Available online at www.sciencedirect.com
SCIENCE DIRECT®
Acta Materialia 53 (2005) 1015–1027
Acta MATERIALIA
www.actamat-journals.com

On the synergy between texture classification and deformation process sequence selection for the control of texture-dependent properties

Veeraraghavan Sundararaghavan, Nicholas Zabaras *

Materials Process Design and Control Laboratory, Sibley School of Mechanical and Aerospace Engineering, 188 Frank H.T. Rhodes Hall, Cornell University, Ithaca, NY 14853-3801, USA

Received 19 August 2004; received in revised form 1 November 2004; accepted 1 November 2004
Available online 8 December 2004

Abstract

Because of the computational complexity involved in multi-length scale formulations involving polycrystal plasticity, innovative algorithms need to be incorporated in techniques for designing processes to realize materials with optimized properties. This paper demonstrates the synergy between classification of fcc polycrystal texture and multi-scale process design for achieving desired properties in such materials. The inverse problem of designing processing stages that lead to a desired texture or texture-dependent property is addressed by mining a database of orientation distribution functions (ODFs). Given a desired ODF, the hierarchical classifier matches its ODF features in the form of pole density functions of important orientation fibers to a class of textures in the database. Texture classes are affiliated with processing information; hence, enabling identification of multiple process paths that lead to a desired texture. The process parameters identified by the classifier are fine-tuned using a gradient optimization algorithm driven by continuum sensitivity analysis of texture evolution. An adaptive reduced-order model for texture evolution based on proper orthogonal decomposition in which the reduced ODF modes corresponding to the intermediate stages of the design process are adaptively selected from the database is employed.

Keywords: Texture library; Materials-by-design; Data-mining; Adaptive proper orthogonal decomposition

1. Introduction

Many engineering materials are polycrystalline in nature and the presence of crystallographic texture affects several important physical properties. During deformation processes, the crystallographic slip and lattice rotation are the primary means of plastic deformation in such materials. By controlling the deformation, it is possible to control texturing and design microstructures with highly optimized directional material properties. Deformation process design for control of microstructure sensitive properties involves the development of a multi-scale virtual environment where it would be possible to design the required process sequence and the macroscopic process parameters (e.g. die and preform shapes, forging velocities, etc.). Computational complexity associated with such algorithms necessitates development of tools that can accelerate materials design.

Very few published works in literature discuss design of textures leading to stipulated material performance requirements (materials by design/microstructure-sensitive design). Significant contributions include [1]

* Corresponding author. Tel.: +1 607 255 9104; fax: +1 607 255 1222.
E-mail address: zabaras@cornell.edu (N. Zabaras).
URL: http://www.mae.cornell.edu/zabaras.

1359-6454/$30.00
doi:10.1016/j.actamat.2004.11.001

where the authors discuss the design of a compliant beam so as to maximize the deflection without plastically deforming the beam. In [2], the authors design the optimal microstructure for a thin orthotropic plate with a circular hole subjected to an in-plane tensile load so as to maximize the load carrying capacity of the plate while avoiding plastic deformation. These analyses employ the reduced spectral series representation of the ODF that provide global support, requiring a significant number of terms in the series to obtain accurate representation of sharp textures. Alternatively, in [3,4] a reduced representation of ODF using the method of proper orthogonal decomposition was employed using the finite element representation of the ODF over an explicit discretization of the orientation space [5,6]. Applications of the technique for the control of properties such as yield strength and R-value at a material point were demonstrated in [3] through the design of appropriate deformation processes.

In the present paper, we extend the methodologies presented in [3,4] and formulate a new data-mining method and an adaptive reduced-order optimization approach for the design of appropriate processing sequences that lead to desired properties. ODFs from experiments or direct simulations of texture evolution are stored within a database from which processing routes leading to desired properties can be identified through data-mining. Class hierarchies of ODFs are created based on features in the form of pole density functions over prominent fiber families [5,7] in the fundamental region. Several processing paths are associated with each class of textures, enabling identification of multiple processing sequences that can lead to the desired properties. Once the processing sequences and associated parameters are identified through classification, fine-tuning of the parameters is performed through a reduced-order gradient-based optimization approach. As demonstrated in [3], reduced-order representation of the ODF results in reduction in degrees of freedom in the representation of texture and appreciable computational gains in the control of texture. However, one needs to select an ODF basis that also represents the new physical mechanisms encountered in the intermediate stages of the control problem. Such a basis is selected from the existing ODFs in the database using the adaptive basis approach [8].

The rest of the paper is organized as follows. In Section 2, the representation of fcc texture in Rodrigues space is defined and the concept of orientation fibers is introduced. In Section 3, a texture classification framework is introduced. In Section 4, a texture evolution model based on reduced representation of texture is discussed followed by the process design methodology in Section 5. Finally, Section 6 presents applications of the methodology.

2. Texture representation and feature extraction

2.1. Representation of fcc texture in Rodrigues–Frank space

The orientation distribution function (ODF) is employed for the quantification of crystallographic texture [9,10]. Various methods are available for computing the ODF evolution during deformation processing [11]. Texture evolution methodologies use parameterizations for the crystal lattice rotation which together with the crystal symmetry define the problem domain. We employ the axis-angle parametrization of the orientation space proposed by Rodrigues [5]. The Rodrigues parametrization is created by scaling the axis of rotation as $\mathbf{r} = \mathbf{n}\tan\left(\frac{\theta}{2}\right)$. A proper rotation $\mathbf{R}$ relates the lattice orientation to a reference orientation. Given the Rodrigues parametrization $\mathbf{r}$, the rotation $\mathbf{R}$ can be obtained as,

$$\mathbf{R} = \frac{1}{1+\mathbf{r}\cdot\mathbf{r}}(\mathbf{I}(1-\mathbf{r}\cdot\mathbf{r}) + 2(\mathbf{r}\otimes\mathbf{r} + \mathbf{I}\times\mathbf{r})). \tag{1}$$

The fundamental region represents a region of the orientation space such that each crystal orientation is represented uniquely within the space. Fundamental region for the cubic symmetry group results in a truncated cube. The planes that form the faces of the cube are introduced by symmetry rotations about $\langle 1\,0\,0\rangle$ family of axes and the corners are truncated by planes introduced by rotations about the $\langle 1\,1\,1\rangle$ axes. The ODF $\mathscr{A}$ describes the local density of crystals over this fundamental region of orientation space. The volume fraction of crystals within a part $\mathfrak{R}^*$ of the fundamental region is given by $v_f(\mathfrak{R}^*) = \int_{\mathfrak{R}^*}\mathscr{A}\,\mathrm{d}v$.

2.2. Classification for identifying processes that lead to desired ODF

The ODF classification framework creates a self-organizing database of textures from which relationships between processes and textures can be identified. The task of the classifier is to identify a class of textures (and associated process parameters) that may result in a desired ODF. Apart from using numerical control algorithms, the underlying inverse problem can also be approached using pattern recognition approaches over large and comprehensive databases. Direct classification of the ODFs using the finite element representation (the nodal values) over such databases are not computationally feasible due to high-dimensionality of the data set. We employ the approach in [12,13] where classification is carried out over a hierarchy of classes using lower-order microstructural features. The lower order features for the ODF are determined in the form of pole-density functions of important orientation fibers in the fundamental region and are used to create the class hierarchy. These features are extracted from a given desired ODF

and pattern recognition is employed to propagate the information over the existing class hierarchy to identify the required texture class and process parameters. The significance of the lower order features employed is briefly explained in the following sub-section.

2.2.1. Orientation fibers

A $\langle \mathbf{h} \rangle$ fiber about sample axis $\mathbf{y}$ connects orientations that align the crystal $\mathbf{h}$ axis with the sample $\mathbf{y}$ direction. The rotation $\mathbf{r}$ required to align $\mathbf{h}$ with $\mathbf{y}$ is based on a rotation of $\mathbf{h}$ through an angle $\phi = \cos^{-1} \mathbf{h} \cdot \mathbf{y}$ about axis $\mathbf{h} \times \mathbf{y}$. Note that $\mathbf{h}$ and $\mathbf{y}$ remain aligned even if the orientations change due to rotations about $\mathbf{h}$ or $\mathbf{y}$ axis. These orientations define the orientation fiber. In the Euler angle space, the fibers are curves described by trigonometric functions. Orientation fibers reduce to straight lines over Rodrigues' space. The orientations along the fiber over Rodrigues space is obtained by varying the parameter λ in the following equation,

$$\mathbf{r} = \frac{1}{1 + \mathbf{h} \cdot \mathbf{y}} (\mathbf{h} \times \mathbf{y} + \lambda(\mathbf{h} + \mathbf{y})). \tag{2}$$

Here, $\lambda = \tan(\phi + \bar{\phi})/2$ where ϕ and $\bar{\phi}$ are arbitrary, corresponding to rotations about $\mathbf{h}$ and $\mathbf{y}$, respectively. Fiber textures form when crystal flow during deformation is channeled along particular orientation fibers. Certain families of fibers are of particular importance in fcc textures. For example, the $\langle 1\ 1\ 0 \rangle$ family of fibers appear under uniaxial compression, plane strain compression and simple shear. fcc metals are typically associated with texturing to $\langle 1\ 1\ 1 \rangle$ and $\langle 1\ 0\ 0 \rangle$ fibers under tension and $\langle 1\ 1\ 0 \rangle$ under compression. In torsion tests, the z-axis $\langle 1\ 1\ 1 \rangle$ fibers and x-axis $\langle 1\ 1\ 0 \rangle$ are seen to predominate. The texturing of fcc metals under plane strain compression is dominated by the α fiber ND $\langle 1\ 1\ 0 \rangle$ connecting the ideal Goss and brass orientations, and the β fiber connecting the brass, S, and copper orientations [5].

For a particular $\mathbf{h}$, the pole figure takes values $\mathscr{P}(\mathbf{h},\mathbf{y})$ at locations $\mathbf{y}$ on a unit sphere. The pole density function, $\mathscr{P}(\mathbf{h},\mathbf{y})$ gets contributions from orientations for which the mapping $\mathbf{r}$ brings $\pm\mathbf{h}$ (or symmetric equivalent) into alignment with the sample axis $\mathbf{y}$ as, $\mathbf{R} \cdot \mathbf{h} = \hat{\mathbf{h}}, \hat{\mathbf{h}} \| \mathbf{y}$. Using the crystal symmetries in the ODF, the expression for pole density function can be obtained as [7],

$$\mathscr{P}(\mathbf{h},\mathbf{y}) = \tfrac{1}{2}[\mathscr{P}^0(\mathbf{h},\mathbf{y}) + \mathscr{P}^0(-\mathbf{h},\mathbf{y})], \tag{3}$$

where, $\mathscr{P}^0(\mathbf{h},\mathbf{y})$ represents a path integral given as,

$$\mathscr{P}^0(\mathbf{h},\mathbf{y}) = \frac{1}{2\pi} \int_{\hat{\mathbf{h}} \| \mathbf{y}} \mathscr{A}\, \mathrm{d}\theta. \tag{4}$$

The integration is performed over all fibers in the fundamental region corresponding to crystal direction $\mathbf{h}$ and sample direction $\mathbf{y}$. Given the finite element discretization of the fundamental region, integration is done by tracking the fiber through each finite element. Within a finite element, $\mathscr{A}$ is interpolated using the element shape functions and the nodal point values associated with the element. The vector of all independent nodal values is represented by A^{np}. The pole density function $\mathscr{P}(\mathbf{h},\mathbf{y})$ of an orientation fiber family $\mathbf{h}$ is found over a sample direction $\mathbf{y}$ using a system vector $m(\mathbf{h},\mathbf{y})$, computed using Eqs. (3) and (4) through a vector dot product as,

$$\mathscr{P}(\mathbf{h},\mathbf{y}) = m(\mathbf{h},\mathbf{y})^{\mathrm{T}} A^{np}. \tag{5}$$

The feature vector $\mathbf{x}_i$ for the ith ODF in the database at level l in the classification scheme is found as follows. The level l is associated with a particular fiber family $\mathbf{h}$ and the pole density functions are calculated at various values of $\mathbf{y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_m]$ as, $\mathbf{x}_i = \mathbf{M} A_i^{np}$ where the system matrix $\mathbf{M}$ is formed as $\mathbf{M} = [m(\mathbf{h},\mathbf{y}_1)^{\mathrm{T}}; m(\mathbf{h},\mathbf{y}_2)^{\mathrm{T}}; \ldots; m(\mathbf{h},\mathbf{y}_m)^{\mathrm{T}}]$.

The system matrices for the fiber families used are calculated and stored prior to classification. The use of RF space provides several advantages for the ODF classification problem. The local basis used in RF space discretization captures sharp textures that are important features for classification. Another convenient property is that the integration paths for pole projections used for finding the features fall along straight lines in Rodrigues space. Moreover, due to the symmetry of RF space, textures take on a simple structure and most ideal orientations are present close to the boundaries of the fundamental region, providing an ease of interpretation of texture clusters.

Fig. 1 shows the classification scheme for textures based on pole density functions as lower-order features at various levels. An advantage of the clustering scheme

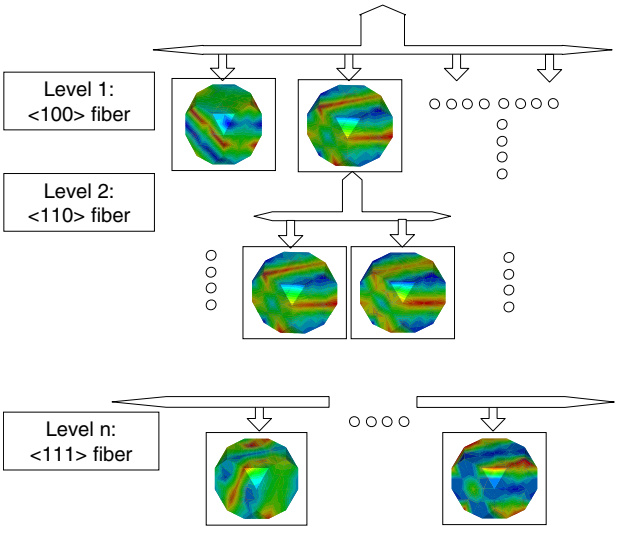

Fig. 1. The classification hierarchy for ODFs. The feature vector contains the pole density functions at different sample directions for the family of fibers specified at each classification level.

is its ability to capture non-uniqueness in the process-design solutions. Through classification, identification of several processing paths that can lead to a desired texture is made possible.

3. Classifier model

The unsupervised classification problem aims to unearth the relationships between a set of data without the need for any user-defined data in the form of class labels. Given a data-set D consisting of n features of the ODF, $\mathbf{x}_i$, $i = 1, \ldots, n$, with each feature attribute being a vector of m values as $\mathbf{x}_i^{\mathrm{T}} = \{x_{1i}, \ldots, x_{mi}\}$, $\mathbf{x}_i \in \Re^m$, the unsupervised classification problem is posed as follows:

Find the cluster centers $\{\mathbf{C}^1, \mathbf{C}^2, \ldots, \mathbf{C}^k\}$ *in* $\Re^m$ *such that the sum of the 2-norm distance squared between each feature* $\mathbf{x}_i$ *and its nearest cluster center* $\mathbf{C}^h$ *is minimized.*

The clustering problem can be written using the above mentioned 'distortion measure' as the problem of finding the cluster centers $\{\mathbf{C}^1, \mathbf{C}^2, \ldots, \mathbf{C}^k\}$ so that the cost function J is minimized,

$$J(\mathbf{C}^1, \ldots, \mathbf{C}^k) = \sum_{i=1}^{n} \min_{h=1,\ldots,k} \left(\tfrac{1}{2}\|\mathbf{x}_i - \mathbf{C}^h\|_2^2\right). \tag{6}$$

3.1. K-means clustering algorithm

The cluster center $\mathbf{C}^i$ needs to be the centroid of the closest set of features $\mathbf{x}_i$ for the distortion to be the minimum. Given a database D of n points in $\Re^m$ and cluster centers $\{\mathbf{C}^{1,i}, \mathbf{C}^{2,i}, \ldots, \mathbf{C}^{k,i}\}$ in $\Re^m$ at iteration i, the Lloyd's algorithm computes the cluster centers, $\{\mathbf{C}^{1,i+1}, \ldots, \mathbf{C}^{k,i+1}\}$ at iteration $i+1$ in the following 2 steps [14]:

1. *Cluster assignment*: For each data $\mathbf{x}_i$, assign $\mathbf{x}_i$ to cluster $h(i)$ such that center $\mathbf{C}^{h(i),i}$ is nearest to $\mathbf{x}_i$ in the 2-norm.
2. *Cluster update*: Compute $\mathbf{C}^{h(i),i+1}$ as the centroid of all points assigned to cluster h.

The algorithm is stopped when $\mathbf{C}^{h,i+1} = \mathbf{C}^{h,i}$, $h = 1, \ldots, k$, otherwise i is incremented by 1 and steps 1 and 2 are repeated. At the start of the algorithm, the k cluster centers, $\{\mathbf{C}^{1,0}, \mathbf{C}^{2,0}, \ldots, \mathbf{C}^{k,0}\}$ are randomly initialized. The nature of distortion minimization within the k-means algorithm makes the clusters hyper-spherical. The cluster center solutions produced depend on these initial values, and bad initial guesses may result in sub-optimal partitioning. The standard solution is to try several starting configurations. The results of the k-means algorithm depends on the number of clusters k required to be provided by the user. For the ODF classification problem, the number of classes are not known a-priori. We employ the 'x-means' algorithm [15] for discovering the actual number of classes that exist in the set of ODFs. Given the data-set D, the model chosen maximizes the Bayesian information criterion (BIC) given as,

$$\mathrm{BIC} = \hat{l}(D) - \frac{p}{2}\log(n), \tag{7}$$

where, $\hat{l}(D)$ is the log-likelihood of the data taken at the maximum likelihood point, p is the number of free parameters in the model, $p = m \cdot k + k - 1 + k$, consisting of $m \cdot k$ cluster center coordinates, $k - 1$ class probabilities and k variance estimates. The maximum likelihood estimate for the variance, assuming spherical-Gaussian distribution of data within a cluster i consisting of r_i data points $(\mathbf{x}_j^i, j = 1, \ldots, r_i)$ is given as,

$$\hat{\sigma}_i^2 = \frac{1}{r_i - 1}\sum_{j=1}^{r_i} \|\mathbf{x}_j^i - \mathbf{C}^i\|^2. \tag{8}$$

The probabilities of each point within the cluster i is given as,

$$\hat{P}(\mathbf{x}_j^i) = \frac{r_i}{n}\frac{1}{\sqrt{2\pi}\hat{\sigma}_i{}^m} \exp\left(-\frac{1}{2\hat{\sigma}_i{}^2}\|\mathbf{x}_j^i - \mathbf{C}^i\|^2\right). \tag{9}$$

The log-likelihood of all the data within the cluster is given as,

$$\begin{aligned} l(D_i) &= \log \prod_{j=1}^{r_i} P(\mathbf{x}_j^i) \\ &= \sum_{j=1}^{r_i}\left(\log\left(\frac{1}{\sqrt{2\pi}\hat{\sigma}_i{}^m}\right) - \frac{1}{2\sigma_i^2}\|\mathbf{x}_j^i - \mathbf{C}^i\|^2 + \log\frac{r_i}{n}\right). \end{aligned} \tag{10}$$

Hence, at the maximum likelihood estimate, the log-likelihood of the data belonging to cluster i is given as,

$$\begin{aligned} \hat{l}(D_i) = &-\frac{r_i}{2}\log(2\pi) - \frac{r_i \cdot m}{2}\log(\hat{\sigma}_i{}^2) - \frac{r_i - 1}{2} \\ &+ r_i\log(r_i) - r_i\log(n). \end{aligned} \tag{11}$$

The log-likelihood of the entire data set is the sum of the log-likelihoods of all clusters, hence, the BIC (Eq. (7)) for the entire data-set can be written as,

$$\begin{aligned} \mathrm{BIC} = &-\frac{n}{2}\log(2\pi) - \frac{m}{2}\sum_{i=1}^{k} r_i\log(\hat{\sigma}_i{}^2) - \frac{n-k}{2} \\ &+ \sum_{i=1}^{k} r_i\log(r_i) - n\log(n) - \frac{(m+2)k - 1}{2}\log(n). \end{aligned} \tag{12}$$

In the x-means algorithm, the Bayesian information criterion is tested for configurations arising from different values of k, and the best configuration is chosen. Convergence properties are further improved in the algorithm by letting few cluster centers (parent clusters) obtained from the k-means step to split further into

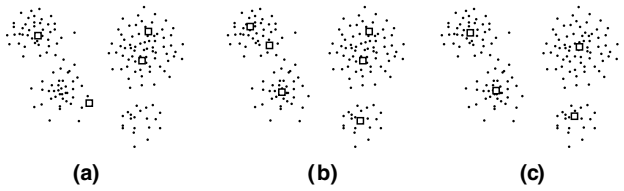

Fig. 2. Results of the x-means and k-means algorithm on a 2D feature set. The squares represent the cluster centers. (a) Clustering using k-means: local optimum produced by the k-means algorithm ($k = 4$), (b) clustering using k-means with number of classes fixed at $k = 6$, (c) four clusters identified by the x-means algorithm.

two centers. This is performed through local k-means operation on the parent cluster using two new centers. The BIC measure is then tested locally within the parent cluster. The parent cluster is retained only if the BIC degrades due to the splitting operation. Fig. 2 shows a comparison of the x-means and the k-means methodologies for a two-dimensional feature set. Fig. 2(a) shows a configuration produced by the k-means algorithm with number of classes k given as 4. The configuration has converged but has not produced the distortion corresponding to the global minimum. Fig. 2(b) shows the k-means results with number of classes $k = 6$. With k-means, it is not possible to extract the true clustering in the data-set. Higher the number of classes, lesser is the distortion but the possibility of overfitting the data increases. Fig. 2(c) shows the cluster centers identified by the x-means algorithm. Based on the BIC measure, x-means identified the four natural clusters in the data-set.

4. Texture evolution model

The evolution of the ODF is governed by the ODF conservation equation. The conventional Eulerian rate form of the conservation equation is given by [5]:

$$\frac{\partial \mathscr{A}(\mathbf{r},t)}{\partial t} + \nabla \mathscr{A}(\mathbf{r},t) \cdot \mathbf{v}(\mathbf{r},t) + \mathscr{A}(\mathbf{r},t) \nabla \cdot \mathbf{v}(\mathbf{r},t) = 0, \quad (13)$$

where $\mathbf{v}(\mathbf{r},t)$ is the Eulerian reorientation velocity. The polycrystal average of an orientation-dependent property, $\chi(\mathbf{r},t)$, is determined using the Eulerian ODF by an integral over the fundamental region:

$$\langle \chi \rangle = \int_{\mathscr{R}} \chi(\mathbf{r},t) \mathscr{A}(\mathbf{r},t)\, \mathrm{d}v. \quad (14)$$

A desired property $\langle \chi \rangle$ distribution at the material point can hence be obtained by controlling the ODF ($\mathscr{A}$). From Eq. (13), it is seen that the evolution of the ODF $\mathscr{A}$ is controlled by the reorientation velocity $\mathbf{v}(\mathbf{r},t)$. Hence, a desired property distribution can be obtained through control of the macro-design variable, namely the velocity gradient, $\mathbf{L}$, which is linked with $\mathbf{v}(\mathbf{r},t)$ using the extended Taylor macro–micro linking hypothesis. The hypothesis equates the crystal velocity gradient and the macro velocity gradient $\mathbf{L}$.

The reorientation velocity is then evaluated through crystal constitutive relations, which involve the crystal velocity gradient. The velocity gradient of a crystal with orientation, $\mathbf{r}$, yields the following form [5]:

$$\boldsymbol{L} = \boldsymbol{\Omega} + \boldsymbol{R} \sum_{\alpha} \dot{\gamma}^{\alpha} \bar{\boldsymbol{T}}^{\alpha} \boldsymbol{R}^{\mathrm{T}}, \quad (15)$$

where $\boldsymbol{\Omega}$ is the lattice spin, $\dot{\gamma}^{\alpha}$ is the shearing rate along the slip system α and $\bar{\boldsymbol{T}}^{\alpha}$ is the Schmid tensor for the slip system α, given by $\bar{\boldsymbol{m}}^{\alpha} \otimes \bar{\boldsymbol{n}}^{\alpha}$, where $\bar{\boldsymbol{m}}^{\alpha}$ is the slip direction and $\bar{\boldsymbol{n}}^{\alpha}$ is the slip plane normal, both in the crystal lattice frame. The expressions for the spin and symmetric parts are obtained as shown below:

$$\boldsymbol{\Omega} = \boldsymbol{W} - \sum_{\alpha} \dot{\gamma}^{\alpha} \boldsymbol{R} \bar{\boldsymbol{Q}}^{\alpha} \boldsymbol{R}^{\mathrm{T}}, \quad (16)$$

$$\bar{\boldsymbol{D}} = \sum_{\alpha} \dot{\gamma}^{\alpha} \bar{\boldsymbol{P}}^{\alpha}, \quad (17)$$

where $\bar{\boldsymbol{P}}^{\alpha}$ and $\bar{\boldsymbol{Q}}^{\alpha}$ are the symmetric and skew parts of the Schmid tensor respectively and $\bar{\boldsymbol{D}}$ is the macroscopic deformation rate expressed in the lattice frame through, $\bar{\boldsymbol{D}} = \boldsymbol{R}^{\mathrm{T}} \boldsymbol{D} \boldsymbol{R}$. The shearing rate on slip systems is given by a power law and it is further assumed that all slip systems have identical hardness.

$$\dot{\gamma}^{\alpha} = \dot{\gamma}^{0} \left| \frac{\tau^{\alpha}}{s} \right|^{1/m} \operatorname{sign}\left(\frac{\tau^{\alpha}}{s} \right), \quad (18)$$

where s is the slip system hardness, m is the strain rate sensitivity, $\dot{\gamma}^{0}$ is a reference rate of shearing and τ^{α} is the resolved shear stress on slip system α. Further, the resolved stress is related to the crystal Cauchy stress as

$$\tau^{\alpha} = \bar{\boldsymbol{\sigma}} \cdot \bar{\boldsymbol{P}}^{\alpha}. \quad (19)$$

By solving the system of Eqs. (17)–(19), the crystal Cauchy stress $\bar{\sigma}$ and the shear rate $\dot{\gamma}^{\alpha}$ can be evaluated. Next, using Eq. (16), one can evaluate the lattice spin vector as,

$$\omega = \operatorname{vect}(\boldsymbol{\Omega}), \quad (20)$$

which is then used to evaluate the reorientation velocity as,

$$\mathbf{v} = \tfrac{1}{2}(\omega + (\omega \cdot \mathbf{r})\mathbf{r} + \omega \otimes \mathbf{r}). \quad (21)$$

Finally, the ODF, $\mathscr{A}$, over the current fundamental region $\mathscr{R}$ is evaluated from the Eulerian form (Eq. (13)) of the conservation equation. The full- and reduced-order methodologies for solving Eq. (13) are given in the next subsection.

4.1. Full and reduced-order approaches

Eq. (13) has the form of the advective transport equation and is subject to discontinuities in the velocity

divergence. The finite element formulation (full-order model) involves SUPG stabilization and takes the following form:

$$\int_{\mathscr{R}} \left\{ \frac{\partial \mathscr{A}}{\partial t} + \nabla \mathscr{A} \cdot \mathbf{v} + \mathscr{A} \nabla \cdot \mathbf{v} \right\} w \, \mathrm{d}v + \int_{\mathscr{R}_t} \nabla \cdot (\epsilon \nabla \mathscr{A}) \psi \, \mathrm{d}v = 0, \tag{22}$$

where w and ψ are the Petrov–Galerkin and classical Galerkin weighting functions respectively and ϵ is the shock capturing parameter.

In multi-scale deformation problems where several ODFs need to be controlled, the full-order model requires considerable computational resources due to large number of degrees-of-freedoms needed for the analysis as well as the associated mathematical and computational complexity. The proper orthogonal decomposition (POD) is a popular reduced-order modeling approach for decreasing the computational burden in such problems. Reduced-order modeling is based on the development of a reduced set of basis functions, $\phi(\mathbf{r})$, to represent the associated ODF. The method of snapshots is introduced for generating the basis. It assumes that the basis ϕ, for a given deformation process, can be expressed as a linear combination of the N ODF snapshots at different time-steps from the full-order model as:

$$\phi_j = \sum_{i=1}^{N} u_i^j \mathscr{A}^{(i)}, \tag{23}$$

where u_i^j can be determined by the solving the eigenvalue problem:

$$\boldsymbol{C}\ \boldsymbol{U} = \boldsymbol{\Lambda}\ \boldsymbol{U}, \tag{24}$$

where, $\boldsymbol{C}$ is the spatial correlation matrix defined as,

$$\boldsymbol{C}_{i,j} = \frac{1}{N} \int_{\mathscr{R}} \mathscr{A}^{(i)}(\mathbf{r}) \mathscr{A}^{(j)}(\mathbf{r}) \, \mathrm{d}v \tag{25}$$

and $\boldsymbol{\Lambda}$ and $\boldsymbol{U}$ are the complete eigen-description of the system. Once the modes have been evaluated, Eq. (23) is used to generate the basis for reduced-order modeling, such that at any processing stage j,

$$\mathscr{A}^{(j)}(\mathbf{r},t) = \sum_{i=1}^{M^{(j)}} a_i^{(j)}(t) \phi_i^{(j)}(\mathbf{r}), \tag{26}$$

where $M^{(j)}$ is the number of modes used in stage j, and $a_i^{(j)}$ are the reduced-order coefficients used for representing the ODFs in stage j. Using this approximation in the weak form of the ODF conservation equation results in the following ordinary differential equation (ODE):

$$\dot{\mathbf{a}} = \mathscr{R}\mathbf{a}, \tag{27}$$

where,

$$\mathscr{R}_{i,j} = -\int_{\mathscr{R}} (\nabla \phi_j \cdot \mathbf{v} \phi_i + \phi_j \phi_i \nabla \cdot \mathbf{v}) \mathrm{d}v. \tag{28}$$

Eq. (27) is solved over n timesteps (0 to t_n) at each stage. At any stage j, $j = 1, \ldots, p$, the initial value of a is given through the following equation,

$$a_i^{(j)}(0) = \int_{\mathscr{R}} \mathscr{A}^{(j-1)}(\mathbf{r},t_n) \phi_i^{(j)}(\mathbf{r}) \mathrm{d}v, \tag{29}$$

where, $\phi^{(j)}(\mathbf{r})$ is the reduced basis at stage j. The initial microstructure is assumed to be random and taken as $\mathscr{A}^{(1)}(\mathbf{r},0) = 2.435$. Eqs. (27)–(29) define the reduced-order model for the ODF conservation equation. Thus, to obtain a desired ODF, only a small finite number of degrees of freedom (i.e. the vector a) needs to be controlled. Further, compared to the full order model, the reduced model does not require any stabilizing modifications.

5. Design of processes

The objective of the microstructure-sensitive design process is to control the properties in the micro-scale through design of appropriate deformation processes. The direct problem described in Section 4 simulates the ODF evolution given the macro velocity gradient. The process design methodology aims to identify the macro velocity gradient that yields a desired ODF (or desired property distribution). Given a good initial guess, gradient based methods converge to a local optimum within a few iterations. Intelligent choice of initial guesses can be made using prior information available in the form of a database through classification. Refer to [3,4] for details on the implementation of the design problem for a single stage using gradient-based approach. The following section addresses the extension of the technique to a multi-stage design process.

5.1. Multi-stage design process

Let us denote the sensitivity of the ODF to a small change in the process parameter $\boldsymbol{\alpha}$ as $\overset{\circ}{\mathscr{A}} = \overset{\circ}{\mathscr{A}}(\mathbf{r},t;\boldsymbol{\alpha},\Delta\boldsymbol{\alpha})$. Design-differentiation of Eq. (13) results in the following:

$$\frac{\partial \overset{\circ}{\mathscr{A}}}{\partial t} + \nabla \overset{\circ}{\mathscr{A}} \cdot \mathbf{v} + \nabla \mathscr{A} \cdot \overset{\circ}{\mathbf{v}} + \overset{\circ}{\mathscr{A}} \nabla \cdot \mathbf{v} + \mathscr{A} \nabla \cdot \overset{\circ}{\mathbf{v}} = 0. \tag{30}$$

This equation can be solved to develop the sensitivity of the ODF field assuming that the sensitivity of the reorientation velocity and its divergence are known. This is evaluated in the constitutive sensitivity problem [3,4].

In the reduced-order sensitivity problem, we utilize the basis developed earlier for the direct problem and approximate the sensitivity fields as linear combinations of these basis functions. The computations are similar to

those performed for the direct analysis, and the reduced system is obtained as follows:

$$\dot{b} = \mathscr{G} b + \mathscr{H}, \tag{31}$$

where

$$\mathscr{G}_{i,j} = -\int_{\mathscr{R}} (\nabla \phi_j \cdot \mathbf{v} \phi_i + \phi_j \phi_i \nabla \cdot \mathbf{v}) \mathrm{d}v, \tag{32}$$

$$\mathscr{H}_i = -\int_{\mathscr{R}} (\nabla \mathscr{A} \cdot \mathring{\mathbf{v}}\, \phi_i + \mathscr{A} \phi_i \nabla \cdot \mathring{\mathbf{v}}) \mathrm{d}v. \tag{33}$$

For the first stage, $\mathring{\mathscr{A}}^{(1)}(\mathbf{r},0) = 0$. At the end of stage j, $\mathring{\mathscr{A}}^{(j)}(\mathbf{r},t_n)$ is calculated from the coefficients obtained from the solution of Eq. (31) at the final time step as,

$$\mathring{\mathscr{A}}^{(j)}(\mathbf{r},t_n) = \sum_{i=1}^{M^{(j)}} b_i^{(j)}(t_n) \phi_i^{(j)}(\mathbf{r}), \tag{34}$$

where $\phi^{(j)}(\mathbf{r})$ the reduced basis at stage j. The initial value of b for $(j+1)$th stage is given through the following equation,

$$b_i^{(j+1)}(0) = \int_{\mathscr{R}} \mathring{\mathscr{A}}^{(j)}(\mathbf{r},t_n) \phi_i^{(j+1)}\, \mathrm{d}v, \tag{35}$$

where $\phi^{(j+1)}(\mathbf{r})$ is the set of reduced basis employed for the sensitivity problem at stage $j+1$. Once sensitivity at the last time step of the final stage is found, expectation of the sensitivity of a property to a small change in the process parameter is found as,

$$\langle \mathring{\chi} \rangle = \int_{\mathscr{R}} \chi(\mathbf{r},t)\, \mathring{\mathscr{A}}^{(p)}(\mathbf{r},t_n) \mathrm{d}v. \tag{36}$$

The design variable, namely the macro-velocity gradient L, is written as follows:

$$\begin{aligned}\mathbf{L} = {} & \alpha_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & -0.5 & 0 \\ 0 & 0 & -0.5 \end{bmatrix} + \alpha_2 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \\ & + \alpha_3 \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \alpha_4 \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \\ & + \alpha_5 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} + \alpha_6 \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ & + \alpha_7 \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} + \alpha_8 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}. \end{aligned} \tag{37}$$

Each matrix in the above decomposition corresponds to a given deformation process namely tension/compression α_1, plane strain compression α_2, shear modes (α_3, α_4, α_5) and rotation modes (α_6, α_7, α_8).

We define the design problem of interest as the selection of the processing sequence, with stages involving tension/compression, plane strain compression, shear or rotation, and the corresponding process parameters $\boldsymbol{\alpha}$ that lead to a desired property Ω that is a function of the ODF. This can be stated as follows:

$$\min_{\boldsymbol{\alpha}} \mathscr{F}(\boldsymbol{\alpha}) = \frac{1}{N_s} \sum_{i=1}^{N_s} (\Omega^i(\mathscr{A}(\boldsymbol{\alpha})) - \Omega^{\mathrm{desired}^i})^2, \tag{38}$$

where N_s is the total number of sampling points, $\Omega^{\mathrm{desired}}$ is the discrete representation of the desired microstructural property and $\boldsymbol{\alpha}$ is the design parameter involved in the iterative optimization algorithm corresponding to the process parameter α_j from stage $j = 1$ to p. The calculation of sensitivities of property χ to a component α_j of α requires solution to sensitivity problems at $p - j + 1$ stages. The ith multi-stage sensitivity problem is driven by $\Delta\alpha_i = 10^{-2}$ with $\Delta\alpha_j = 0$ for $j \neq i$. The gradients of property χ with respect to α_i is calculated as

$$\frac{\partial \chi}{\partial \alpha_i} = \frac{\mathring{\chi}(\mathbf{r},t,\alpha_1,..,\alpha_p,0,..,\Delta\alpha_i,..,0)}{\Delta\alpha_i}. \tag{39}$$

The sensitivities are then used in a gradient descent algorithm to obtain the optimum process parameters that minimize the objective function in Eq. (38).

5.2. Adaptive reduced-order model

The classification technique is database-driven and the availability of existing information can be further utilized to accelerate the texture evolution models. The reduced-order model must fully represent not only the optimal solution but also all the intermediate solutions obtained during the optimization process [3,4]. This calls for a method where the bases for the ODF are chosen adaptively during the control algorithm. Following the method proposed in [8], at processing stage j, optimization step 1 begins with reduced-order modes $\phi^{(j,1)}(\mathbf{r})$ obtained from the database corresponding to an initial process parameter estimate up to stage j, $\boldsymbol{\alpha}^{(j,1)} = [\alpha^{(1,1)}, \ldots, \alpha^{(j,1)}]$ obtained through classification. These reduced-order modes are now used in the gradient optimization algorithm to find the new iterate in the second optimization step, $\boldsymbol{\alpha}^{(j,2)} = [\alpha^{(1,2)}, \ldots, \alpha^{(j,2)}]$. In general, the reduced-order basis corresponding to a new iterate $\boldsymbol{\alpha}^{(j,i+1)}$ at optimization step $(i+1)$ in stage j is found through the following steps:

1. Select the new reduced-order basis $\phi^{(j,i+1)}(\mathbf{r})$ from the existing database by searching for the closest parameter $\boldsymbol{\beta}^{\boldsymbol{D}}$ within a user-defined tolerance limit,

$$\|\boldsymbol{\beta}^{\boldsymbol{D}} - \boldsymbol{\alpha}^{(j,i+1)}\|_2 \leqslant \epsilon. \tag{40}$$

2. If $\|\boldsymbol{\beta}^{\boldsymbol{D}} - \boldsymbol{\alpha}^{(j,\,i+1)}\|_2 > \epsilon$, then compute the snapshots corresponding to $\boldsymbol{\alpha}^{(j,i+1)}$, generate the new reduced basis $\phi^{(j,i+1)}(\mathbf{r})$ from the snapshots and update the existing database with the new $\boldsymbol{\alpha}^{(j,i+1)}$ and $\phi^{(j,i+1)}(\mathbf{r})$.

Remark 1. An initial uniform texture is assumed at the first processing stage. The reduced-order basis corresponding to the pure deformation modes (obtained from an ensemble of data corresponding to a deformation $\alpha = [1]$, with appropriate mode used in the first stage) is found to be sufficient to represent the texturing in the first stage.

Remark 2. The sensitivity problem uses the same basis as the direct problem. However, a sensitivity problem of stage i (where process variable corresponding to stage i is perturbed) uses the stage i basis for the sensitivity problems in stages $i + 1$ to p.

Remark 3. Over large databases, the search procedure in step (1) of the adaptive reduced-order algorithm can be addressed efficiently using classification algorithms.

5.3. Effect of adaptive basis threshold

The difference between the reduced- and full-order control solutions depends on the sensitivity of desired property to the numerical error induced by the introduction of a reduced basis. Selection of the threshold parameter ϵ plays a critical role in the adaptive basis scheme described above. Small thresholds result in more accurate solutions but are computationally expensive due to frequent basis changes. Larger thresholds involve less frequent basis changes but may result in inaccurate solutions since the basis might not model the process employed accurately. Further, the sensitivities may be inaccurate leading to divergence in the objective function. Fig. 3 shows the increase in error caused with increasing values of ϵ used for the basis selected. The strain rate for the first stage is fixed and that of the second stage is increased which results in different values of ϵ. The ODF resulting from a basis with $\epsilon = 0$ after a time of 0.1 sec is used as the reference $\mathscr{A}^{\text{ref}}$. The error is defined as $100 \times \frac{\|\mathscr{A}^{\text{ref}} - \mathscr{A}\|_2}{\|\mathscr{A}\|_2}$. The error not only depends on the threshold but also on the types of processing stages involved. With tension as the second processing stage, changing the first stage to shear from plane strain compression results in about 30% increase in error at the same threshold. Within a processing sequence, however, the increase in error due to change in processing parameters is small. The results also indicate that the thresholds can be varied based on the processing sequence, a tension-plane strain compression processing sequence can have twice as much threshold than the shear-tension sequence with similar errors induced by the reduced-order approximation.

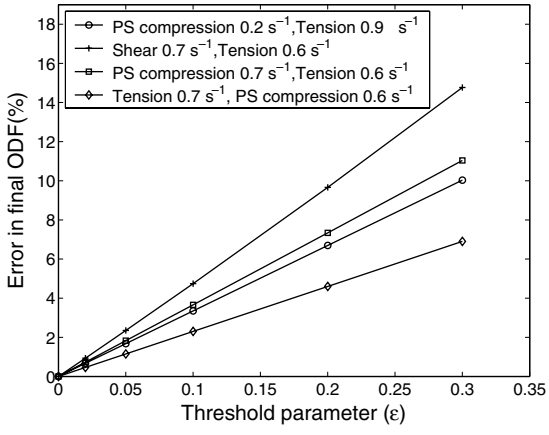

Fig. 3. Error induced due to different thresholds for the basis, the error not only depends on the threshold but also on the sequence of processing stages involved.

5.4. Database architecture

The database contains several data sets, each set corresponding to a particular process sequence and associated processing parameters. Every data set also contains a reduced ODF basis for the process that it represents. The final ODF in each data set is used in the classification scheme to identify classes of textures. The processing parameters that lead to a desired final texture are found by first identifying the class of texture to which the desired texture belongs. Required processing parameters and process sequences are then found from the data sets in the identified class. The best process path can then be selected from this set.

Let data set 'A' containing a processing sequence of tension and shear be found to result in a particular desired texture using the classifier. In a control problem involving the above 2 stages, a basis corresponding to the pure deformation mode (tension) is used for the first processing stage and is unchanged during the intermediate iterations of the control problem. For the second stage (shear), a basis of data set 'A' is initially used. If during an intermediate stage of the control problem, the process parameters for the tension and shear stages change beyond the allowed threshold ϵ, then the database is initially searched for a data set with process parameters within the allowed threshold. If such a data set is not available, a new data set is added to the database corresponding to the new process parameters. The basis for this data set is used in subsequent iterations of the design problem until the process parameters once again change above the selected threshold. Using this scheme, just three modes of the basis (with three unknowns) are found to adequately represent the texturing at any stage in an optimization step, enhancing the computational efficiency of the algorithm.

The success of the data-mining approach is limited to the amount of information in the database. Selection of good processing sequence solutions require a compre-

hensive database with data sets containing rich combination of processes. New information added to the database during the optimization process improves the possibility of identification of processing parameters and reduced bases directly from the database in future optimization runs.

6. Applications in materials design

A validation of the adaptive reduced-order approach is provided together with relevant design problems utilizing the classification framework. fcc materials with 12 slip systems are modeled in the examples. An initial uniform texture which corresponds to a value of $\mathscr{A}(\mathbf{r},0) = 2.435$ is assumed in the 448 element fundamental region. Relevant material constants used in the design examples are $\dot{\gamma}^{\alpha} = 1.0\ \mathrm{s}^{-1}$, $m = 0.05$ and $s = 27.17$ MPa [5]. The reduced-order basis for each design iteration is generated from an ensemble of data obtained from a deformation test for a time of 0.1 s with a time step of $\delta t = 0.01$ s. The first three modes of the selected reduced basis are used for modeling texture evolution at any deformation stage. For the examples, the time for which each deformation stage acts is fixed at 0.1 s. The optimization problem is executed until the objective function normalized with the initial objective showed less than 10^{-4} improvement between iterations.

A study of the adaptive reduced basis algorithm was conducted comparing reduced-order results with the full-order approach. A three stage test with each processing stage corresponding to a deformation test for a time of 0.1 s was performed. The stages employed were (1) Tension (strain rate: 0.8 s^{-1}); (2) Plane strain compression (strain rate: 0.2 s^{-1}); (3) Shear (strain rate: 0.2 s^{-1}). The full-order and reduced-order ODFs at the end of the three stages are shown in Fig. 4. In Fig. 5, the final reduced-order sensitivity of the ODF with a perturbation of 0.01 s^{-1} in the strain rate of the first stage is compared with the sensitivities obtained using (1) the full-order sensitivity problem and (2) finite difference method by perturbation of the full-order direct problem. The sensitivity of the first stage was transferred to the second stage and subsequently to the third stage. The basis of the first stage was utilized for all three stages of the sensitivity problem.

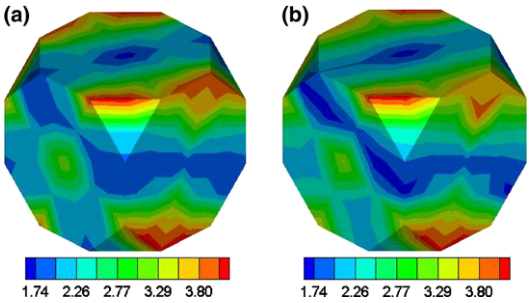

Fig. 4. Comparison of the ODF at the third and final stage obtained through (a) the full-order model, (b) the reduced-order model.

6.1. Design for desired ODF

The optimization problem involves designing the macro velocity gradient to obtain desired ODFs. Given the initial processing sequence and the parameters identified by the classifier, the reduced-order optimization scheme identifies the processing parameters that lead to the desired texture. As an example, the desired ODF shown in Fig. 6(a) was initially identified by the classifier to arise from a two-stage problem, with plane strain compression and compression modes respectively. The initial ODF corresponding to the strain rates for the two stages, 0.65 and $-0.1\ \mathrm{s}^{-1}$ respectively is shown in Fig. 6(b). The strain rates for the two processes after the adaptive reduced-order optimization procedure is obtained as 0.9472 and $-0.2847\ \mathrm{s}^{-1}$ respectively and the optimized ODF is shown in Fig. 6(c).

The advantage of the data-mining methodology lies in the identification of multiple processing paths that lead to a desired texture. Fig. 7(b) shows a class of textures with different processing routes that can result in a desired ODF. Given the desired ODF, the classifier uses the lower order features, namely, the pole

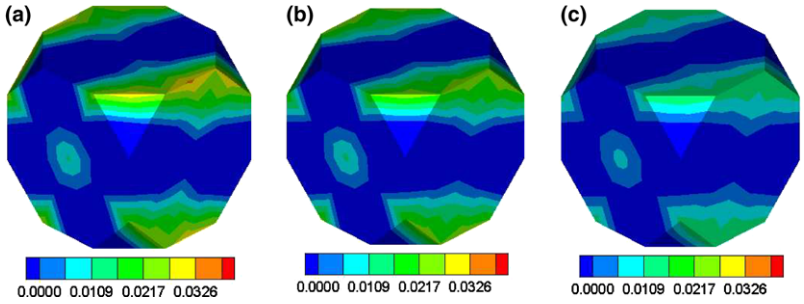

Fig. 5. Comparison of the sensitivity of the ODF at the third and final stage due to perturbation in the process parameter (α) of the first stage obtained using (a) the full-order model, (b) the reduced-order model and (c) FDM solution at the final stage ($t = 0.30$ s).

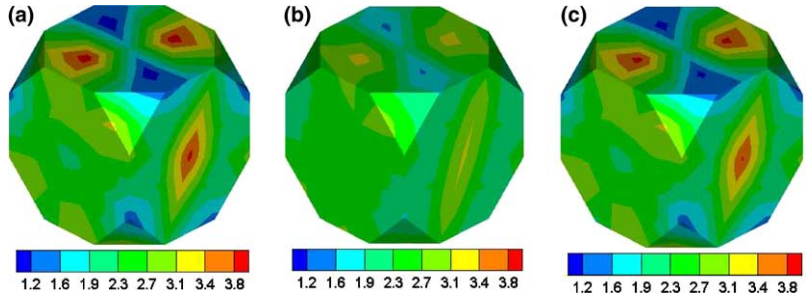

Fig. 6. Control of material texture: (a) the desired texture, (b) the initial guess identified by the classifier and (c) reduced-order optimized ODF.

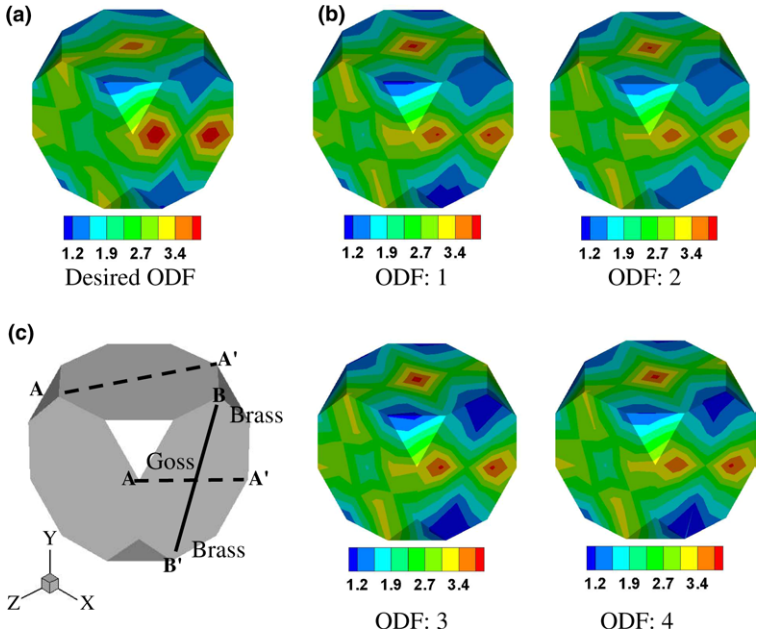

Fig. 7. ODF: 1, 2, 3, 4 represent a class of ODFs similar to the desired ODF in their lower-order features. Positions of z-axis $\langle 1\ 1\ 0\rangle$ (AA′) and alpha fibers (BB′) in the boundaries of the fundamental region are indicated in (c).

density functions, over 4 levels in the class hierarchy corresponding to the fibers in the $\langle 1\ 1\ 0\rangle$, $\langle 1\ 0\ 0\rangle$, $\langle 1\ 1\ 1\rangle$, and $\langle 2\ 1\ 1\rangle$ fiber families, respectively. The orientation fibers are chosen based on their particular importance in fcc textures (see Section 2.2.1) and their close affiliation with the processes involved. The desired texture in Fig. 7(a) is seen to be dominated by two fibers, the z-axis $\langle 1\ 1\ 0\rangle$ fiber and the alpha fiber (running from brass to the goss component) shown in Fig. 7(c). ODF intensities in the alpha fiber are associated with the process of plane strain compression, although the texture shows stronger development of the brass component relative to Goss. Texturing to the z-axis $\langle 1\ 1\ 0\rangle$ fiber is normally associated with fcc metals under compression along the z-axis. From the processing sequences identified (Table 1), we observe that these two processes are dominant in all the processing sequences found by the classifier.

6.2. Design for desired elastic modulus

The data-mining methodology can be extended to classification of ODFs based on material property distribution exhibited by the ODF. Given a desired texture-dependent property, the classification is performed based on the property feature (variation of the property as a function of angle from the rolling direction) for the ODFs in the database. The clustering scheme enables identification of ODFs and the corresponding processes that can reproduce a desired property distribution.

This example demonstrates the control of the velocity gradient of a sequence of processes in order to obtain a particular distribution of the elastic modulus of an fcc Copper polycrystal about the normal direction away from the rolling direction. The stiffness in the crystal is given (values in GPa) in the crystal lattice frame for crystals with cubic symmetry as follows:

Table 1
Process parameters of the ODF class in Fig. 7

ODF	Stage 1	Stage 2	Stage 3
1	PSC (-0.677 s^{-1})	Shear (-0.165 s^{-1})	Tension (-0.881 s^{-1})
2	Tension (-0.835 s^{-1})	PSC (-0.606 s^{-1})	
3	Tension (-0.917 s^{-1})	Shear (-0.074 s^{-1})	PSC (-0.760 s^{-1})
4	Tension (-0.907 s^{-1})	PSC (-0.669 s^{-1})	Rotation (0.179 s^{-1})

Through data mining methods, we obtain several possible processing sequences that may result in a desired ODF.

$$\boldsymbol{C} = \begin{bmatrix} 168.0 & 121.4 & 121.4 & 0 & 0 & 0 \\ 121.4 & 168.0 & 121.4 & 0 & 0 & 0 \\ 121.4 & 121.4 & 168.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 75.4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 75.4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 75.4 \end{bmatrix}.$$

The polycrystal stiffness, $\bar{\boldsymbol{C}}$, is computed through a weighted average (over $\mathscr{A}$) of the stiffness of individual crystals expressed in the sample reference frame. The elastic modulus is then computed through this polycrystal stiffness as

$$\boldsymbol{E} = \frac{1.0}{(\bar{\boldsymbol{C}})^{-1}_{(11)}}. \tag{41}$$

Furthermore, the elastic modulus about an angle with the rolling direction (RD) can be evaluated using the above equation, but after a coordinate transformation of $\bar{\boldsymbol{C}}$.

The classification scheme captures the non-uniqueness in process design, identifying several different textures (and processes) that might result in a desired property distribution. An example of a class of ODFs obtained from the database based on the Youngs modulus property variation from rolling direction to the transverse direction in the sample is shown in Fig. 8(b). The property distribution feature for a set of 4 ODFs within a class is shown in Fig. 8(a). In contrast to the texture design problem, the property design problem clearly illustrates the presence of multiple solutions. A range of different processing sequences (indicated in Fig. 8(a)) yield similar distributions of the young's modulus. Thus, the methodology enables identification of new processes and selection of the economical process routes that leads to a desired property distribution based on available database of information.

To achieve a desired Young's modulus distribution as shown in Fig. 9(a), we resort to the gradient based optimization scheme with the processing sequences found using the classifier as the initial guess. As an example, a processing sequence of stage 1 of shear mode (mode 1) and stage 2 of tension mode is employed in the optimization procedure for achieving the desired property. A threshold of 0.05 is used for the selection of the adaptive basis. Using an initial guess -0.7 and 0.15 s^{-1} for the strain rates as found from classification, the final optimized process parameters were obtained as -0.03579 and 0.17339 s^{-1}, respectively. The elastic property distribution corresponding to the optimized process parameters identified is shown along with the desired distribution in Fig. 9(a). The variation in the objective during the iterations is shown in Fig. 9(b). The classification methodology is general and can be extended towards problems involving design of several other texture-dependent properties through appropriate

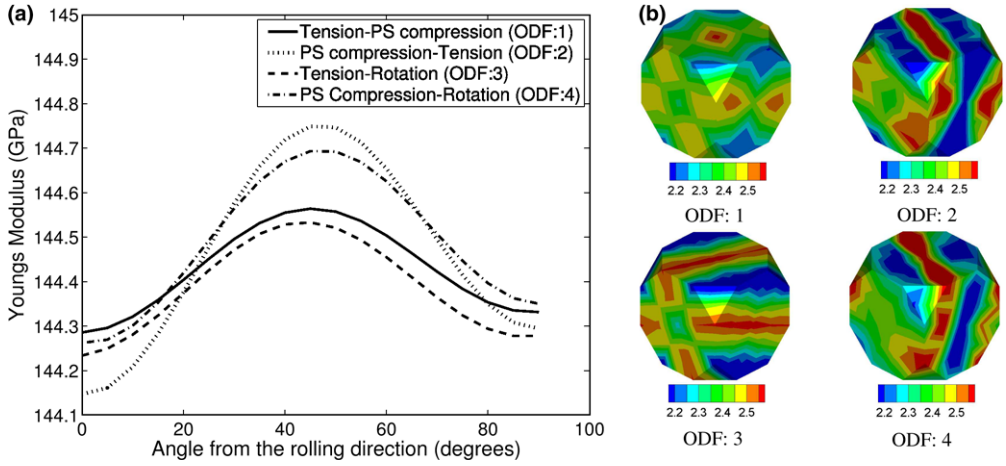

Fig. 8. (a) Classification based on property distribution: Young's modulus distribution of a class of ODFs (b) The corresponding ODFs.

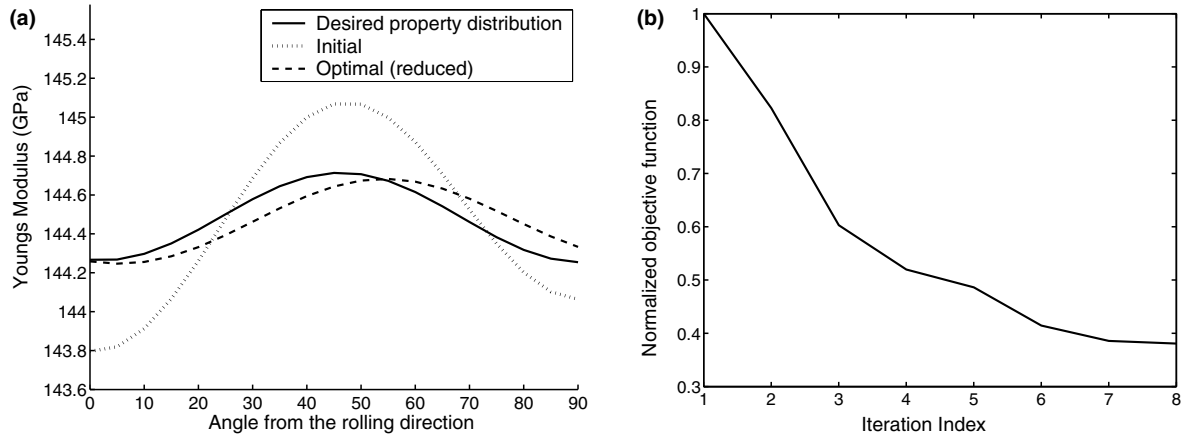

Fig. 9. Optimization of Youngs modulus distribution: (a) comparison of the desired distribution and optimized distribution and (b) variation of the objective function with iterations.

design of the processing sequence and process parameters.

7. Conclusions

The paper presents a data-driven reduced-order optimization procedure for the design of a process sequence to control the texture and texture-dependent properties. The inverse problem of identifying processes corresponding to desired texture is initially solved using the classifier algorithm over a database. The classifier matches the lower order features of the texture in the form of the pole density function over a class hierarchy to identify the sequence of processes that lead to the desired texture, hence, identifying multiple process paths that lead to the desired texture. These parameters are then fine tuned using gradient based optimization schemes. An unsupervised classifier based on the k-means algorithm is used for the identification of natural clusters within the database. The number of classes in the texture database is not known a-priori, hence, a Bayesian information criterion is used to identify the number of clusters.

Reduced-order control provides an efficient method for solving process design problems in reasonable time. The method of proper orthogonal decomposition provides a systematic way to obtain reduced-order models. A technique was presented in which modes corresponding to the intermediate stages of the design process are adaptively selected from a database. The database continuously improves during the optimization problems by adding new, unknown data sets, which would be useful during future optimization runs.

In the future, we visualize virtual databases containing material information at different length-scales built as a shared resource enabling interactive utilization as a knowledge base. Further, these databases can be integrated with computational process design algorithms to enable efficient design of materials at the micro-scale using deformation processes [16]. Such integration may allow the development of a computational design laboratory for process selection that leads to products of desired material properties in addition to accounting for other macro design constraints.

Acknowledgements

The work presented here was funded by the Mechanical Behavior of Materials program (Dr. D. Stepp, program manager) of the Army Research Office (grant W911NF-04-1-0283) and by the Computational Mathematics program (Dr. F. Fahroo, program manager) of the Air Force Office of Scientific Research (grant FA9550-04-1-0070). The work was conducted using the supercomputing facilities of the Cornell Theory Center.

References

[1] Adams BL, Henrie A, Henrie B, Lyon M, Kalidindi SR, Garmestani H. Microstructure sensitive design of a compliant beam. J Mech Phys Solids 2001;49:1639–63.
[2] Kalidindi SR, Houskamp JR, Lyons M, Adams BL. Microstructure sensitive design of an orthotropic plate subjected to tensile load. Int J Plasticity 2004;20(8–9):1561–75.
[3] Acharjee S, Zabaras N. A proper orthogonal decomposition approach to microstructure model reduction in Rodrigues space with applications to the control of material properties. Acta Mater 2003;51(18):5627–46.
[4] Ganapathysubramanian S, Zabaras N. Design across length scales: a reduced-order model of polycrystal plasticity for the control of microstructure-sensitive material properties. Comput Meth Appl Mech Engng 2004;193(45–47):5017–34.
[5] Kumar A, Dawson PR. Computational modeling of fcc deformation textures over Rodrigues' space. Acta Mater 2000;48:2719–36.

[6] Kumar A, Dawson PR. Modeling crystallographic texture evolution with finite elements over neo-Eulerian orientation spaces. Comput Meth Appl Mech Engng 1998;153(3–4):259–302.
[7] Barton NR, Boyce DE, Dawson PR. Pole figure inversion using finite elements over Rodrigues space. Textures Microstruct 2002;35(2):113–44.
[8] Ravindran SS. Adaptive reduced-order controllers for a thermal flow system using proper orthogonal decomposition. SIAM J Sci Comput 2002;23(6):1924–42.
[9] Kocks UF, Tóme CN, Wenk HR. Texture and anisotropy: preferred orientations in polycrystals and their effect on materials properties. Cambridge: Cambridge University Press; 1998.
[10] Bunge HJ. Texture analysis in materials science, mathematical methods. New York: Butterworth-Heinemann; 1983.
[11] Ganapathysubramanian S, Zabaras N. Modeling the thermoelastic-viscoplastic response of polycrystals using a continuum representation over the orientation space. Int J Plasticity 2005;21/1:119–44.
[12] Sundararaghavan V, Zabaras N. A dynamic material library for the representation of single phase polyhedral microstructures. Acta Mater 2004;52/14:4111–9.
[13] Sundararaghavan V, Zabaras N. Classification of three-dimensional microstructures using support vector machines. Comput Mater Sci 2005;32(2):223–39.
[14] Duda RO, Hart PE, Stork DG. Pattern classification. 2nd ed. New York: John Wiley and Sons; 2001.
[15] Pelleg D, Moore A. X-means: Extending *K*-means with efficient estimation of the number of clusters. In: Seventeenth International Conference on Machine Learning, Stanford University, 2000.
[16] Acharjee S, Zabaras N. The continuum sensitivity method for the computational design of three-dimensional deformation processes, Comput Meth Appl Mech Engng, in press.