A GREEDY METHOD FOR RECONSTRUCTING POLYCRYSTALS FROM THREE-DIMENSIONAL X-RAY DIFFRACTION DATA

Arun K. Kulshreshth

Department of Computer Science
The Graduate Center, CUNY, NY 10016, USA
and
Center for Fundamental Research: 'Metal Structures in Four Dimensions'
Risø DTU, Technical University of Denmark, DK-4000 Roskilde, Denmark

Andreas Alpers

Center for Fundamental Research: 'Metal Structures in Four Dimensions'
Risø DTU, Technical University of Denmark, DK-4000 Roskilde, Denmark

Gabor T. Herman

Department of Computer Science
The Graduate Center, CUNY, NY 10016, USA

Erik Knudsen

Center for Fundamental Research: 'Metal Structures in Four Dimensions'
Risø DTU, Technical University of Denmark, DK-4000 Roskilde, Denmark

Lajos Rodek

Department of Computer Science
The Graduate Center, CUNY, NY 10016, USA
and
Center for Fundamental Research: 'Metal Structures in Four Dimensions'
Risø DTU, Technical University of Denmark, DK-4000 Roskilde, Denmark

Henning F. Poulsen

Center for Fundamental Research: 'Metal Structures in Four Dimensions'
Risø DTU, Technical University of Denmark, DK-4000 Roskilde, Denmark

(Communicated by John Schotland)

Abstract. An iterative search method is proposed for obtaining orientation maps inside polycrystals from three-dimensional X-ray diffraction (3DXRD) data. In each step, detector pixel intensities are calculated by a forward model based on the current estimate of the orientation map. The pixel at which the experimentally measured value most exceeds the simulated one is identified. This difference can only be reduced by changing the current estimate at a location from a relatively small subset of all possible locations in the estimate and, at each such location, an increase at the identified pixel can only be achieved by changing the orientation in only a few possible ways. The method selects the location/orientation pair indicated as best by a function that measures data consistency combined with prior information on orientation maps. The superiority of the method to a previously published forward projection Monte Carlo optimization is demonstrated on simulated data.

1. Introduction.

1.1. **Relevance of the application field.** The inverse problem of recovering poly-crystal orientation maps from X-ray diffraction data arises as an imaging problem in materials science. Many materials − such as metals, ceramics and alloys − are composed of crystalline elements. These elements, called *grains*, might all share the same crystal lattice structure, but they typically differ in size, shape and orientation of the lattice. For deformed materials even the orientation might differ slightly within the grain. An image of the material at the grain level should therefore provide for each location i two quantities $f(i)$ and $o(i)$: $f(i)$ is the label of the grain that occupies i, while $o(i)$ is its orientation at i. Thus recovering an image at the grain level is equivalent to recovering the *grain map f* and the *orientation map o*, which both act as functions on the image domain.

The physical, chemical and mechanical properties of the material are to a large extent governed by the geometrical features of the grains and their relative positions (such features include the correlation between the orientation of two neighboring grains and the morphology of the boundary separating them). Properties of grains are studied in a number of different applications in materials science. Such applications include the study of polycrystalline deformation, recrystallization, and phase transformation, [8, 19, 22].

Until recently, no nondestructive methods existed for imaging grains in 3D. The Three Dimensional X-Ray Diffraction (3DXRD) microscope, resulting from collaborative efforts between Risø DTU, Technical University of Denmark and the European Synchrotron Radiation Facility (ESRF), is a novel instrument that for the first time allows nondestructive structural characterization of polycrystals [27]. The 3DXRD microscope utilizes X-rays that generate diffraction patterns on a detector. From these diffraction patterns one tries to recover the grain/orientation maps. So far, 3D maps have been generated of the morphology of the individual grains within millimeter thick specimens [2, 3, 20, 23, 28, 29, 30, 34]. Furthermore, in favorable cases 3D animations can be produced of grain growth processes [32, 33].

1.2. **Methodological contributions.** Four methods for recovering maps from 3DXRD diffraction data can be found in the literature [3, 28, 30, 34]. Only the algorithm presented in [30], which we shall from now on call SARGM (Stochastic Algorithm for Reconstruction of Grain Maps), can recover orientation maps, the other algorithms implicitly assume that the orientations are constant within the grain (i.e., the material is undeformed). We present a new method that recovers maps of moderately deformed materials, which outperforms SARGM in terms of running time and quality of reconstruction. We follow the approach of all previous methods by focusing on layer-by-layer reconstructions, where 2D sections (layers) in the sample are illuminated consecutively and reconstructed independently. To generate 3D maps the results for the layers have to be stacked.

In [3, 30] tools from the relatively new mathematical discipline of discrete tomography [7, 14, 17, 18] were successfully included, to penalize solutions with unphysical variations in local orientations or erratic grain boundaries. We follow the same approach by trying to find a grain/orientation map pair that is indicated as best by a function on the maps; the function combines data consistency with prior map information. Previously, Monte Carlo methods have been used, because the function has numerous local optima. The sampled solution space, however, is extremely large since potentially every location in the maps can be processed. In

the present paper we demonstrate that it suffices to process (deterministically) the much fewer diffraction spot pixels on the detector, and backproject changes into the grain/orientation map space. This method is similar to the Algebraic Reconstruction Technique (ART) used in tomography where detector pixel intensities refer to line integrals. ART methodology has been used in photon migration imaging for cases where the integrals are not taken over lines [5, 11] − the case of hyperbolas in a discrete setting, as in our application, seems to be new.

We propose a scheme that determines the order in which detector pixels are processed. The scheme prioritizes detector pixels at which the difference of the actual measurement and the pseudo-measurement (i.e., the simulated diffraction intensity of the current solution estimate) are maximal. This is a so-called *greedy scheme*, which has already been successfully adopted in other applications of discrete tomography [14]. However, we apply this scheme to select pixels on the detector (and not in the image space), which − to our knowledge − is new.

Although the size of the search space is reduced considerably using the mentioned techniques, in practice it will remain rather large. Our method is turned into a practical method only by paying careful attention to details of implementation, which are therefore explained in the paper.

1.3. **Outline of the paper.** In what follows we summarize in Section 2 basic concepts of 3DXRD reconstruction, and we discuss the representation of orientations that we use for our algorithm. In Section 3 we give a description of the proposed algorithm. The implementational details are explained in Section 4. The simulations in Section 5 are used to test the quality of reconstructions in noiseless cases, and also as a function of noise and running time. We discuss the prospective uses and possible extensions of the algorithm in Section 6.

2. **Background.**

2.1. **3DXRD experimental setup.** Figure 1 shows the relevant 3DXRD setup. A sample fixed to a turntable is illuminated by a parallel and monochromatic incoming X-ray beam. Parts of the microstructure that happen to have orientations that fulfill *Bragg's law* (see next subsection) will give rise to *diffraction spots*, which are recorded by a high-spatial-resolution area detector placed in close proximity to the sample. The turntable (and thus the sample) is rotated by an angle ω around an axis perpendicular to the beam, and diffraction images are acquired for equally spaced values of $\omega \in [0, 90°]$.

2.2. **Diffraction.** Grains, as mentioned in the introduction, are smaller crystals (at different orientations) inside a polycrystal. The translational symmetry of a crystal can be described using the concept of the *Bravais lattice*. This lattice is the set

$$(1) \qquad \mathcal{L} = \{\mathbf{x} \in \mathbb{R}^3 \mid \mathbf{x} = \alpha_1 \mathbf{a} + \alpha_2 \mathbf{b} + \alpha_3 \mathbf{c}, \ \alpha_1, \alpha_2, \alpha_3 \in \mathbb{Z}\},$$

where $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^3$ are given vectors not all lying in the same plane. A crystal contains different families of parallel lattice planes, each of which is determined by an integer triple $\{hkl\}$ with no common factors, the so-called *Miller indices*. The triple $\{hkl\}$ denotes the infinite family of parallel equidistant planes, one of which passes through the origin, and the next nearest intersects the lattice axes in the points $\mathbf{a}/h$, $\mathbf{b}/k$, and $\mathbf{c}/l$ (if one of the indices is zero, it means that the plane does not intersect the corresponding axis). Figure 2 shows a Bravais lattice and the

family of lattice planes determined by the Miller indices {100}. For given lattice planes {hkl} we denote by **h** the uniquely determined vector that is perpendicular to these planes, pointing away from them, and whose magnitude is the reciprocal of the spacing.

An incoming X-ray beam with wavelength λ hitting the first lattice plane of a crystal at an angle θ will be reflected at the same angle. The reflection, however, interferes (constructively or destructively) with reflections that occur on parallel lattice planes that are deeper in the crystal. If constructive interference occurs, it produces a so-called *diffraction spot*. *Bragg's law*, giving a condition on θ, **h** and λ, characterizes exactly this situation. Let $\mathbf{s}_0$ and **s** denote unit vectors in the direction of the incoming and reflected beams. Constructive interference along **s** will occur if and only if the following condition (known as *Bragg's law in vector form* [35]) holds:

$$(2) \qquad \frac{\mathbf{s} - \mathbf{s}_0}{\lambda} = \mathbf{h}.$$

Figure 3 gives an illustration of Bragg's law in the 3DXRD setting where the laboratory coordinate system $(\widehat{x_l}, \widehat{y_l}, \widehat{z_l})$ is fixed as in Figure 1. In this system we have $\mathbf{s}_0 = (1, 0, 0)^T$ for the incoming beam, and **s** for the reflected beam, characterized in spherical coordinates by the two angles 2θ and η (Figure 1)

$$(3) \qquad \begin{pmatrix} \cos(2\theta) \\ -\sin(2\theta)\sin(\eta) \\ \sin(2\theta)\cos(\eta) \end{pmatrix}.$$

It should be noted that finite crystals, such as grains, contain a finite number of {hkl}-families. Above we have seen how **h**, in the coordinate system of the grain, can be computed from its Miller indices {hkl}. An *a priori* known orthogonal transformation, describing the rotation of the sample and the crystal orientation, transforms **h** into coordinates of the laboratory system. Alternative descriptions and additional details can be found in [2, 27].

2.3. **Forward and backward projections.** In the previous subsection we have seen that there are finitely many candidates **h** describing lattice planes for which diffraction can occur. We have discussed how **s** and $\mathbf{s}_0$ can be expressed in the laboratory system, and thus, by (2), we can calculate for each **h** the angles θ and η fulfilling Bragg's law (λ is given by the experimental setup). Suppose that the pair (θ, η) fulfills Bragg's law. Thus, for a given point (x_l, y_l, z_l) of the grain, we can easily compute, using (3), the associated diffraction point on the detector (L, y_{det}, z_{det}) as the intersection point of the line $(x_l, y_l, z_l)^T + \mathbb{R}\mathbf{s}$ with the detector plane. We thus obtain

$$(4) \qquad \begin{aligned} y_{det} &= -(L - x_l)\tan(2\theta)\sin(\eta) + y_l \\ z_{det} &= (L - x_l)\tan(2\theta)\cos(\eta) + z_l. \end{aligned}$$

In the present case of a line beam configuration (where the X-ray beam is used for illuminating one layer, say $z_l = 0$, in the specimen) we can find the set of points in the laboratory system that potentially can contribute diffracted intensity to a given point $\mathbf{d} = (y_{det}, z_{det})$ on the detector. For a given Bragg angle 2θ we obtain a one-dimensional curve, called the *projection curve*, that consists of all points (x_l, y_l) in the sample plane that satisfy

$$(5) \qquad (y_{det} - y_l)^2 - (L - x_l)^2\tan^2(2\theta) = -z_{det}^2, \quad x_l < L,$$

see Figure 4. The formula follows immediately from (4), and it is easily seen that the projection curve is a hyperbola. The diffraction spots on the detector result from a finite number of $\{hkl\}$-families (Section 5 describes the five $\{hkl\}$-families that are used in our simulations), which correspond to a finite number of θ values. Considering all those finite possibilities of θ, we form the set $\sigma(\mathbf{d})$ of all projection curves that potentially can contribute diffracted intensity at $\mathbf{d}$. We call $\sigma(\mathbf{d})$ the *set of projection curves* of $\mathbf{d}$.

2.4. **Representation of grains and orientations.** Our goal, as mentioned in the introduction, is to recover grain and orientation maps. Let the total number of locations in the $2D$ area of interest (sample layer) be I. We assign to each location i ($1 \leq i \leq I$) both a grain label, $f(i)$, and an orientation, $o(i)$. Our approach simultaneously generates the grain map, f, and the orientation map, o. Grains are labeled by $\ell \in \{1, 2, \ldots, n\}$, where n, the total number of grains, is assumed to be known *a priori*. For each grain ℓ an average orientation and the maximum orientation spread Θ_ℓ must be known. Orientations can be represented in various ways, most common are representations in *Euler angles*, *Rodrigues vectors* and *unit quaternions* [4, 26]. These representations describe rotations in $3D$-space. In this article (as in [30, 31]), we choose representations by unit quaternions, mainly for two reasons: quaternions form a division ring that, in our case, leads to simpler equations (which in turn leads to faster algorithms), and the orientation space is bounded and has no singularities (which is favorable for discretizations) [26].

We write unit quaternions in the form

$$(6) \qquad \mathbf{q} = (a, b, c, d) = (cos(\varphi/2), \mathbf{n} \sin(\varphi/2)),$$

where the 3D unit vector $\mathbf{n}$ and the real scalar φ are the axis and angle of rotation, respectively. Due to crystal symmetries, multiple unit quaternions may represent the same orientation. The distance between orientations represented by unit quaternions $\mathbf{q_1}$ and $\mathbf{q_2}$ is defined as

$$(7) \qquad d(\mathbf{q_1}, \mathbf{q_2}) = \min_{\mathbf{s_1}, \mathbf{s_2} \in S} r(\mathbf{q_1} \mathbf{s_1}, \mathbf{q_2} \mathbf{s_2}),$$

where S is the set of symmetry rotations as given by crystallography and r is

$$(8) \qquad r(\mathbf{q_1}, \mathbf{q_2}) = 1 - |a_3| = 1 - |\cos(\varphi_{12}/2)|.$$

Here a_3 is the first component of the quaternion $\mathbf{q_3} = \mathbf{q_2} \overline{\mathbf{q_1}}$ ($\mathbf{q_2}$ multiplied by the conjugate of $\mathbf{q_1}$) and φ_{12} is the rotation angle associated with the rotation from $\mathbf{q_1}$ to $\mathbf{q_2}$. Usually we specify a distance $d(\mathbf{q_1}, \mathbf{q_2})$ by the associated angle of disorientation, which is defined as that $\psi \in [0, 180°]$ for which $\cos(\psi/2) = 1 - d$.

For unit quaternions, each component has a value with magnitude less than or equal to 1. We can choose a to be non-negative (because the unit quaternions $\mathbf{q}$ and $-\mathbf{q}$ define the same rotation) and then it is uniquely determined by the other three components using $a^2 + b^2 + c^2 + d^2 = 1$. Hence we can discretize the whole unit quaternion space by sampling the values of components b, c, and d in the interval $[-1, 1]$ and allowing only values that satisfy $b^2 + c^2 + d^2 \leq 1$.

With respect to this discretization we define the discretized space $Q(\ell)$ of possible orientations for each location in the grain ℓ with average orientation $\mathbf{q}_\ell = (a_\ell, b_\ell, c_\ell, d_\ell)$ and maximum orientation spread Θ_ℓ as follows

$$(9) \qquad Q(\ell) = \{\mathbf{q} = (a, b, c, d) : a \geq 0, d(\mathbf{q}_\ell, \mathbf{q}) \leq \Theta_\ell\},$$

$$b = b_\ell + i_1 u, \quad c = c_\ell + i_2 u, \quad d = d_\ell + i_3 u,$$
$$a^2 + b^2 + c^2 + d^2 = 1,$$

where i_1, i_2, i_3 are integers and u is the sampling unit of each component.

3. **Algorithm.** Our iterative algorithm is deterministic, but has many similarities with the stochastic *Metropolis algorithm* [24, 36]. It maximizes an objective function $\gamma(f, o)$ corresponding to a grain-map f and an orientation map o for the layer of interest.

3.1. **Objective function.** The objective function, from [30], has the form

$$\text{(10)} \qquad \gamma(f, o) = \exp\left(-\beta\left(\alpha \|\mathbf{P}_o - \mathbf{P}_{input}\|_1 + H(f, o)\right)\right).$$

where $\alpha, \beta > 0$ are predefined system parameters, which we specify in Subsection 5.2, and the two terms $\|\mathbf{P}_o - \mathbf{P}_{input}\|_1$ and $H(f, o)$ are computed from f and o as follows. The first term is a measure of the correspondence of the experimental pixel intensities on the detector − arranged in a vector $\mathbf{P}_{input}$ − and of the simulated (projected) pixel intensities − arranged in a vector $\mathbf{P}_o$ − given the orientation map o. The second term comprises prior information about grain orientations and morphology. As such it splits into two terms

$$\text{(11)} \qquad H(f, o) = H_1(f, o) + H_2(f).$$

The H_1 term, measuring the degree of homogeneity of the orientation map, is defined as follows. Let $\mathcal{C}_{NN}^{f,\ell}$ denote the set of all pairs (i, j) of locations that are 4-connected nearest neighbors (adjacent either vertically or horizontally) within a grain ℓ (i.e., the set of all locations i such that $f(i) = \ell$). Similarly, let $\mathcal{C}_{NNN}^{f,\ell}$ denote the set of all location pairs (i, j) that are nearest neighbors (adjacent diagonally) within grain ℓ. Then

$$\text{(12)} \qquad H_1(f, o) = -\sum_{\ell=1}^{n}\left(\sum_{C \in \mathcal{C}_{NN}^{f,\ell}} \lambda_1 \Phi_C(o) + \sum_{C \in \mathcal{C}_{NNN}^{f,\ell}} \lambda_2 \Phi_C(o)\right),$$

with

$$\text{(13)} \qquad \Phi_C(o) = e^{-\frac{(d(o(i), o(j)))^2}{2\delta^2}}.$$

The parameters λ_1, λ_2, and δ are system parameters specified in Subsection 5.2 below. The H_2 term, measuring the likelihood of the grain morphologies in the grain map f, is based on a weighted counting of seven predefined local 3×3 features $G_0, \cdots, G_6$ in the grain map. Let $N(G_c, f)$ count the number of times a feature G_c occurs in f, and let U_c denote a predefined weight (preference) for that class. Then,

$$\text{(14)} \qquad H_2(f) = -\sum_{c=0}^{6} N(G_c, f) U_c.$$

The weights U_c and features G_c for our simulations, including "grain interior," "corner" and "random structure," shall be defined in Subsection 5.2. We remark that (10) is inspired by statistical mechanics: $\gamma(f, o)/Z$ with a normalization factor Z is a well-known probability measure - the so-called *Gibbs measure* [36]. The application of Gibbs priors for 3DXRD data from undeformed specimens was derived and motivated in [3].

3.2. **General outline of our approach.** The input to our algorithm is a grain map f and an orientation map o. How these maps are derived from physical data is discussed in Subsection 5.3.

Our algorithm is a deterministic iterative method, which locally changes grain labels and orientations. We aim for changes that have maximal effect (in terms of improving the objective function) by selecting detector pixels using a *greedy approach*. The changes, however, are made in the location space − and it should be noted that such a change might in fact decrease the objective function value. Our method is therefore not a greedy approach in location space. Furthermore, we combine this method with a technique similar to *tabu search* [12] that forces the algorithm to avoid reprocessing recently processed detector pixels. For this purpose, we keep a *tabu list* $\mathcal{T}$ of a fixed length that contains such pixels. Both greedy and tabu search are local search techniques used in a broad range of applications [1, 12]. Tabu search, in particular, tries to avoid getting trapped in local optima by prohibiting the option of visiting a recently left local optimum.

The general outline of our algorithm can be seen in Algorithm 1. We iteratively select a pixel $\mathbf{d}$ on the detector and select a local change in grain label and orientation that best affects $\mathbf{d}$. We update to new maps, and terminate the iterative procedure if a termination criterion is met. At termination we return the current pair (f, o) since we found that this yields already a high quality reconstruction. While it might be better to return the pair (f, o) that is the optimal one observed during the whole iterative process, but in the experiments reported below this was found to be unnecessary. We now explain the steps of the algorithm in more detail.

Input: Initial estimates (f, o) of a grain and an orientation map.
Output: Maps (f, o) for which the value of γ is nearly maximal.

Set $\mathcal{T} := \emptyset$;
Repeat
    Let $\mathcal{D}$ be the set of all detector pixels $\mathbf{d}$ for which the difference $\Delta_{\mathbf{d}}$
      of the actual measurement and the pseudo-measurement exceeds
      the noise level;
    If $\mathcal{D} \setminus \mathcal{T} = \emptyset$ **then** set $\mathcal{T} := \emptyset$;
    If $\mathcal{D} \neq \emptyset$ **then**
    Begin
        Choose $\mathbf{d} \in \mathcal{D} \setminus \mathcal{T}$ with $\Delta_{\mathbf{d}} = \max_{\mathbf{d}' \in \mathcal{D} \setminus \mathcal{T}} \Delta_{\mathbf{d}'}$;
        Insert $\mathbf{d}$ into $\mathcal{T}$;
        Select from all triples (i, f_i, o_i), such that grain f_i at location i with
          orientation o_i leads to the highest $\gamma(f', o')$ if f' is obtained from f
          by changing the label at i to f_i, and o' is obtained from o by chang-
          ing the orientation at i to o_i;
        Set $f := f'$ and $o := o'$;
        Update simulated diffraction intensities;
    End;
Until termination criterion is met;
Return (f, o).

Algorithm 1: General description of our greedy method.

3.2.1. *Detector Pixel Selection.* We choose the detector position $\mathbf{d}$, among those which are not in $\mathcal{T}$, for which the difference $\Delta_{\mathbf{d}}$ of the actual measurement and the pseudo-measurement (simulated diffraction intensity for the current estimate) has the highest value. The idea is that we want to increase at this detector pixel the simulated diffraction intensity − thus decreasing $\Delta_{\mathbf{d}}$. For the selection of $\mathbf{d}$ we only take pixels into account for which $\Delta_{\mathbf{d}} > \varepsilon$, where $\varepsilon \geq 0$ is chosen to reflect the level of noise. If no pixel with $\Delta_{\mathbf{d}} > \varepsilon$ exist then the algorithm terminates (Subsection 3.2.3). If all $\mathbf{d}$ with $\Delta_{\mathbf{d}} > \varepsilon$ are elements of $\mathcal{T}$ then we empty the tabu list, which enables us then to select an appropriate $\mathbf{d}$. Pixel $\mathbf{d}$ is inserted into $\mathcal{T}$ in the following way: If the length of $\mathcal{T}$ does not exceed a predefined parameter T_{max} then $\mathbf{d}$ is appended to the list, otherwise it replaces the element that has been a member of $\mathcal{T}$ for the last T_{max} iterations.

3.2.2. *Update.* For the selected detector pixels $\mathbf{d}$, we consider all triples (i, f_i, o_i) with

$$(15) \qquad i \in \sigma(\mathbf{d}) \text{ (Subsection 2.3 describes the set of projection curves } \sigma(\mathbf{d})),$$

$$(16) \qquad f_i \in \left\{ f(i) \cup \bigcup_{j \in N_4(i)} f(j) \right\},$$

$$(17) \qquad o_i \in Q(f_i),$$

where $N_4(i)$ denotes the set of all locations that are 4-connected neighbors to i. Among those triples (i, f_i, o_i) we keep only those in a list $\mathcal{S}(\mathbf{d})$ that potentially contribute to the intensity in $\mathbf{d}$. Thus, every triple can only have the grain label from one of its 4-neighbors, or it might remain unchanged. For each element (i, f_i, o_i) in $\mathcal{S}(\mathbf{d})$ we define the grain map f' to be the same as f except that $f'(i) = f_i$ and the orientation map o' to be the same as o except that $o'(i) = o_i$. We select an (f', o') to replace (f, o) for which $\gamma(f', o')$ is maximal.

3.2.3. *Termination.* Our algorithm terminates if one of the following termination criteria is met: (1) There are no detector pixels $\mathbf{d}$ with $\Delta_{\mathbf{d}} > \varepsilon$ (i.e., with this noise level no further detector image improvements are possible), (2) a maximum number of iterations is exceeded, (3) a maximum time for iterations is reached, or (4) the command $\mathcal{T} := \emptyset$ has been executed a prescribed number of times (in our implementation this number is set to 5).

4. **Implementational details.** The algorithm has been implemented as a C program running under Linux. The simulations were performed on a PC with a 1.86 GHz Intel pentium M processor and 512 MB RAM. Since the calculation of possible assignments during each optimization step is potentially expensive (directly proportional to the orientation spread of the grain), considerable effort has been invested in optimizing the code. For reference purposes we summarize the relevant details of implementation.

4.1. **Look-up tables.** To save time during optimization we use look-up tables for calculations that can be pre-calculated.

- We pre-calculate $Q(\ell)$ since for every grain the average orientation and orientation spread are known *a priori*.

- During optimization we need to find a possible assignment (f, o) at some sample location $i \in \sigma(\mathbf{d})$ (Subsection 2.3) that will definitely give rise to a spot at the currently chosen detector location $\mathbf{d}$. Hence the list of forward projections that need to be calculated at location i will not change during the optimization process and it can be pre-calculated for all the sample locations (irrespective of $\sigma(\mathbf{d})$). These forward calculations may not be able to fit into the memory of the computer, so currently they are being swapped to the hard drive and are loaded into the memory as required.
- The computation of the $H_2(f)$ term in the objective function (Subsection 3.1) can be reasonably speeded up by employing a look-up table for the potential values of specific 3×3 local configurations, as discussed by [3].
- We also keep a look-up table for calculating the distance between any two quaternions $\mathbf{q}_1$ and $\mathbf{q}_2$ of the discretized unit quaternion space [30].

4.2. **Additional speedups.** The relevant values of the objective function are updated by taking into account only those terms that have changed since the preceding iteration. Further speedups are obtained by using efficient data structures.

- Instead of repeatedly finding the $\mathbf{d}$ in $\mathcal{D}$ that maximizes $\Delta_{\mathbf{d}}$, we order the elements $\mathbf{d}$ of $\mathcal{D}$ according to the size of $\Delta_{\mathbf{d}}$ once at the beginning, and then we update this ordered list after each change. The ordered list, called *priority queue*, is implemented as *binary heap* [9], which is a data structure that enables fast updates.
- Building the complete list $\mathcal{S}(\mathbf{d})$ for the proposed triples (i, f_i, o_i) in the algorithm is expensive, but it will have to be done at most once (then it is stored and so no recalculation is needed) for any $\mathbf{d}$ (and not at all for most $\mathbf{d}$).

5. **Simulations.** We performed simulations under idealized conditions as in [30] (details see below) to demonstrate the performance of our new approach − as well as to investigate the dependence of the current algorithm on experimental noise.

5.1. **Test data.** Three 64×64 location test maps of aluminum microstructures of varying complexity were defined in [30]. In each case the orientation map was generated by electron microscopy (EBSD), which was then discretized such that all resulting orientations were representable as quaternions in the discretized unit quaternion space. Furthermore, by applying crystal symmetry operations we ensured that every $\mathbf{q}$ belonging to grain ℓ is an element of $Q(\ell)$. Based on these discretized test maps, 3DXRD data sets were simulated as detailed in [30]. As we shall see, the fact that the simulations are based on the special discretizations used in our algorithm enables us to obtain perfect reconstructions (This idealization is further discussed in Subsection 5.4.) In passing we mention that 91 projections were generated for each case, corresponding to equally spaced rotation angles over a rotation range of $90°$. For simulating the diffraction spots, we particularly included the $\{111\}$, $\{200\}$, $\{220\}$, $\{222\}$, and $\{311\}$ *hkl*-planes since they give rise to the brightest spots on the detector.

The three orientation maps represent different complexities in terms of the number of grains and the orientation spread within grains (defined as the maximum distance between the average orientation of the grain and the orientation of any location in the grain). Visually, the test orientation maps are indistinguishable from their reconstructions, which are shown in the top two rows of Figures 5 to 7. However, the numerical values in the reconstructions produced by the two methods

that are being compared are not identical, as can be seen in the bottom two rows of Figures 5 that depict the differences between the test maps and their reconstructions.

- **Case I:** This map comprises 11 grains; on average the orientation spread within grains is $3°$, but one grain has a spread of $7°$.
- **Case II:** This map comprises 26 grains; the average orientation spread is about $7°$, but the spreads within grains vary from $3°$ up to $22°$.
- **Case III:** This map comprises 3 grains; the average orientation spread is $14°$.

The simulated detector images were subjected to varying degrees of noise. As an approximation of Poisson noise, the noise was implemented by increasing/decreasing every pixel's intensity (photon count) I_0 by a uniformly distributed random number in the range $[-I_0\mu, I_0\mu]$, subject to the constraint of non-negativity. In the following, noise levels μ will be indicated in percents.

5.2. **System parameters.** The system parameters in the optimization function, see (10) and (12), are $\alpha = \beta = \lambda_1 = 1$ and $\lambda_2 = 1/\sqrt{2}$ as described in [30]. The Gibbs potentials U_c in (14) were selected as for the set U^1 in [3]. Furthermore, we assume that the material is single phase and that the crystallographic space group and the lattice parameters are also known in advance. The value of δ in (13) for the three test cases was set to $11.5°$, $16.2°$, and $14.5°$, respectively. The value of the sampling unit u in (9) was set to 0.02. The default length T_{max} of the tabu list was set to 15% of the number of elements in $\mathcal{D}$ initially.

5.3. **Input maps for the initialization step.** As input we need maps f and o. We obtain these maps by using SARGM, which only needs the center-of-mass information for each grain. SARGM yields reasonably good reconstructions in a very short time. However, frequently it stops increasing the figure-of-merit (specified below in Subsection 5.4) after some time (Figure 8). Our goal is to demonstrate that further improvements can be achieved deterministically, with our algorithm using the maps produced by SARGM as input. Thus, for the simulations we take a two-phase approach: We obtain grain and orientation maps from center-of-mass "seeds" by running SARGM for a short while, then these maps are used as input (f, o) for our algorithm. For comparison reasons, we monitor simultaneously the progress of SARGM at these later stages.

5.4. **Quality of reconstruction.** The quality of reconstruction is determined by a figure-of-merit (FOM) that measures the distance between the original orientation $o^{orig}(i)$ and the resulting orientation $o^{res}(i)$ for each location i:

$$(18) \qquad FOM = 1 - \frac{1}{I \cdot d_{max}} \sum_{i=1}^{I} d\left(o^{orig}(i), o^{res}(i)\right).$$

Here d_{max} is the maximal possible value of d, which for cubic lattice systems corresponds to the angle $\varphi_{12} \approx 62.8°$, and I is the number of locations. Evidently FOM = 1 for a perfect reconstruction.

In Figures 5 to 7 the results of the new algorithm are compared to SARGM for the three test cases in the idealized situation of no noise. In all cases, the new algorithm recovers the test phantom exactly, but SARGM does not.

Figure 8 compares the rate of convergence of the two algorithms for Test Case I. As initial maps for our algorithm we used the maps produced by SARGM after 1 second. SARGM initially is faster, but eventually slows down. Even after 8 hours

it is hovering around the FOM value of 0.99966. In contrast, the new algorithm converges nearly linearly with time and finds an exact reconstruction (FOM = 1) in 54 seconds. Note that exact reconstructions are possible due to the fact that the simulated projection data were generated from discretized test phantoms; one cannot expect perfect reconstructions from data physically collected from real (undiscretized) objects.

Figure 9 summarizes our results for 50% and 100% of noise. The noise in experimental data is estimated to be of the order of 10%, so the effect is clearly exaggerated in these simulations. In all cases the FOM of the orientation maps produced with the new algorithm was found to be better than the FOM for the corresponding orientation maps produced by SARGM.

6. **Discussion.** Grain maps are inherently discrete objects in the sense that the reconstructed grain labels are drawn from a discrete set. In fact, the subproblem of reconstructing a single grain (with known orientation) amounts to solving a system $Ax = b$ of linear equation, where the vector of unknowns x is a 0-1 vector [3], [28]. This is a classical problem in *combinatorial optimization*, and it is well-known that it is $\mathbb{NP}$-hard in general (and therefore believed to be computationally intractable [21]). In our case, A has a special structure that might be exploited, because the equations correspond to intensity summations along hyperbolas, all in the same plane. However, the slightly different case in which the intensities are summed up along straight lines is known to lead to $\mathbb{NP}$-hard problems [10]. A practically useful convergence proof would have to be for an algorithm that converges in polynomial time, and for the reasons just expressed we do not expect that such an algorithm exists for our problem.

Heuristics seeking for approximate rather than exact solutions, however, can be quite effective. The theoretical foundation of tabu search (that we use in our approach) is not far developed, but the method has been successfully applied to many other $\mathbb{NP}$-hard problems, such as *maximum satisfiability* [6], *max k-cut* [16], and 0-1 *knapsack* [15]. In common with these earlier papers that represent in-practice successful algorithms for problems for which there is no algorithm that is guaranteed to converge in polynomial time in all cases, our paper also lacks theoretical convergence results and error analysis, but demonstrates instead the actual performance of the proposed algorithm on examples from the intended application area.

In the following we present some possible generalizations of the algorithm:

- It is known that − given a particular sample location − the orientations that can contribute to the intensity at a certain detector pixel at a certain rotation angle ω are positioned on a circle in quaternion space [25]. It seems useful to restrict the search in $Q(\ell)$ to this circle. However, for the test cases used in this paper, such a procedure was found to be computationally slower than an unconstrained search. Nevertheless, use of this additional information may be relevant, e.g., in the case of highly deformed samples where the size of $Q(\ell)$ will be large.
- 3DXRD experiments often involve several settings of the detector at different sample-to-detector distances (Figure 1). Including such additional data in the reconstruction is straightforward. We speculate that such an extension would enhance accuracy substantially in case there are large overlaps of diffraction spots, i.e., for large maps with grains exhibiting a large mosaic spread. Other relevant extensions involve the use of more general sample geometries and of

tilted detectors (in practice it can be difficult to ensure that the detector plane is truly perpendicular to the beam).

- In the experiments reported above, the discretization of the test phantoms exactly matches the discretization of the orientation space used in the algorithm. This is clearly an idealized situation, which makes it possible to have an exact reconstruction. Evaluation of reconstruction quality when physically collected data from real (undiscretized) objects are used is a nontrivial task and is left for future investigations.

7. **Conclusions.** We have introduced a *greedy* deterministic algorithm for the reconstruction of orientation maps from 3DXRD data of moderately deformed specimens. We have evaluated its performance on simulated data (in the idealized case of objects with discretized orientations) and have shown that (at least for such data) the new algorithm outperforms a previously published one that makes use of *random sampling* [30], both in terms of the quality of reconstruction and of running time.
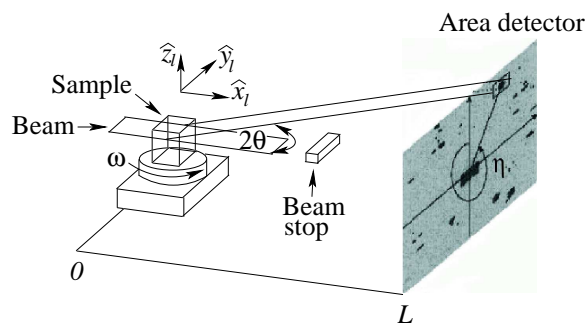
FIGURE 1. Sketch of the 3DXRD experimental setup. The Bragg angle 2θ, the rotation angle ω and the azimuthal angle η are indicated for a part of the grain that gives rise to diffraction. The laboratory coordinate system is given as $(\widehat{x_l}, \widehat{y_l}, \widehat{z_l})$. Adapted from [27].

REFERENCES

[1] E. Aarts and J. K. Lenstra (eds.), "Local Search in Combinatorial Optimization," Princeton University Press, Princeton, 2003.

[2] A. Alpers, E. Knudsen, H. F. Poulsen and G. T. Herman, *Resolving ambiguities in reconstructed grain maps using discrete tomography*, Electron. Notes Discrete Math., **20** (2005), 419–437.
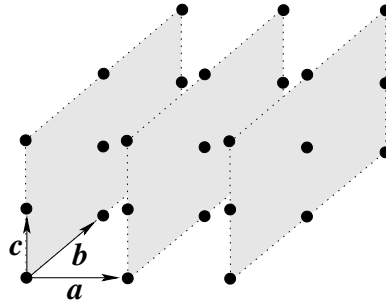
FIGURE 2. A Bravais lattice and {100} lattice planes.
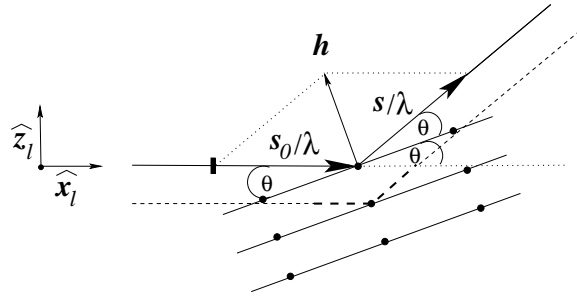
FIGURE 3. Illustration showing the relevant relations, in the laboratory system $(\widehat{x}_l, \widehat{y}_l, \widehat{z}_l)$, for Bragg's law. The incoming beam (in direction $\mathbf{s}_0/\lambda$) is diffracted at the angle 2θ along $\mathbf{s}/\lambda$. Since the $\widehat{y}_l$-axis points away from the reader, it just appears as a point.
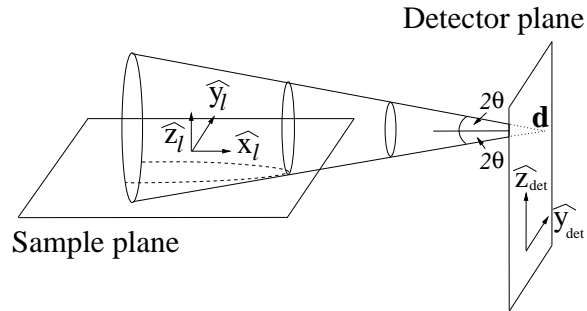
FIGURE 4. Geometry of the set of projection curves in direct space. In our case $z_l = 0$. The dotted curve represents a curve of $\sigma(\mathbf{d})$ for a particular θ value (see (5)).

[3] A. Alpers, H. F. Poulsen, E. Knudsen and G. T. Herman, *A discrete tomography algorithm for improving the quality of three-dimensional X-ray diffraction grain maps*, J. Appl. Cryst., **39** (2006), 582–588.

[4] S. L. Altmann, "Rotations, Quaternions, and Double Groups," Clarendon Press, Oxford, 1986.

[5] S. R. Arridge, *Optical tomography in medical imaging*, Inverse Probl., **15** (1999), R41–R93.
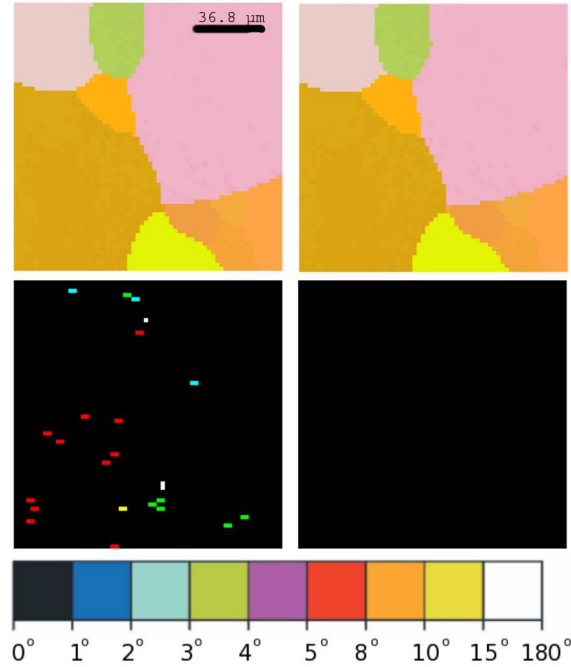
Figure 5. The reconstruction of Test Case I using noiseless projections. Upper-left: Orientation map reconstructed by SARGM. Upper-right: Orientation map reconstructed by our new algorithm. The color code for both of these maps is based on the mapping of components (b, c, d) of unit quaternions to the color (red, green, blue) $= I_{max}(1 - |b|, 1 - |c|, 1 - |d|)$, where I_{max} designates the maximal intensity of any color primary. Lower-left: Difference of the reference and the orientation maps reconstructed by SARGM. Lower-right: Difference of the reference and the reconstructed orientation maps using our new algorithm. The color of the pixels is determined by the distance (rotation angle) of corresponding orientation pairs, as shown in the color key at the bottom of the figure.

[6] R. Battiti and M. Protasi, *Reactive search, a history-based heuristic for MAX-SAT*, ACM J. Exper. Algor., **2** (1997).

[7] B. M. Carvalho, G. T. Herman, S. Matej, C. Salzberg and E. Vardi, *Binary tomography for triplane cardiography*, in "Information Processing in Medical Imaging" (eds. A. Kuba, M. Samal and A. Todd-Pokropek), Springer-Verlag, Berlin, (1999), 29–41.

[8] J. W. Christian, "The Theory of Transformations in Metals and Alloys," Pergamon, Oxford, 1981.

[9] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms," 2nd edition, MIT Press, Cambridge, 2001.

[10] R. J. Gardner, P. Gritzmann and D. Prangenberg, *On the computational complexity of reconstructing lattice sets from their X-rays*, Discrete Math., **202** (1999), 45–71.

[11] R. J. Gaudette, D. H. Brooks, C. A. DiMarzio, M. E. Kilmer, E. L. Miller, T. Gaudette and D. A. Boas, *A comparison study of linear reconstruction techniques for diffuse optical tomographic imaging of absorption coefficient*, Phys. Med. Biol., **45** (2000), 1051–1070.

[12] F. W. Glover and M. Laguna, "Tabu Search," Springer-Verlag, Berlin, 1997.
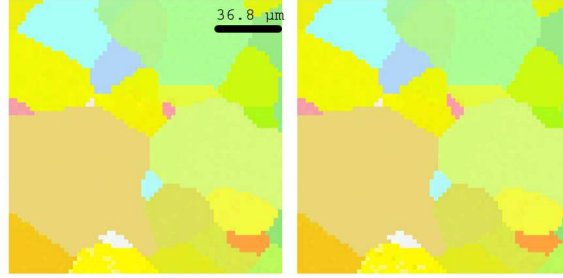
FIGURE 6. The reconstruction of Test Case II using noiseless projections. Left: Orientation map reconstructed by SARGM. Right: Orientation map reconstructed by our new algorithm.
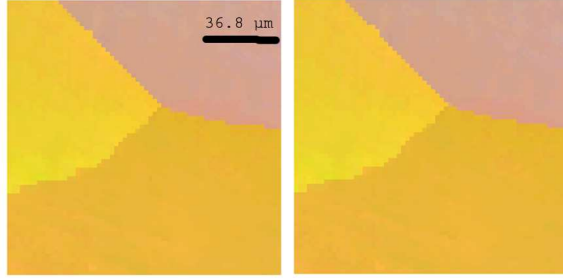

FIGURE 7. The reconstruction of Test Case III using noiseless projections. Left: Orientation map reconstructed by SARGM. Right: Orientation map reconstructed by our new algorithm.
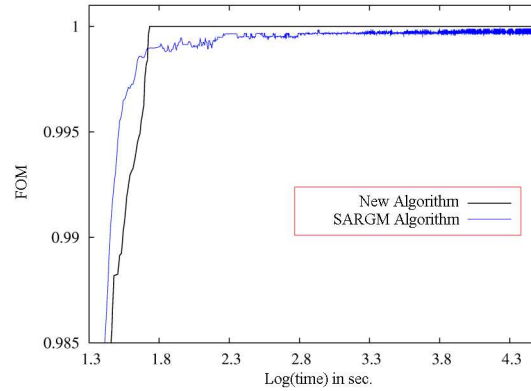

FIGURE 8. Plot of the FOM (18) as a function of time for both algorithms for Case I.

[13] H. Grimmer, *Disorientations and coincidence rotations for cubic lattices*, Acta Cryst., **A30** (1974), 685–688.

[14] P. Gritzmann, S. de Vries and M. Wiegelmann, *Approximating binary images from discrete X-rays*, SIAM J. Optim., **11** (2000), 522–546.
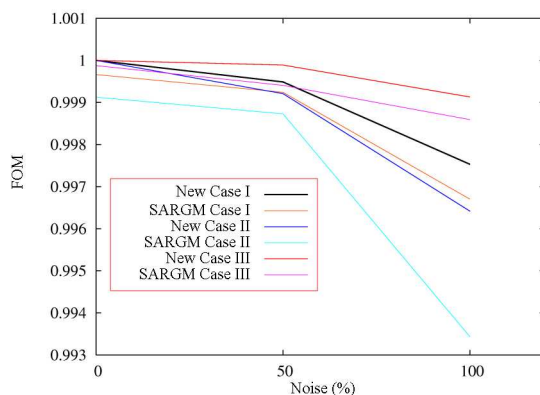
Figure 9. Comparing the quality of reconstruction by the new algorithm and by SARGM as a function of noise for the three test cases.

[15] S. Hanafi and A. Freville, *An efficient tabu search approach for the* 0-1 *multidimensional knapsack problem*, European J. Oper. Res., **106** (1998), 659–675.

[16] G. T. Herman and M. Kalinowski, *Classification of heterogeneous electron microscopic projections into homogeneous subsets*, Ultramicroscopy, **108** (2008), 327–338.

[17] G. T. Herman and A. Kuba (eds.), "Advances in Discrete Tomography and its Applications," Birkhäuser, Boston, 2007.

[18] G. T. Herman and A. Kuba (eds.), "Discrete Tomography: Foundations, Algorithms and Applications," Birkhäuser, Boston, 1999.

[19] F. J. Humphreys and M. Hatherly, "Recrystallization and Related Annealing Phenomena," Pergamon, Oxford, 1995.

[20] G. Johnson, A. King, M. G. Honnicke, J. Marrowa and W. Ludwig, *X-ray diffraction contrast tomography: A novel technique for three-dimensional grain mapping of polycrystals. II. The combined case*, J. Appl. Cryst., **41** (2008), 310–318.

[21] R. Karp, *Reducibility among combinatorial problems*, in "Proceedings of a Symposium on the Complexity of Computer Computations," Plenum Press, New York, (1972), 85–103.

[22] U. F. Kocks, C. N. Tome and H. R. Wenk, "Texture, Anisotropy," Cambridge University Press, Cambridge, 1998.

[23] T. Markussen, X. Fu, L. Margulies, E. M. Lauridsen, S. F. Nielsen, S. Schmidt and H. F. Poulsen, *An algebraic algorithm for generation of three-dimensional grain maps based on diffraction with a wide beam of hard X-rays*, J. Appl. Cryst., **37** (2004), 96–102.

[24] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller, *Equation of state calculations by fast computing machines*, J. Chem. Phys., **21** (1953), 1087–1092.

[25] L. Meister and H. Schaeben, *A concise quaternion geometry of rotations*, Math. Methods Appl. Sci., **28** (2005), 101–126.

[26] A. Morawiec, "Orientations and Rotations. Computations in Crystallographic Textures," Springer-Verlag, Berlin, 2004.

[27] H. F. Poulsen, "Three-Dimensional X-ray Diffraction Microscopy," Springer-Verlag, Berlin, 2004.

[28] H. F. Poulsen and X. Fu, *Generation of grain maps by an algebraic reconstruction technique*, J. Appl. Cryst., **36** (2003), 1062–1068.

[29] H. F. Poulsen, S. F. Nielsen, E. M. Lauridsen, S. Schmidt, R. M. Suter, U. Lienert, L. Margulies, T. Lorentzen and D. Juul Jensen, *Three-dimensional maps of grain boundaries and the stress state of individual grains in polycrystals and powders*, J. Appl. Cryst., **34** (2001), 751–756.

[30] L. Rodek, H. F. Poulsen, E. Knudsen and G. T. Herman, *A stochastic algorithm for reconstruction of grain maps of moderately deformed specimens based on X-ray diffraction*, J. Appl. Cryst., **40** (2007), 313–321.

[31] L. Rodek, H. F. Poulsen, E. Knudsen and G. T. Herman, *Discrete tomographic reconstruction of 2D polycrystal orientation maps from X-ray diffraction projections using Gibbs priors*, Electron. Notes Discrete Math., **20** (2005), 439–453.

[32] S. Schmidt, S. F. Nielsen, C. Gundlach, L. Margulies, X. Huang and D. Juul Jensen, *Watching the growth of bulk grains during recrystallization of deformed metals*, Science, **305** (2004), 229–232.

[33] S. Schmidt, U. L. Olsen, H. F. Poulsen, H. O. Sørensen, E. M. Lauridsen, L. Margulies, C. Maurice and D. Juul Jensen, *Direct observation of 3D grain growth in Al-0.1%Mn*, Scripta Mater., **59** (2008), 491–494.

[34] R. M. Suter, D. Hennessy, C. Xiao and U. Lienert, *Forward modeling method for microstructure reconstruction using X-ray diffraction microscopy: Single-crystal verification*, Rev. Sci. Instr., **77** (2006), 123905 (12 pages).

[35] B. E. Warren, "X-Ray Diffraction," Dover Publications, New York, 1990.

[36] G. Winkler, "Image Analysis, Random Field and Markov Chain Monte Carlo Methods," 2nd edition, Springer-Verlag, Berlin, 2003.

E-mail address: arun.kulshreshth@yahoo.com

E-mail address: aal@imm.dtu.dk

E-mail address: gabortherman@yahoo.com

E-mail address: erik.knudsen@risoe.dk

E-mail address: lajos.rodek@gmail.com

E-mail address: henning.friis.poulsen@risoe.dk