Acta Materialia 111 (2016) 242–252

Contents lists available at ScienceDirect

Acta Materialia

journal homepage: www.elsevier.com/locate/actamat

Full length article

Towards a quantitative comparison between experimental and synthetic grain structures

P.G. Callahan [a], M. Groeber [b], M. De Graef [a, *]

[a] Department of Materials Science and Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, USA
[b] Materials and Manufacturing Directorate, AFRL/MLLMD, Wright-Patterson AFB, OH 45433, USA

ARTICLE INFO

Article history:
Received 4 January 2016
Received in revised form
29 March 2016
Accepted 30 March 2016
Available online 6 April 2016

Keywords:
Nickel alloys
3D characterization
Synthetic microstructure
Moment invariants

ABSTRACT

A method is proposed to quantitatively compare 3-D microstructural data using the Hellinger distance. Statistics from an experimentally observed 3-D IN100 microstructure are used to generate six synthetic microstructures. These are compared quantitatively to the experimental microstructure by computing the Hellinger distances between distributions of microstructural descriptors, such as the volume, the aspect ratio, and the affine moment invariant $\overline{\Omega}_3$. The synthetic microstructures use three starting shape classes, namely ellipsoids, superellipsoids, and shapes generated by truncating a cube with an octahedron; this latter shape class is proposed here for use in microstructure generation, and we derive relevant morphological descriptors. Synthetic microstructures are generated using either random grain placement or by constraining the grain placement to the experimental grain centroids. We present a method for determining microstructure similarity by random sampling from a reference microstructure as well as quantitative shape comparisons between synthetic and experimental microstructures.

1. Introduction

It has been well established that grain size and grain shape in a polycrystalline material strongly influence the mechanical properties of that material [1,2]. While grain size is relatively easy to measure and use in theoretical models, a proper incorporation of grain shape into a theoretical description is significantly more challenging because of the wide variety of possible shapes and the potential complexity of grain shapes (e.g., grains can exhibit a mixture of convex and concave facets). Progress can be made by application of grain shape averaging procedures to create $\mathcal{N}$-hedra [3,4], which are topological proxies for each corresponding class of irregular network polyhedra that contains the same number of faces. Such an averaging procedure discards details of the grain shapes and only employs topological characteristics that can be used subsequently to quantify volume and surface area evolution [4].

Recent experimental advances have facilitated the rapid collection of 3-D microstructural information, using destructive serial sectioning techniques in both optical [5] and electron microscopes [6], or using non-destructive techniques such as near-field high energy X-ray diffraction microscopy (nf-HEDM) [7] or x-ray tomography [8]. There has been a significant amount of work on characterizing this new 3-D data, particularly on quantification of the microstructures. Rowenhorst et al. studied microstructures using the integral of mean curvature of grain faces [9]; the integral of mean curvature is one of the 3-D Minkowski functionals [10] whose use in materials science has been proposed by Ohser and Mücklich [11]. Groeber studied the microstructure of an IN100 superalloy using normalized grain volume distributions, equivalent sphere diameters, and aspect ratios [12]. Moment invariants, which have traditionally been used for pattern recognition in other fields [13], and which will be described in more detail below, have recently been used alongside other shape descriptors, such as the shape quotient, for the characterization of microstructural features, such as precipitates and grains in both 2-D [14] and 3-D [15].

In addition to gathering and characterizing 3-D microstructural data, there has been significant recent development of software algorithms for the rapid generation of synthetic 3-D microstructures. Both *mbuilder* [16] and *DREAM.3D* [17] are freely available software tools that offer the capability to generate 3-D polycrystalline microstructures. *DREAM.3D* provides an open-source workbench for analysis of experimental data sets and generation

* Corresponding author. Department of Materials Science and Engineering, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213-3890, USA.
E-mail addresses: pcallahan@engineering.ucsb.edu (P.G. Callahan), Michael.Groeber.ctr@wpafb.af.mil (M. Groeber), degraef@cmu.edu (M. De Graef).

http://dx.doi.org/10.1016/j.actamat.2016.03.078
1359-6454/

of synthetic microstructures according to statistical parameters extracted from experimental data. Once a synthetic microstructure is generated, a comparison can be made between the constituent shapes, in this case grains, of the synthetic microstructures and a real microstructure. The ability to quantify grain shapes is a crucial prerequisite for the evaluation of the realism of synthetic microstructures; it is not necessarily sufficient to match only basic parameters, such as the average grain size, or the average number of grain neighbors.

In this paper, we describe the use of second order 3-D moment invariants to characterize the shape distributions of different microstructures. To compare different distributions, we use a similarity metric known as the Hellinger distance d_H. In our experience, the use of more familiar statistical methods, such as the Student's t-test, produce tenuous results when used to compare distributions describing microstructures. We begin this paper with a brief description of how 3-D grain-based microstructures are generated numerically (Section 2.1); then we describe the concept and applications of moment invariants in Section 2.2. In Section 3 we present a detailed comparison between an experimental IN100 alloy microstructure and several synthetic microstructures, generated using a range of starting shapes. We also introduce the Hellinger distance as a convenient comparison metric for distributions of shape descriptors.

2. Numerical algorithms

2.1. Synthetic microstructure generation

In this section, we present a brief overview of the numerical algorithms used in this paper. We begin with the generation of synthetic IN100 microstructures, which is described in more detail by Groeber et al. [18,19]. The synthetic microstructures are based on experimental data collected from an IN100 sample using focused ion beam serial sectioning [12]; statistical parameters were extracted from the experimental data, and samples from the resulting distributions are then used to generate synthetic microstructures using the *DREAM.3D* open source package. The microstructure generator employs several different classes of initial grain shapes (ellipsoids, superellipsoids, and truncated cuboctahedra); it is one of the goals of the present paper to evaluate the influence of this initial shape selection on the realism of the final microstructure.

The microstructure generation begins by creating a set of initial grains by sampling input distributions for size, aspect ratio and morphological orientation. The set of grains is then assigned to a particular shape class (ellipsoid, superellipsoid or cuboctahedra). Depending on the selected shape class, the grains must be assigned an additional attribute to completely define the shape. This final attribute is labeled and will be defined in the following section. Once the set of grains has been generated, they are placed at random positions inside a digital volume; the extent to which the arrangement is space-filling (amount of gaps and overlapping regions), as well as the number and size of neighboring grains in the synthetic data set are then determined. If the current arrangement does not adequately match the experimental distributions of number and size of neighboring grains or fill space to a high level, then grains are iteratively swapped and moved until a satisfactory arrangement is obtained or a set number of iterations is reached. Because no arrangement will completely fill space using the shape classes discussed here (barring extreme edge-cases), a final coarsening step is performed where placed grains expand heterogeneously, with speeds relative to their volume, aspect ratios and morphological orientation until all space is filled. Finally, crystallographic orientations are assigned to the grains to match an orientation distribution function, a misorientation distribution function, and a microtexture function; the precise choice of these orientation distribution and microtexture functions is not relevant to the shape characterization described in the present paper, so the accuracy with which these distributions are reproduced is not addressed here; the interested reader may consult [18,19] for additional information.

2.2. Moment invariants

The automated analysis of grain shapes can be performed by considering the shape indicator function, $D(\mathbf{r})$, of each individual grain; this 3-D step function equals unity inside the grain, and vanishes outside. The union of all disjoint grain indicator functions generates the complete bulk of the microstructure. One can consider the indicator function as an explicit descriptor of the spatial mass distribution of the grain, and hence it stands to reason that mass-derived quantities, such as the moment-of-inertia tensor, may be relevant to the description of the grain shape. The moment-of-inertia tensor components are derived from the object moments, μ_{pqr}, which are defined in Cartesian form as:

$$\mu_{pqr} = \iiint \mathrm{d}x\mathrm{d}y\mathrm{d}z x^p y^q z^r D(x,y,z), \tag{1}$$

where the integral extends over all of 3-D space. Special cases of this integral include the object volume $V = \mu_{000}$, and the center-of-mass coordinates $(x_c, y_c, z_c) = (\mu_{100}, \mu_{010}, \mu_{001})/V$. It is convenient to translate the object so that its center-of-mass coincides with the coordinate origin, which leads to the translation-invariant *central moments*, denoted by:

$$\overline{\mu}_{pqr} = \iiint \mathrm{d}x\mathrm{d}y\mathrm{d}z (x-x_c)^p (y-y_c)^q (z-z_c)^r D(x,y,z); \tag{2}$$

Numerically, moments of arbitrary order $n = p + q + r$ are easily computed for a shape defined on a discrete coordinate grid by means of the algorithm described by Novotni and Klein [20].

While the central moments $\overline{\mu}_{pqr}$ are translation-invariant, it is desirable to introduce additional invariance under rotations and scaling operations. This leads to the concept of *moment invariants* [21]. Moment invariants are combinations of central moments that are invariant with respect to a class of transformations; the moment invariants discussed here are invariant with respect to similarity (translation, rotation, isotropic scaling) or affine (similarity + anisotropic scaling and shearing) transformations. A similarity transformation S in 2-D is defined by the relation $\mathbf{x}_2 = S\mathbf{x}_1 = [sR\ \mathbf{t}]\mathbf{x}_1$, where s is a scalar representing the scaling between the original point given by an augmented vector $\mathbf{x}_1$ and the transformed point given by an augmented vector $\mathbf{x}_2$, R is the familiar rotation matrix:

$$R = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix},$$

and $\mathbf{t}$ is a column vector with the x and y translation components. The affine transformation is given by $\mathbf{x}_2 = A\mathbf{x}_1$ where [22]:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix}.$$

Note that for affine transformations, it is not required that $a_{11} = a_{22}$ and $a_{12} = -a_{21}$, so that we can have anisotropic scaling and shear. A brief description of the 3-D second order moment invariants follows; for a more complete derivation see Refs. [23,24].

In 3-D, the second order moment invariants are given by (from here on we omit the bars over the central moment symbols):