Efficient exploration of microstructure-property spaces via active learning, Front. Mater. 8 (2022) 824441.

[31] A. Thomas, A.R. Durmaz, M. Alam, P. Gumbsch, H. Sack, C. Eberl, Materials fatigue prediction using graph neural networks on microstructure representations, Sci. Rep. 13 (1) (2023) 12562.

[32] S. Kumar, S. Tan, L. Zheng, D.M. Kochmann, Inverse-designed spinodoid metamaterials, npj Comput. Mater. 6 (1) (2020) 73.

[33] A. Raßloff, P. Seibert, K.A. Kalina, M. Kästner, Inverse design of spinodoid structures using bayesian optimization, preprint, arXiv:2402.13054, 2024.

[34] A. Heyrani Nobari, W. Chen, F. Ahmed, Pcdgan: a continuous conditional diverse generative adversarial network for inverse design, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 606–616.

[35] A. Wijaya, J. Wagner, B. Sartory, R. Brunner, Analyzing microstructure relationships in porous copper using a multi-method machine learning-based approach, Commun. Mater. 5 (1) (2024) 59.

[36] P. Seibert, A. Raßloff, M. Ambati, M. Kästner, Descriptor-based reconstruction of three-dimensional microstructures through gradient-based optimization, Acta Mater. 227 (2022) 117667.

[37] A. Fallani, L. Medrano Sandonas, A. Tkatchenko, Inverse mapping of quantum properties to structures for chemical space of small organic molecules, Nat. Commun. 15 (1) (2024) 6061.

[38] K.M. Jablonka, P. Schwaller, A. Ortega-Guerrero, B. Smit, Leveraging large language models for predictive chemistry, Nat. Mach. Intell. 6 (2) (2024) 161–169.

[39] D. Sivan, K. Satheesh Kumar, A. Abdullah, V. Raj, I.I. Misnon, S. Ramakrishna, R. Jose, Advances in materials informatics: a review, J. Mater. Sci. 59 (7) (2024) 2602–2643.

[40] Y. Nahshon, L. Morand, M. Büschelberger, D. Helm, K. Kumaraswamy, P. Zierep, M. Weber, P. de Andrés, Semantic orchestration and exploitation of material data: a dataspace solution demonstrated on steel and cooper applications, preprint, arXiv:2406.19509, 2024.

[41] M. Scheffler, M. Aeschlimann, M. Albrecht, T. Bereau, H.-J. Bungartz, C. Felser, M. Greiner, A. Groß, C.T. Koch, K. Kremer, et al., Fair data enabling new horizons for materials research, Nature 604 (7907) (2022) 635–642.

[42] S.T. Bukkapatnam, Autonomous materials discovery and manufacturing (amdm): a review and perspectives, IISE Trans. 55 (1) (2023) 75–93.

[43] B.D. Nguyen, P. Potapenko, A. Demirci, K. Govind, S. Bompas, S. Sandfeld, Efficient surrogate models for materials science simulations: machine learning-based prediction of microstructure properties, Mach. Learn. Appl. 16 (2024) 100544.

[44] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, R. Shah, Signature verification using a" siamese" time delay neural network, Adv. Neural Inf. Process. Syst. 6 (1993).

[45] J. Kruskal, Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis, Psychometrika 29 (1964) 1–27.

[46] M.A. Cox, T.F. Cox, Multidimensional scaling, in: Handbook of Data Visualization, Springer, Berlin Heidelberg, 2008, pp. 315–347.

[47] J. Zhang, A.C. Sanderson, Jade: adaptive differential evolution with optional external archive, IEEE Trans. Evol. Comput. 13 (5) (2009) 945–958.

[48] V. Mnih, K. Kavukcuoglu, D. Silver, A.A. Rusu, J. Veness, M.G. Bellemare, A. Graves, M. Riedmiller, A.K. Fidjeland, G. Ostrovski, et al., Human-level control through deep reinforcement learning, Nature 518 (7540) (2015) 529–533.

[49] H. Van Hasselt, A. Guez, D. Silver, Deep reinforcement learning with double q-learning, in: Thirtieth AAAI Conference on Artificial Intelligence, 2016.

[50] T. Schaul, J. Quan, I. Antonoglou, D. Silver, Prioritized experience replay, arXiv: 1511.05952, 2015.

[51] Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, N. De Freitas, Dueling network architectures for deep reinforcement learning, arXiv:1511.06581, 2015.

[52] R.S. Sutton, A.G. Barto, R.J. Williams, Reinforcement learning is direct adaptive optimal control, IEEE Control Syst. Mag. 12 (2) (1992) 19–22.

[53] S.R. Kalidindi, C.A. Bronkhorst, L. Anand, Crystallographic texture evolution in bulk deformation processing of fcc metals, J. Mech. Phys. Solids 40 (3) (1992) 537–569.

[54] X. Ling, M. Horstemeyer, G. Potirniche, On the numerical implementation of 3d rate-dependent single crystal plasticity formulations, Int. J. Numer. Methods Eng. 63 (4) (2005) 548–568.

[55] R. Quey, P. Dawson, F. Barbe, Large-scale 3d random polycrystals for the finite element method: generation, meshing and remeshing, Comput. Methods Appl. Mech. Eng. 200 (17–20) (2011) 1729–1745.

[56] R. Quey, A. Villani, C. Maurice, Nearly uniform sampling of crystal orientations, J. Appl. Crystallogr. 51 (4) (2018) 1162–1173.

[57] Y. Rubner, C. Tomasi, L.J. Guibas, The earth mover's distance as a metric for image retrieval, Int. J. Comput. Vis. 40 (2000) 99–121.

[58] M. Cuturi, Sinkhorn distances: lightspeed computation of optimal transport, in: C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems, vol. 26, Curran Associates Inc., 2013.

[59] D.Q. Huynh, Metrics for 3D rotations: comparison and analysis, J. Math. Imaging Vis. 35 (2) (2009) 155–164.

[60] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, Pytorch: An Imperative Style, High-Performance Deep Learning Library, Advances in Neural Information Processing Systems, vol. 32, Curran Associates, Inc., 2019, pp. 8024–8035.

[61] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, J. Mach. Learn. Res. 13 (10) (2012) 281–305.

[62] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the 13th International Conference on Artificial Intelligence and Statistics, in: JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[63] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, in: 3rd International Conference on Learning Representations, 2015.

[64] L. Prechelt, Early stopping-but when?, in: Neural Networks: Tricks of the Trade, Springer, 1998, pp. 55–69.

[65] C. Bauckhage, F. Beaumont, S. Müller, Ml2r coding nuggets: Hopfield nets for max-sum diversification, Tech. Rep., MLAI, University of Bonn, 2021.

[66] L. Morand, T. Iraki, D. Helm, Crystallographic texture-property data set originating from a simulated multi-step metal forming process, Fraunhofer Fordatis repository, https://fordatis.fraunhofer.de/handle/fordatis/319, 2023.

[67] L. Morand, J. Dornheim, J. Pagenkopf, D. Helm, Simulation of texture evolution for a multi-step metal forming process, Fraunhofer Fordatis repository, https://fordatis.fraunhofer.de/handle/fordatis/201.2, 2021.