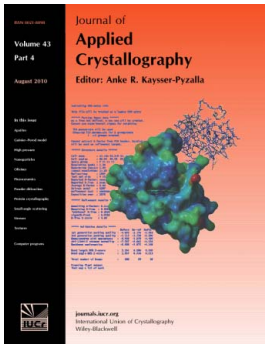
Journal of
Applied Crystallography
ISSN 0021-8898
Editor: **Anke R. Kaysser-Pyzalla**

The use of orthogonal projection to visualize mosaic domains from topographic data collected on protein crystals

Jeffrey J. Lovelace and Gloria E. O. Borgstahl

J. Appl. Cryst. (2010). **43**, 907–912



Many research topics in condensed matter research, materials science and the life sciences make use of crystallographic methods to study crystalline and non-crystalline matter with neutrons, X-rays and electrons. Articles published in the *Journal of Applied Crystallography* focus on these methods and their use in identifying structural and diffusion-controlled phase transformations, structure–property relationships, structural changes of defects, interfaces and surfaces, *etc.* Developments of instrumentation and crystallographic apparatus, theory and interpretation, numerical analysis and other related subjects are also covered. The journal is the primary place where crystallographic computer program information is published.

Journal of
Applied Crystallography
ISSN 0021-8898

Received 25 February 2010
Accepted 27 May 2010

The use of orthogonal projection to visualize mosaic domains from topographic data collected on protein crystals

Jeffrey J. Lovelace and Gloria E. O. Borgstahl*

Eppley Institute for Research in Cancer and Allied Diseases, Omaha, NE 68198, USA. Correspondence e-mail: gborgstahl@unmc.edu

Owing to the relatively low intensity of diffraction and resulting low contrast topographic images, and also the richly detailed domain structure, it can be difficult to extract individual mosaic domains (size and shape) for protein crystals. Here, orthogonal projection was tested and found to be a superior approach over previous methods to extract mosaic domain information from a fine-sliced topographic sequence of images. The topographic sequence was collected at room temperature from a lysozyme crystal with a low mosaicity. Orthogonal projection can be applied to image sequences that are spatially invariant and composed of linearly additive image formation processes. The domains were determined by using a particle swarm optimizer to fit Gaussians to the integrated intensity profile of the reflection as a function of angle, although any basis function could have been used. This optimization method was more amenable to automation and converged to the global minimum more efficiently. The number of domains can normally be determined by visually inspecting the integrated intensity profile; this only needs to be done once for each crystal, because the number of domains remains constant during collection, and not for each reflection. The results can be used to provide a picture of the internal structure of the crystal and may be useful to aid in the improvement of crystal growth techniques.

1. Introduction

Fine φ slicing (Fourme *et al.*, 1995; Helliwell, 1988; Snell *et al.*, 1995; Vahedi-Faridi *et al.*, 2003; Snell *et al.*, 2001) and topography (Fourme *et al.*, 1995; Stojanoff *et al.*, 1996; Otálora *et al.*, 1999; Boggon *et al.*, 2000) can be used to study mosaic domains of crystals. In fine φ slicing the mosaic domain analysis is performed by fitting Gaussians or other peak functions to the integrated intensity rocking curves. The collection of functions can then be studied in terms of width. The overriding premise for this type of analysis is that a crystal is composed of a collection of nearly perfect domains that are slightly misaligned with respect to each other (Darwin, 1922), resulting in a complex rocking curve. Fine slicing with an area detector allows for the simultaneous collection of multiple reflections, thus forming a statistically significant pool (Bellamy *et al.*, 2000). This gives a good description of the overall quality of a crystal but still does not provide comprehensive details about how the domains are distributed in the crystal or the shape of these domains.

Topography provides an in-depth view of the internal crystal structure. Although topography using film (large surface area, high resolution) can simultaneously record many topographs (Stojanoff & Siddons, 1996; Stojanoff *et al.*, 1997) there can be challenges in digitizing and registering the resulting sequences. Currently, digital topography is only suitable for a single reflection or a very small number of reflections because of the small active area of the CCD. By collecting a sequence of finely sliced topographic images of an individual reflection a very detailed picture of the crystal appears. Other than some very simple metrics, much of the information can only be interpreted in a qualitative fashion (Lovelace *et al.*, 2006). Our goal in this paper is to take the domain information that is extracted from profile fitting and use it to separate the various domains of the crystal from the topographic data.

One possibility is to use an orthogonal projection, sometimes referred to as simultaneous diagonalization (Miller *et al.*, 1982), which will create a projection from the data set by combining a signature vector that is unique to the desired information with signatures that need to be repressed. From the signatures, a filter vector can be found that is as close as possible to parallel to the desired signature while at the same time being orthogonal to the undesired signatures. This approach is commonly used in multispectral remote sensing (observation satellites) to break down the imagery into different components (Shayesteh *et al.*, 1989). For orthogonal projection to be successful the image sequence must be spatially invariant and made up of linearly additive image formation processes (Miller *et al.*, 1992). For fine-sliced topographic sequences collected on a high-quality room-temperature crystal the amount of movement (spatially invariant component) is basically negligible based on the amount of sample rotation. The image formation processes are linearly additive (each domain diffracts independently of the other domains) as long as the CCD is not saturated. Since our image sequence was basically close to these two restraints then this process may work. Previous applications did not specify how close to these conditions a sequence needed to be for the results to be acceptable. In fact the orthogonal projection filter worked better than expected because our results

show that for high-quality room-temperature data the mosaic domains can be separated.

2. Experimental methods

2.1. Image sequences

The image sequences presented are from data collected at BioCARS (Advanced Photon Source, Argonne National Laboratory). Crystallization, sample mounting, beamline parameters and other detailed information related to the experiment can be found elsewhere (Lovelace *et al.*, 2006). The beam was configured for parallel radiation and the crystal was fully enveloped by the beam. The topographic image sequences presented are 1 s exposures with an 8 × 8 µm pixel size CCD, using a 10 bit resolution, and were captured as 0.005° stills. Using wavelet resolution enhancement the images have been rendered with a 4 × 4 µm pixel resolution.

2.2. Eigenimage filtering

The goal of our image processing was to extract information about a single domain from the existing angular-based sequence of images. The ideal filter would be one that extracts the domain we are interested in and at the same time represses the other undesired domains and features in the data set. Effectively we want to maximize the ratio of signal (desired features) to noise (other features and noise) as shown in equation (1).

$$R_{\rm e}(x) = E_{\rm d}/(E_{\rm u} + E_{\rm n}). \tag{1}$$

$R_{\rm e}$ is the energy ratio as a function of the filter vector (x). $E_{\rm d}$ and $E_{\rm u}$ are the energies of the desired and undesired components, and $E_{\rm n}$ is the energy associated with the noise. If we think of $\mathbf{x}$ as a vector then we can rewrite equation (1) in the following vector format:

$$R_{\rm e}(\mathbf{x}) = \frac{\mathbf{x}^{\rm t}\mathbf{D}\mathbf{D}^{\rm t}\mathbf{x}}{\mathbf{x}^{\rm t}(\mathbf{U}\mathbf{U}^{\rm t} + w\mathbf{I})\mathbf{x}}, \tag{2}$$

where

$$\mathbf{D} = \left[\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_a\right] \tag{3}$$

is a matrix of desired signature vectors,

$$\mathbf{U} = \left[\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_b\right] \tag{4}$$

is a matrix of undesired signature vectors, w is a term representing the Gaussian white noise in the data and $\mathbf{I}$ is the identity matrix.

Substituting $\mathbf{A}$ for $\mathbf{D}\mathbf{D}^{\rm t}$ and $\mathbf{B}$ for $(\mathbf{U}\mathbf{U}^{\rm t} + w\mathbf{I})$ and rearranging equation (2) yields

$$\mathbf{B}^{-1}\mathbf{A}\mathbf{x} = R_{\rm e}(\mathbf{x})\,\mathbf{x} \tag{5}$$

or just the generalized eigenvector problem. In this case the filter vector ($\mathbf{x}$) we are searching for is the eigenvector ($\mathbf{x}$) with the largest eigenvalue [$R_{\rm e}(\mathbf{x})$] from equation (5). This type of filter is known as eigenimage filtering because the filtered image is obtained with an eigenvector. For our case we will always have only one desired process (a single domain, so $\mathbf{D} = \mathbf{d}$) and many undesired processes (everything else). Additionally, since we are only interested in the eigenvector for the largest eigenvalue it is possible to exactly determine and directly calculate this solution without having to perform a tedious eigenvector calculation. The eigenvector can be calculated as

$$\mathbf{x} = \mathbf{I} - \mathbf{U}(\mathbf{U}^{\rm t}\mathbf{U} + w\mathbf{I})^{-1}\mathbf{U}^{\rm t}\mathbf{d} \tag{6}$$

and the eigenvalue can be calculated as

$$R_{\rm e}(\mathbf{x}) = \frac{1}{w}\mathbf{d}^{\rm t}\Big[\mathbf{I} - \mathbf{U}(\mathbf{U}^{\rm t}\mathbf{U} + w\mathbf{I})^{-1}\mathbf{U}^{\rm t}\mathbf{d}\Big]. \tag{7}$$

A rigorous mathematical proof of the derivation of equations (6) and (7) from equation (5) is given by Miller *et al.* (1992).

2.3. Pixel vectors

Now that we have a method to determine the filter that will maximize the signal to noise for a given collection of processes, how do we apply this filter to the topographic image sequence to extract domain images? It is necessary to rethink the organization of the image sequence. Typically an image sequence is thought of as a collection of pictures where an image is selected from the sequence and then a pixel is selected from that image to determine the intensity as in the following equation (see Fig. 1*a*):

$$\text{Intensity} = \mathbf{Image}_i(x, y). \tag{8}$$

An alternative approach is to think of the image sequence as a collection of pixel vectors where a location is first selected on the image that contains a vector of all of the intensity values for that pixel over the entire sequence as in the following equation (see Fig. 1*b*):

$$\text{Intensity} = \mathbf{PixelVector}_{x,y}(i). \tag{9}$$

The two representations are equivalent but the pixel vector form allows the sequence to be easily filtered by taking the dot product of the filter vector with each of the pixel vectors. The only thing left to do is determine the signature vectors for the desired and undesired processes.

2.4. Particle swarm optimization

Protein crystals have interesting domain structures that are not observed in small-molecule crystals primarily because small-molecule crystals are of a superior quality. Each of the domains in the protein crystal acts as an independent image formation process in the images of the sequence. In the intensity profile they are manifested as peaks, which can be fitted with peak functions. In this case we chose to use Gaussians because they seemed to provide a reasonable fit to the intensity profiles from previous experience of fitting thousands of super-fine-sliced reflection profiles (Bellamy *et al.*, 2000; Vahedi-Faridi *et al.*, 2003; Borgstahl *et al.*, 2001; Juers *et al.*, 2007; Snell *et al.*, 2001). Each Gaussian represents the contribution of an individual domain to the entire intensity profile. From the fit it is known how the domain contributes but not necessarily what regions of the image are

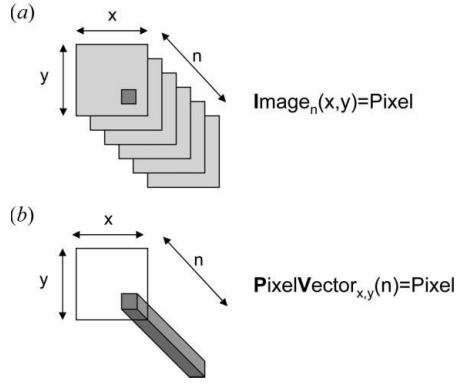


Figure 1
Two representations of an image sequence. (*a*) Normal approach where a pixel is located within the sequence by first selecting an image and then selecting a pixel on that image. (*b*) Pixel vector approach where a pixel is located by first selecting a pixel vector and then selecting an element within the pixel vector.

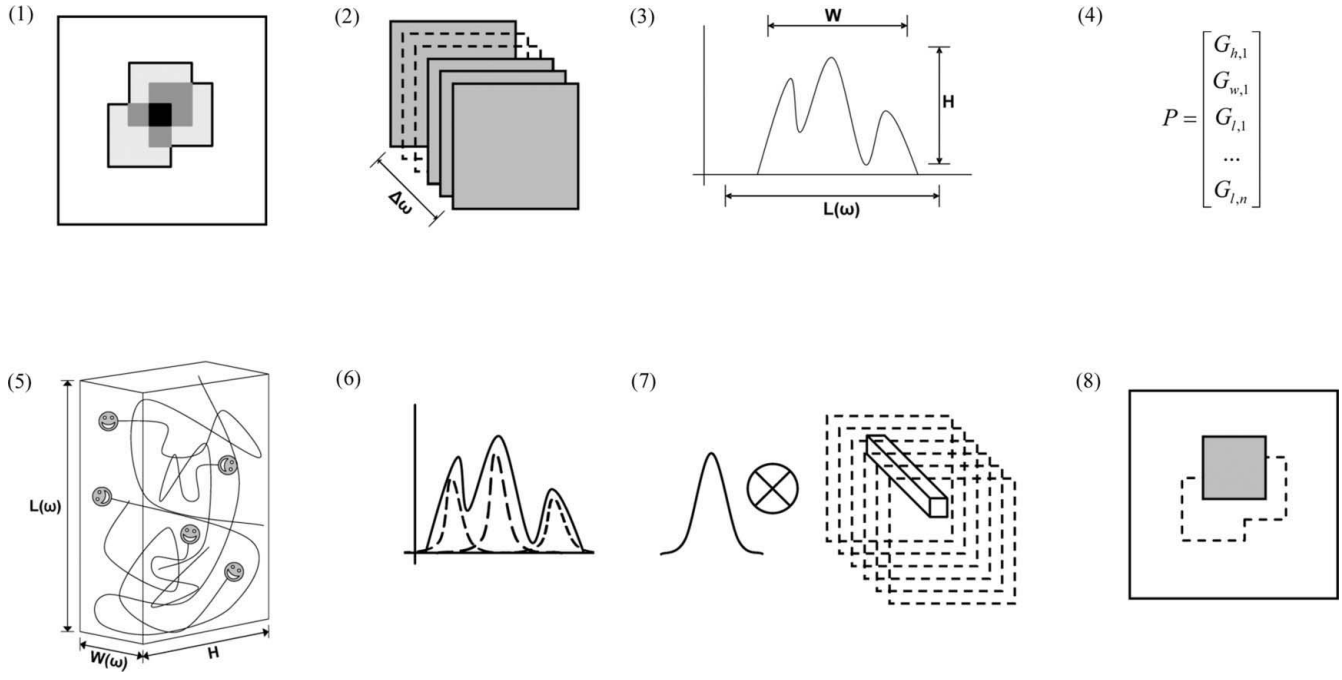


Figure 2
Processing steps to extract the domains with the orthogonal projection approach. (1) The reflection; (2) reflection broken down into the image sequence as a function of angle; (3) intensity profile as a function of angle; (4) particle representation of a Gaussian fit; (5) particle swarm optimization: the particles really enjoy minimizing cost functions as imagined by the authors; (6) final intensity profile fit as a collection of Gaussians; (7) a signature is used to perform the orthogonal projection operation over all of the pixel vectors in the image sequence; (8) an individual domain is extracted from the reflection sequence.

contributing. The automatic fitting of Gaussians to arbitrary peak shapes is a difficult nonlinear fitting problem. Previously we have used Levenberg–Marquett (Marquardt, 1963) and genetic algorithms (Holland, 1975; Wormington *et al.*, 1999) to perform this type of peak fitting. For this study a particle swarm optimizer was used to fit the peaks (Kennedy & Eberhart, 1995). The particle swarm optimization was selected because it is very simple to code as compared to the Levenberg–Marquett approach and is easy to extend for parallel execution on modern multi-core processors. Although similar in complexity to the genetic algorithm, it seems to be able to work itself out of local minima better. Particle swarm optimizers mimic the behavior observed in swarms of insects. They operate with a very simple and straightforward set of equations and random numbers. The solution parameters can be constrained. The two basic formulas governing the particle swarm are

$$\mathbf{V}_{n+1,i} = \mathbf{V}_{n,i} + C1\,(\text{Rand})\left(\mathbf{S}_{\text{localbest},i} - \mathbf{P}_i\right) + C2\,(\text{Rand})\left(\mathbf{S}_{\text{globalbest}} - \mathbf{P}_i\right) \quad (10)$$

and

$$\mathbf{P}_{i+1} = \mathbf{P}_i + \mathbf{V}_{n+1,i}, \quad (11)$$

where $\mathbf{V}_{n,i}$ is the velocity at iteration n for particle i, $C1$ and $C2$ are learning factors (the value is typically 2 for both), Rand is a random number between 0 and 1 updated at every iteration for every particle, $\mathbf{S}_{\text{localbest},i}$ is the best solution this particle has ever discovered, and $\mathbf{S}_{\text{globalbest}}$ is the best solution ever discovered by the swarm.

Equation (10) represents the velocity of each particle in the swarm and equation (11) represents the position of the particle in the swarm. The swarm is initialized with random positions. The initial best-fit values are determined from those positions. The state of the swarm is updated by repeating equations (10) and (11) for each particle and updating the best-fit values as better solutions are located. This occurs until a maximum number of iterations are reached or a minimum number of criteria are matched.

The solutions (particles) of the swarm were defined as

$$\mathbf{P} = \left[\, G_{\text{h},1} \quad G_{\text{w},1} \quad G_{\text{l},1} \ \dots\ G_{\text{h},n} \quad G_{\text{w},n} \quad G_{\text{l},n} \,\right]^{\text{t}}, \quad (12)$$

where each of the Gaussians was described by three parameters – height (h), width (w) and location (l) – and the number of Gaussians in the solutions was determined by n. Based on the overall intensity profile n was set to 3.

Equation (10) was modified to

$$\mathbf{V}_{n+1,i} = \mathbf{V}_{n,i} + C1(n)\,(\text{Rand})\left(\mathbf{S}_{\text{localbest},i} - \mathbf{P}_i\right) + C2(n)\,(\text{Rand})\left(\mathbf{S}_{\text{globalbest}} - \mathbf{P}_i\right) + \text{Pert}(n), \quad (13)$$

where $C1$ and $C2$ were throttled as a function of improvements in the fit values. This allowed the swarm to concentrate and quickly take advantage of improved solutions. It also allows the swarm to disperse and search wider areas of solution space when local improvements were not significant. Pert is a small random perturbation added to the velocity. It allows for a more efficient searching of solution space and more closely models the randomness found in swarms. Additionally we used a method to split the swarm into hunter and gather particles. Hunter particles spend more time away from the best solution in the hope of striking gold and finding some new superior solution, while gatherers spend their time milling around the current best global solution trying to incrementally improve it. These two types of particles are visible in the fitting movie provided as supplementary material.[1]

[1] Supplementary material for this paper is available from the IUCr electronic archives (Reference: HE5480). Services for accessing this material are described at the back of the journal.

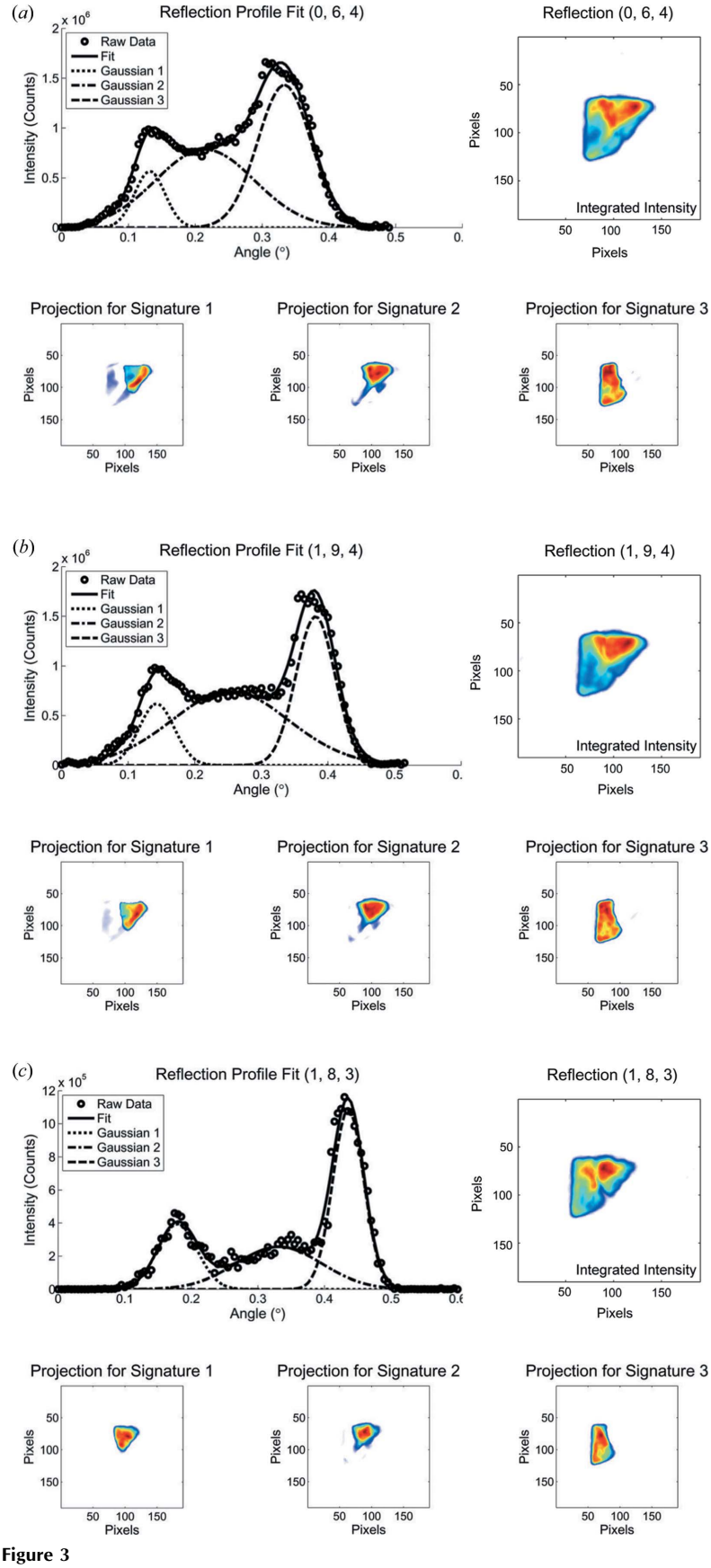


Figure 3
Reflection data showing the reflection profile (raw data, total fit and Gaussians), integrated intensity image of the reflection and orthogonal projections of the three domains for each reflection (blue represents low values and red represents high values). (*a*) RT1, (*b*) RT2 and (*c*) RT3.

2.5. Image processing and filtering

The fine-sliced topographic image sequences were processed as previously reported (Lovelace *et al.*, 2004). Each frame was integrated into the sequence to construct an intensity profile. Multiple Gaussians were fitted to the intensity profile using the particle swarm optimizer. The number of Gaussians used in profile fitting determined the number of filter vectors created with equation (6). For each filter vector one Gaussian was selected to be the desired signature and the others were used as the undesired signatures. Each filter vector was a dot product multiplied by every pixel vector. Values of the dot product indicate how well an individual pixel vector matched the filter vector. When arranged in a two-dimensional pattern, matching the image dimensions, the high values show where the domains defined by the selected signature Gaussian are located. Based on a visual inspection of the profile the fits were performed with three Gaussians. The complete process, from reflection to extracted domain, is shown in Fig. 2.

3. Results and discussion

We used three room-temperature data sets (fine-sliced topographic image sequences for the three reflections) for this study. The data sets are labeled RT1 for reflection (064), RT2 for reflection (194) and RT3 for reflection (183). The reflection images and profiles are shown in Fig. 3. The first challenge was to fit Gaussians to the intensity profile. This was accomplished with a particle swarm optimizer. We selected three Gaussians for the fit based on a visual inspection of the intensity profiles. Fig. 4 shows how the particle swarm optimizer fitted the Gaussians for RT1. A movie of a fit for RT3 is provided as supplementary material. In Fig. 4 each particle (solution) spans all three plots. The plots show the bounding box for the fit with respect to the minimum and maximum allowed values for the height (H), location (L) and width (W) for each Gaussian. The gray plus (+) signs are the starting locations of the swarm. The large black dots show the progression of the best solution the swarm has found. The small black dots are the final swarm locations at the time the optimizer reached one of the termination conditions. One of the initial particles (solutions) in the swarm has all three Gaussians the same size and spaced evenly along the intensity profile. Typically this particle also represents the best initial solution in the swarm. For the run shown in Fig. 4 this was also the case because the initial best solution for all three Gaussians is roughly at the center of the graphs.

One problem with using a constraining box is what to do with particles that wander outside of the box. Our solution was to allow some of the particles to stick to the wall about half of the time and the other half of the time randomly reposition them back in the allowed solution space. This kept the particles from being trapped next to the edge while at the same time allowing solutions that were close to the edge of the box to be explored.

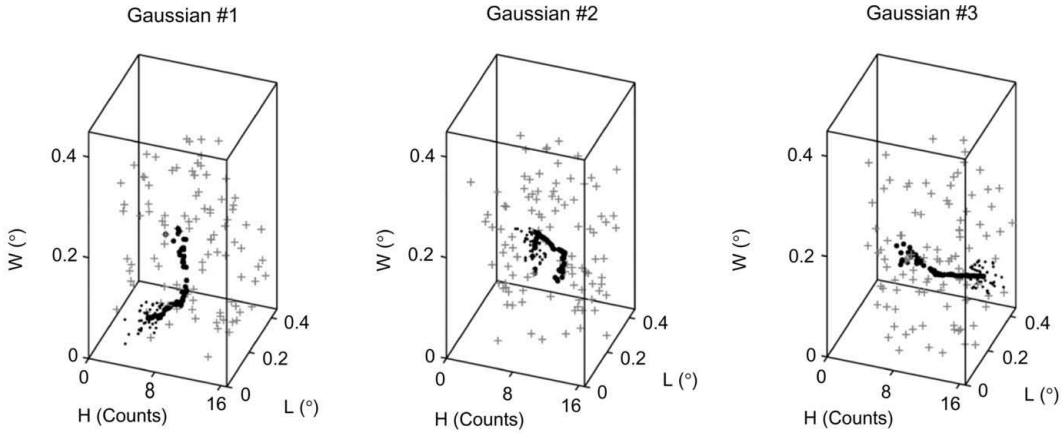


Figure 4
Particle swarm results for RT1 profile fitting. Each particle contains nine values (three for each Gaussian). There are three plots showing the evolution of the Gaussians during the particle swarm fit. Each plot shows the parameters for one of the Gaussians. All three are fitted at the same time. The gray plus signs show the starting positions of the particles in the swarm. The large black dots show the progression of best solutions discovered by the swarm. The small dots show where the particles were when the fit was terminated.

Once the Gaussians were fitted to the profiles the orthogonal projection was applied three times for each data set. In each case one Gaussian was selected as the desired signature and the other two were used as the undesired signature. The resulting filter vector from equation (6) was then multiplied (dot product) with all of the pixel vectors in the image sequence. The dot products were mapped to the proper location on the image and color coded according to intensity. Each signature extracted basically had the same shape and location relative to the original reflections for all three data sets. The domains defined by signatures 1 and 2 seem to be mostly overlapped with each other and then a separate third larger domain (described by signature 3) makes up the majority of the reflection.

Previously we simply color coded the angular location where the maximum intensity occurred (max@) within each pixel vector (Lovelace & Borgstahl, 2006). This max@ type of mapping shows the mosaic domains in the crystal. The results from that mapping are shown in Figs. 5(*a*)–5(*c*) for the three reflections. A problem with this approach is that it only shows the strongest domain for each pixel and it is not possible to visualize overlapping domains. Also domains are not explicitly labeled but inferred from regions that have a similar color. The orthogonal projections allow for a better estimate of the domains. Domains were simply defined as any values that mapped to dark blue or higher in the signature images. Composite images of all three domains overlaid for each reflection were generated. The images are shown in Figs. 5(*d*)–5(*f*). From these images it is clear that to some extent all three domains are overlapping each other. Domains 1 and 2 (as observed in the projections of signature vectors 1 and 2) overlap the most. Comparing this new domain map with the original domain map it appears as though domain 2 was the weakest diffracting region of the three, because in the area where there is overlap domains 1 and 3 are generally more visible in the original max@ domain map (compare Figs. 5*a* and 5*b* with Figs. 5*e* and 5*f*). Integrated intensity images (all intensities for a given pixel in the images sequence are summed together) for the three reflections are shown in Figs. 5(*g*)–5(*i*). The regions where there was overlap correspond to the areas with the highest integrated intensity values. This is easiest to see by comparing the domain map in Fig. 5(*f*) with Fig. 5(*i*). The location where only the weakest domain was present has the lowest intensity, but where domains 1 + 2 overlap and 2 + 3 overlap there is a large amount of intensity present.

In Figs. 3(*a*) and 3(*b*) the projection for domain 1 is not as intense in the overlapped region as it is in Fig. 3(*c*). For this approach to work the image formation processes need to be linearly additive. For these two reflections the intensity was high enough that it may have overloaded pixels in the CCD. Once a pixel in the CCD becomes overloaded then the image formation processes are no longer linearly additive, which makes it impossible to perfectly separate the domains. This effect manifests itself here where the overlapped region of

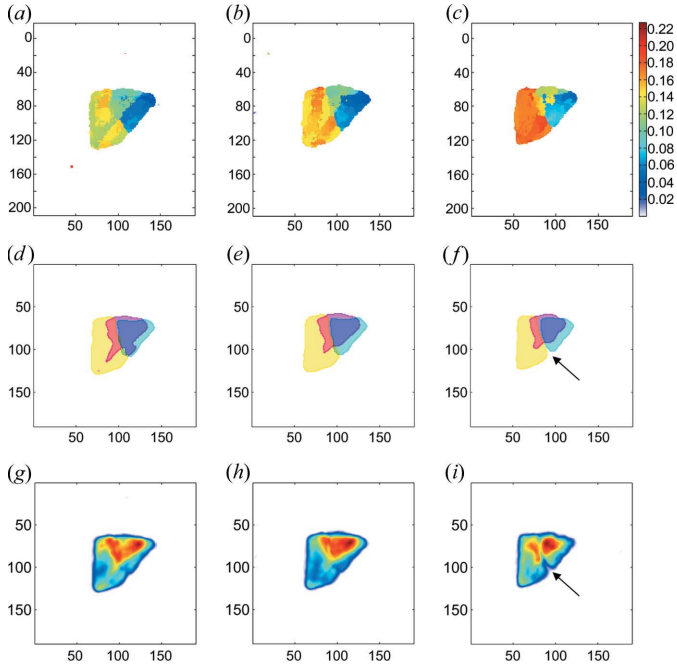


Figure 5
Comparison of different representations of the topography data. All of the axes values are in terms of pixels where each pixel was 4 × 4 µm. For parts (*a*)–(*c*) the color indicates at what angle during the image sequence the maximum intensity occurred (blue early, red later). For parts (*d*)–(*f*) the colors indicate the domain (blue for the domain defined by signature 1, red for the domain defined by signature 2, yellow for the domain defined by signature 3). For parts (*g*)–(*i*) the colors represent the integrated intensity for the entire image sequence for a particular pixel (blue lower values, red higher values). (*a*) Max@ for RT1, (*b*) max@ for RT2, (*c*) max@ for RT3, (*d*) orthogonal projection for RT1, (*e*) orthogonal projection for RT2, (*f*) orthogonal projection for RT3, (*g*) integrated intensity for RT1, (*h*) integrated intensity for RT2 and (*i*) integrated intensity for RT3. Arrows for (*f*) and (*i*) highlight the fact that the domain separation seen in the projection (*f*) is visible in the integrated intensity image (*i*).

domain 1 for reflections 1 and 2 does not have a very high value as compared to the non-overlapped region for domain 1. Reflection 3, which was a much weaker reflection (no overloaded pixels), does not have this problem: both domains are intense.

What happens if the wrong number of Gaussians is searched for? An example showing results for two, three and four Gaussians is provided in the supplementary material. When the number of Gaussians used was two, signature 1 is a combination of signatures 1 and 2 for the three-Gaussian case. Signature 2 in the two-Gaussian case looks identical to signature 3 in the three-Gaussian case. For four Gaussians the results are more complicated and begin to become dependent on the constraint conditions. If the minimum height is set to 1% of the maximum intensity then the best fit will look identical to the three-Gaussian case with a flat fourth Gaussian. The projections for this case do not make sense when compared with the fine-sliced frames. If the minimum height is set to 5% of the maximum, which will force the fit to use this Gaussian, then the resulting projections look the same as for the three-Gaussian case except that the projection for signature 3 in the three-Gaussian case is split between signatures 3 and 4 in the four-Gaussian case and the splitting does not match well with the data in the fine-sliced frames.

4. Conclusions

An improved method for automatically extracting domains from a fine-sliced sequence of topographic images was demonstrated. The only parameter that is required from the user is the number of domains to search for. There may be some metric that could be used to automatically select the number of domains, but with our limited data set it was not possible to accurately perform these tests. As long as the image sequence remains in the linearly additive and spatially invariant regime then the orthogonal projection approach should be effective at extracting domains from finely sliced topographic sequences. The particle swarm optimization algorithm was better at finding the global minimum than the Levenberg–Marquett and genetic algorithms used previously and was thereby more amenable to automation. The results demonstrate that orthogonal projection is a powerful approach for extracting domains, even when they are overlapped. This provides a much better way to characterize the multiple domains within protein samples than approaches reported previously when the topographs are collected as a super-fine-sliced sequence.

NASA grant NAG8-1983 supported this work. We would like to thank Jason Porta and Henry Bellamy for their support and important discussion.

References

Bellamy, H. D., Snell, E. H., Lovelace, J., Pokross, M. & Borgstahl, G. E. O. (2000). *Acta Cryst.* D**56**, 986–995.
Boggon, T. J., Helliwell, J. R., Judge, R. A., Olczak, A., Siddons, D. P., Snell, E. H. & Stojanoff, V. (2000). *Acta Cryst.* D**56**, 868–880.
Borgstahl, G. E. O., Vahedi-Faridi, A., Lovelace, J., Bellamy, H. D. & Snell, E. H. (2001). *Acta Cryst.* D**57**, 1204–1207.
Darwin, C. G. (1922). *Philos. Mag.* **43**, 800–829.
Fourme, R., Ducruix, A., Ries-Kautt, M. & Capelle, B. (1995). *J. Synchrotron Rad.* **2**, 136–142.
Helliwell, J. R. (1988). *J. Cryst. Growth*, **90**, 259–272.
Holland, J. (1975). *Adaptation in Natural and Artificial Systems.* Ann Arbor: University of Michigan Press.
Juers, D. H., Lovelace, J., Bellamy, H. D., Snell, E. H., Matthews, B. W. & Borgstahl, G. E. O. (2007). *Acta Cryst.* D**63**, 1139–1153.
Kennedy, J. & Eberhart, R. (1995). *International Conference on Neural Networks*, edited by IEEE, pp. 1942–1948. Piscataway: IEEE.
Lovelace, J. J. & Borgstahl, G. E. O. (2006). *J. Appl. Cryst.* **39**, 466–467.
Lovelace, J. J., Murphy, C. R., Pahl, R., Brister, K. & Borgstahl, G. E. O. (2006). *J. Appl. Cryst.* **39**, 425–432.
Lovelace, J. J., Soares, A. S., Bellamy, H. D., Sweet, R. M., Snell, E. H. & Borgstahl, G. E. O. (2004). *J. Appl. Cryst.* **37**, 481–485.
Marquardt, D. M. (1963). *J. Soc. Ind. Appl. Math.* **11**, 431–441.
Miller, J. W. V., Shridhar, M., Farison, J. B. & Shin, Y. (1992). *J. Vis. Commun. Image Representation*, **3**, 381–391.
Miller, J. W. V., Windham, J. P. & Kwatra, S. C. (1982). *Proceedings of the IEEE Conference on Pattern Recognition and Image Processing, Las Vegas, Nevada*, pp. 538–542. Piscataway: IEEE.
Otálora, F., García-Ruiz, J. M., Gavira, J. A. & Capelle, B. (1999). *J. Cryst. Growth*, **196**, 546–558.
Shayesteh, S., Farison, J. B. & Klingler, J. W. (1989). *IEEE International Conference on Systems Engineering, Dayton, Ohio*, pp. 609–613. Piscataway: IEEE.
Snell, E. H., Judge, R. A., Crawford, L., Forsythe, E. L., Pusey, M. L., Sportiello, M., Todd, P., Bellamy, H., Lovelace, J., Cassanto, J. M. & Borgstahl, G. E. O. (2001). *Cryst. Growth Des.* **1**, 151–158.
Snell, E. H., Weisgerber, S., Helliwell, J. R., Weckert, E., Hölzer, K. & Schroer, K. (1995). *Acta Cryst.* D**51**, 1099–1102.
Stojanoff, V. & Siddons, D. P. (1996). *Acta Cryst.* A**52**, 498–499.
Stojanoff, V., Siddons, D. P., Monaco, L. A., Vekilov, P. & Rosenberger, F. (1997). *Acta Cryst.* D**53**, 588–595.
Stojanoff, V., Siddons, D. P., Snell, E. H. & Helliwell, J. R. (1996). *Synchrotron Rad. News*, **9**, 25–26.
Vahedi-Faridi, A., Lovelace, J., Bellamy, H. D., Snell, E. H. & Borgstahl, G. E. O. (2003). *Acta Cryst.* D**59**, 2169–2182.
Wormington, M., Panaccione, C., Matney, K. M. & Bowen, D. K. (1999). *Philos. Trans. R. Soc. London Ser. A*, **357**, 2827–2848.