Scanning Microscopy

Volume 1992
Number 6 *Signal and Image Processing in Microscopy and Microanalysis*

Article 5

1992

Imaging Deformed Proteins: Characterising the State Fully

W. O. Saxton
Cambridge

Scanning Microscopy Supplement 6, 1992 (pages 53-70)
Scanning Microscopy International, Chicago (AMF O'Hare), IL 60666 USA

0892-953X/92$5.00+.00

IMAGING DEFORMED PROTEINS: CHARACTERISING THE STATE FULLY

W. O. Saxton

Department of Materials Science and Metallurgy,
Pembroke Street, Cambridge, CB2 3QZ, U.K.

Telephone Number: (44) 223-334566 / FAX number: (44) 223-334567

Abstract

By analysing the positions of individual unit cells in the image of a distorted macromolecular crystal, it is possible to achieve considerably more than is achieved by the correlation averaging or unbending now widely practised. It is possible partly to compensate for individual molecular distortion; and it is possible to identify molecules in equivalent environments (which can be expected to be in equivalent states of strain), selective averaging of which yields images that show how strain is accommodated at the sub-molecular level. The possible presence of surface forces applied to the crystal by its support film complicates the analysis and adds two additional parameters, not previously identified, to those necessary to characterise the environment of each molecule fully; these surface stress parameters can be estimated on the basis of a simple (isotropic) model of the elastic behaviour of a 2-D crystal. The appropriate mathematical description of strain and elasticity in 2-D crystals has been assembled concisely, and a set of new procedures developed allowing their practical exploitation within the Semper image processing system.

Key words: image averaging, correlation averaging, molecular distortion, strain, elasticity, unbending, surface stress, lattice fitting, gradient estimation.

1 Introduction

This paper has the following purposes: (a) to explain the principles of distortion characterisation and compensation in more detail than has been given previously; (b) to describe how these have been implemented in a Semper system (Saxton et al., 1979); (c) to present an analysis of the elastic behaviour of 2-D sheets in general; and in the light of this (d) to show how the stress at the crystal surface can be calculated.

The substantial information is presented in a series of appendices, while the main body of the text outlines their purpose and significance in the context of the complete process. It is hoped that describing the distortion characteristion and compensation processes in the context of a specific implementation will ensure an account sufficiently detailed for others to implement them independently, and provide useful supplementary information for anyone wanting to use the Semper code (available on request); there is perhaps also some modest didactic value in showing what can be written simply at a high level, as *library programs* of Semper commands (commonly called *macros* in other systems) and what must be written at a lower level, e.g. as Fortran.

Figure 1 illustrates what is simultaneously the problem and the opportunity, in the form of an image of a surface layer crystal from Pyrobaculum islandicum (Phipps et al., 1990) in which gross distortions are immediately evident. The basic process of averaging with local distortion compensation can be summarised in five stages, each of which is considered in more detail in turn below:

- o the observation of the distorted positions of all unit cells in the crystal, via conventional cross correlation of a reference with the distorted image;
- o the indexation of these positions in terms of the crystal base vectors

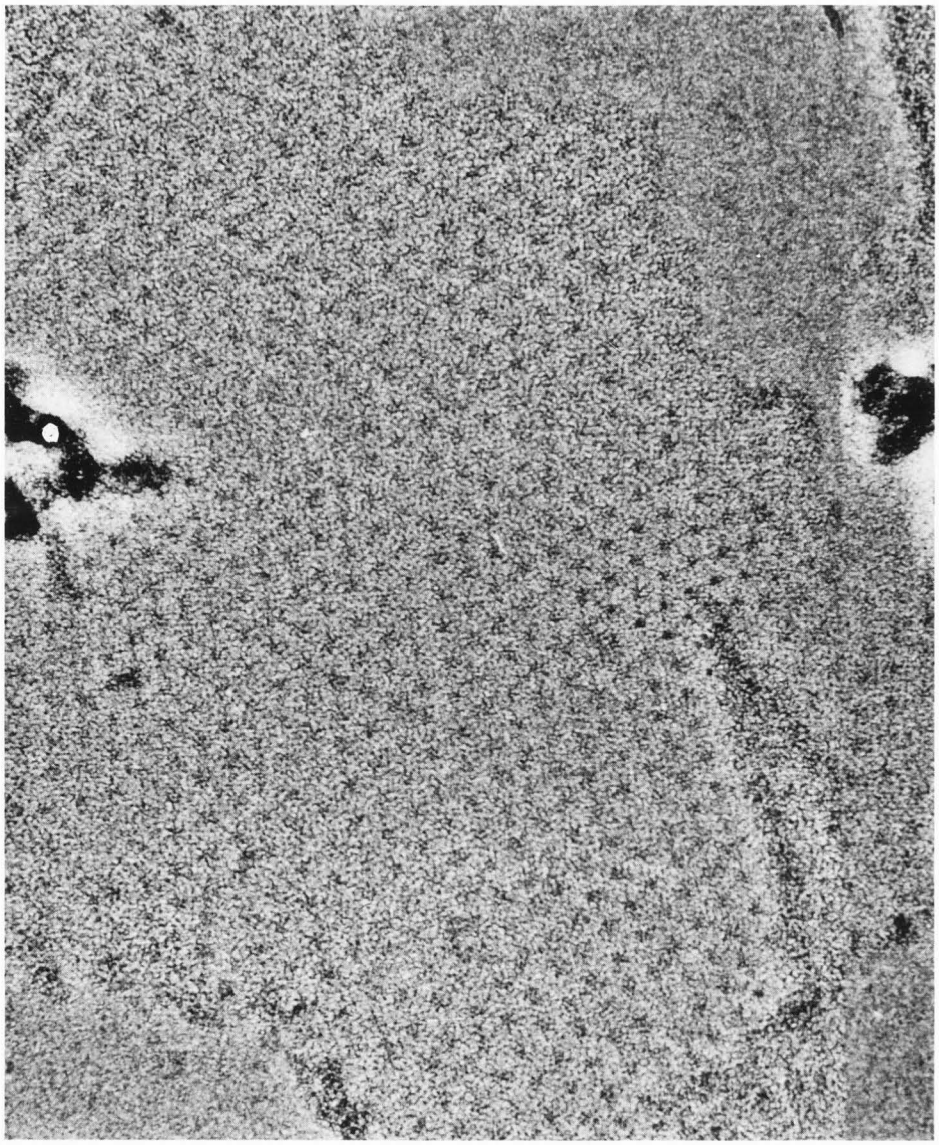

Fig.1 Micrograph of highly distorted P.islandicum crystal (Phipps et al., 1990).

- o the analysis of the distortions, with statistical and graphical display of the resulting parameters;
- o selection of positions to be averaged, on the basis of distortion parameters or otherwise;
- o averaging of subregions with local compensation of distortion.

Small superscript numerals refer to notes at the end of the paper.

2. Observing the Distortion Function.

It does not seem worth reiterating here any details of the initial correlation-averaging process that provides observations of the actual positions at which most unit cells of the distorted crystal are to be found; the procedure has been described before (e.g. Saxton & Baumeister, 1982, Frank, 1982).

It *is* worth reiterating however that the accuracy of the displacement measurements thus obtained is crucial to the value of any subsequent distortion analysis, since the distortion parameters are all based on estimating derivatives of the displacement field - and the method proposed here for estimating the surface forces between crystal and support film in fact involves differentiating it a second time. Care should be taken to ensure the maximum possible accuracy at this stage - by using an initial average as a

reference in a second refinement correlation process, by ensuring that the reference is no larger than it needs to be, and by devoting some effort to the process of estimating peak positions in the correlation function.

We assume here then that the distorted position data have already been obtained. They are stored by Semper in a relatively loosely defined 'picture' class called a *position list*, which consists of a multi-layer array in which corresponding pixels in different layers hold different parameters about a single position. The first two layers almost always hold an x- and y-coordinate, but the contents of subsequent layers vary widely: the correlation process normally stores correlation peak weights (the highest single point, or a locally integrated value) in a third layer; in the context of image analysis, object areas or perimeters may be stored in the other layers, and in the present context, parameters such as local rotation and magnification. The *order* of positions within each layer may or may not have any geometrical significance.

Some useful before indexing is effected. The list may be displayed in the form of a small cross at each listed site, and any spurious positions marked individually with a cursor or drawn round as a group deleted. It is not necessary however to remove positions that are well off the local lattice, as this form of pruning subsequently happens automatically; its value is in removing any large sets of positions outside the crystal area. New Fortran code providing a command PLDELETE has been provided for such purposes (Appendix G).

2.2 Indexing the positions listed

To determine the original position, before distortion, of each of the positions listed, we rely on the knowledge that each must be a site of the original crystal lattice, so that it is sufficient to know the lattice parameters and the indices of each position. Indexation is in fact one of the more difficult steps to automate reliably, especially in the presence of tears and/or tucks in the layer that move individual cells by many lattice constants from their original positions. The procedure described here relies on a reasonably convenient form of manual guidance around such problem areas; it is implemented entirely by Semper library programs.

An initial rough estimate is required of the lattice base vectors; this is found with quite sufficient accuracy by marking an origin and any two nearby independent sites, using a standard library program LATTICE. The user is asked to index these two sites directly, at which point the choice is made amongst alternative possible base vector pairs.

Thereafter, the list is indexed incrementally, in a series of overlapping regions each small enough to prevent mis-indexation internally, with the information accumulated from region to region in turn. The user is invited to outline a region with the cursor; within this the lattice base vectors **a** and **b**, and the position of its origin **c**, are adjusted to match the observed positions of lattice sites using a least-squares criterion as explained in Appendix D; a list of positions reasonably close to lattice sites (i.e. those whose lattice indices are nearly integral) is made, and the positions retained marked in a distinctive form on the display for the user's approval before being added to a cumulative list of positions retained for subsequent output in revised form.

The structure in which the retained positions are accumulated is a three layer position list, which we may call the *index-ordered* list; the three layers hold an x- and y-coordinate for a position, and the position weight (ia any. The layers are two-dimensional, with the point whose coordinates are h,k referring to the lattice site with those indices - so that the structure can be regarded as a table of actual position and weight against lattice indices. The layers are created large enough to accommodate the largest indices anywhere in the crystal - commonly a fixed size of 100 points square is used. The entire structure is initialised to a large number (10^6) which thereafter serves as a flag to indicate a site for which no information is available. The process of adding to this index-ordered list the positions retained for one fitted subregion involves deducing the indices of each position in turn (by calculating its indices rounded to the nearest integers), and storing its coordinates and weight at the point they indicate. The smallest and largest indices used in each direction are accumulated during this process also, for use in subsequent steps.

The general purpose of the index-ordered list is to provide an efficient way of locating a lattice site with given indices; its immediate purpose is the avoidance of duplication of sites in the final list, and other uses are described later as they arise.

For each region after the first, the user is asked to mark a *link point* within the region but already indexed; this serves as the reference point for the extension of indices into the new region. Since the lattice parameters are re-fitted to each region in turn, crystal distortions are normally followed without difficulty even when cumulatively large; on the occasions when it is necessary to move discontinuously to a new region where the local lattice is very different, the current lattice parameters must be reinitialised via the program LATTICE. When indexing is complete, the accumulated list is output as a five layer *indexed* list, in which the five layers hold the position coordinates, its weight (if any) and its two lattice indices.

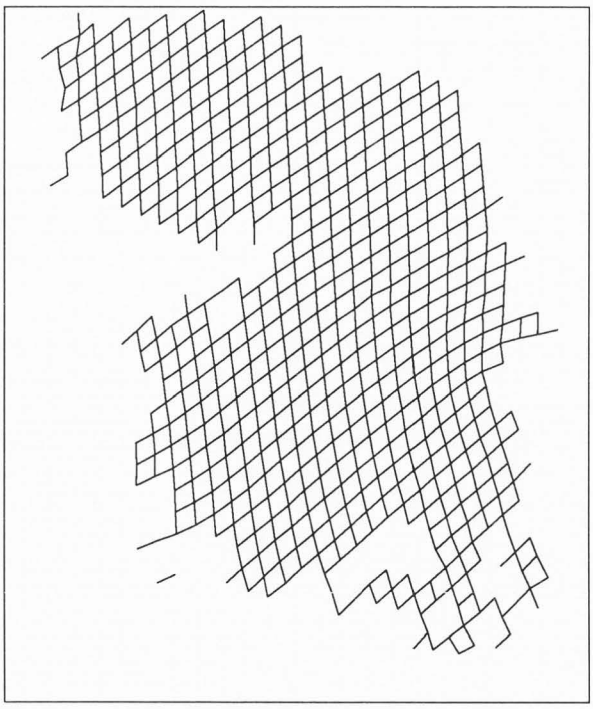

Fig.2 Distorted lattice for crystal shown in Fig.1, after indexation.

While the detailed description just given may appear complex, the process appears simple to the user, who merely runs a program INDPL to initialise the process, then INDPLADD for each region to be added in turn, and finally INDPLOUT to dump the results.

The index-ordered list allows two subsidiary programs to be used if desired to check the self-consistency of the indexing: INDPLTEST examines the neighbours of all listed sites and reports any for which the actual position recorded differs from that at the site itself by an amount that differs greatly from the appropriate lattice vector - though this criterion would be unsatisfactory if really large rotations were present; and INDPLMARK marks lines on the display joining the actual position of each listed site to those of its positive neighbours, which makes any misindexing at once visually obvious.

The indexation is illustrated in Fig.2, which shows the distorted crystal lattice, displayed as just described, for the layer shown in Fig.1

Whenever the user indicates a position with the cursor, e.g. as a link point, the complete list must be searched for the nearest position to the point indicated so that the appropriate rounding can be performed; new Fortran code now provides a command PLFIND for performing this search quickly (Appendix G).

2.3 Analysing the distortion function

The mathematical description of the distortion process, and how the observed distortion field can be analysed in principle to provide values for the local crystal rotation, magnification and elongation are set out in full in Appendix A.

To obtain such a description of each lattice site in the distorted crystal, the first step is to obtain for each site in the indexed list an estimate of either the deformation tensor D or the displacement tensor U described in the appendix; the present code finds the latter, i.e. the four derivatives formed by the rates of change of horizontal and vertical displacement components, each in both horizontal and vertical directions.

For each listed site, the list is searched for any neighbouring sites - currently taken to be those sites for which the indices differ from those of the target site by no more than one without both being identical with those of the target site, though this is not *entirely* satisfactory; a separate list is made of all such neighbouring sites, with their displacement differences between neighbour and target, and the original position differences between them. The original positions are of course all sites of the original ideal lattice, and their generation from the stored indices requires a decision as to what the ideal lattice parameters should have been; initially the estimates at the end of the indexing process can be used, as they are refined in a subsequent step.

The value of U can be fitted to the data in the neighbour list following any of the three procedures described in Appendix B; the present code (a Fortran module FITUIJ) uses the simplest only, and discards any sites that have less than two neighbours.

The resulting displacement tensor is then analysed by a separate Fortran module PSC2D into a local *rotation*[1] , an isotropic *magnification*, and an area-preserving *elongation* - magnification in one direction (the *elongation direction*) with simultaneous demagnification by the same factor at 90°. When each listed position has been analysed, a revised list is output containing the original five layers, four more containing the distortion parameters just calculated, and five more containing other useful but inessential information: the original position coordinates of each site, its displacement components, and a serial number to identify it through any subsequent re-ordering or sorting processes. The resulting fourteen layer position list may be called an *analysed* list.

Once again, the process appears extremely simple as well as rapid to the user, who merely assigns initial values to the lattice parameters and makes one use of a new Fortran-coded command PLANALYSE (Appendix G).

2.4 Examining the distortion parameters

Statistics can be obtained easily about the resulting parameters simply by examining the appropriate layers of the analysed list. The standard facilities of Semper are perfectly adequate for the purpose, allowing the generation of histograms as well as the calculation of means, standard deviations, and modal values. The definition of suitable Semper macros relieves the user from having to know or remember actual layer numbers: for example,.one types HISTOGRAM @MAGN rather than HISTOGRAM LAYER 11).

It is usually convenient at this stage to adjust the parameters used to describe the original ideal lattice. If the crystal is known to have p3, p4 or p6 symmetry on biochemical grounds, then the original lattice assumed should certainly have this symmetry, with any departure from it implying distortion[2] , so any analysis should be made with **b** forced to be 60° , 90°or 120° anticlockwise from **a** as appropriate. After a first analysis, it is convenient (though certainly not essential) to adjust the original lattice parameters so that the mean magnification after analysis is unity and the mean rotation is zero; this is achieved by multiplying the base vectors first used by the mean magnification reported, rotating them anticlockwise by the reported mean rotation, and then repeating the distortion analysis with PLANALYSE.

There are many ways in which the distortion parameters could be displayed graphically. Some parameters, are simple conceptually: the local displacement, for example, can be visualised as short arrows from each original site to the actual position (Fig.3), with lengths exaggerated if necessary, and the rotations by small fixed length lines at each site rotated appropriately. Others are less easily presented; we have found it useful to convey one particular pair simultaneously by placing small squares at each site, oriented so that their sides lie along the elongation direction, and distorted into the appropriate rectangles by the combined effects of magnification and elongation (also shown in Fig.3). Some Fortran code was written to provide a special command PLMARK for marking distortion parameters on the display (Appendix G).

2.5 Selecting positions to average

One way of using the distortion information is in the production of an average in which all the unit cells in the crystal are superimposed with the local distortions compensated. That this can achieve substantially improved resolution for badly distorted crystals is clear (see for example Fig.5 below); the compensation can only be complete however if the entire contents of the unit cell distort uniformly (the *rubber-sheet* model), and while at least one good

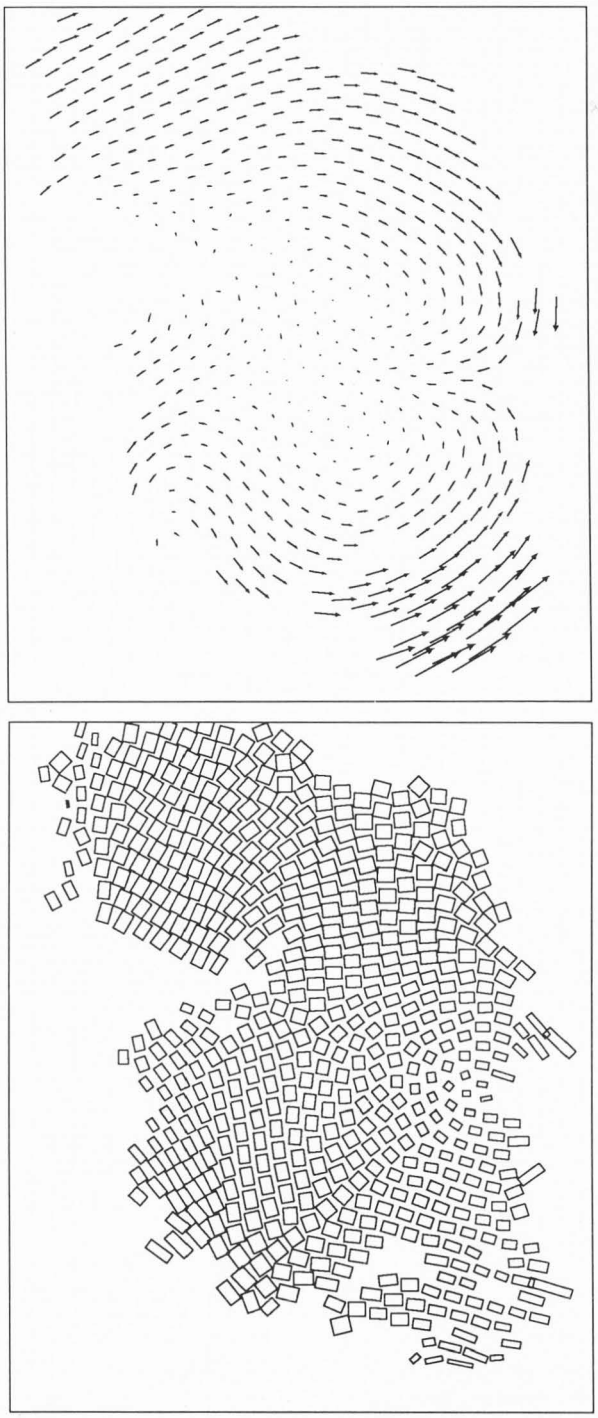

Fig.3 Above: displacement vectors for crystal shown in Fig.1, without exaggeration (i.e. at unit magnification). Below: magnification / elongation pattern, exaggerated five-fold.

example of such behaviour has been reported it is more likely in general that the strain will be distributed unequally over different features within the unit cells - some parts being more labile than

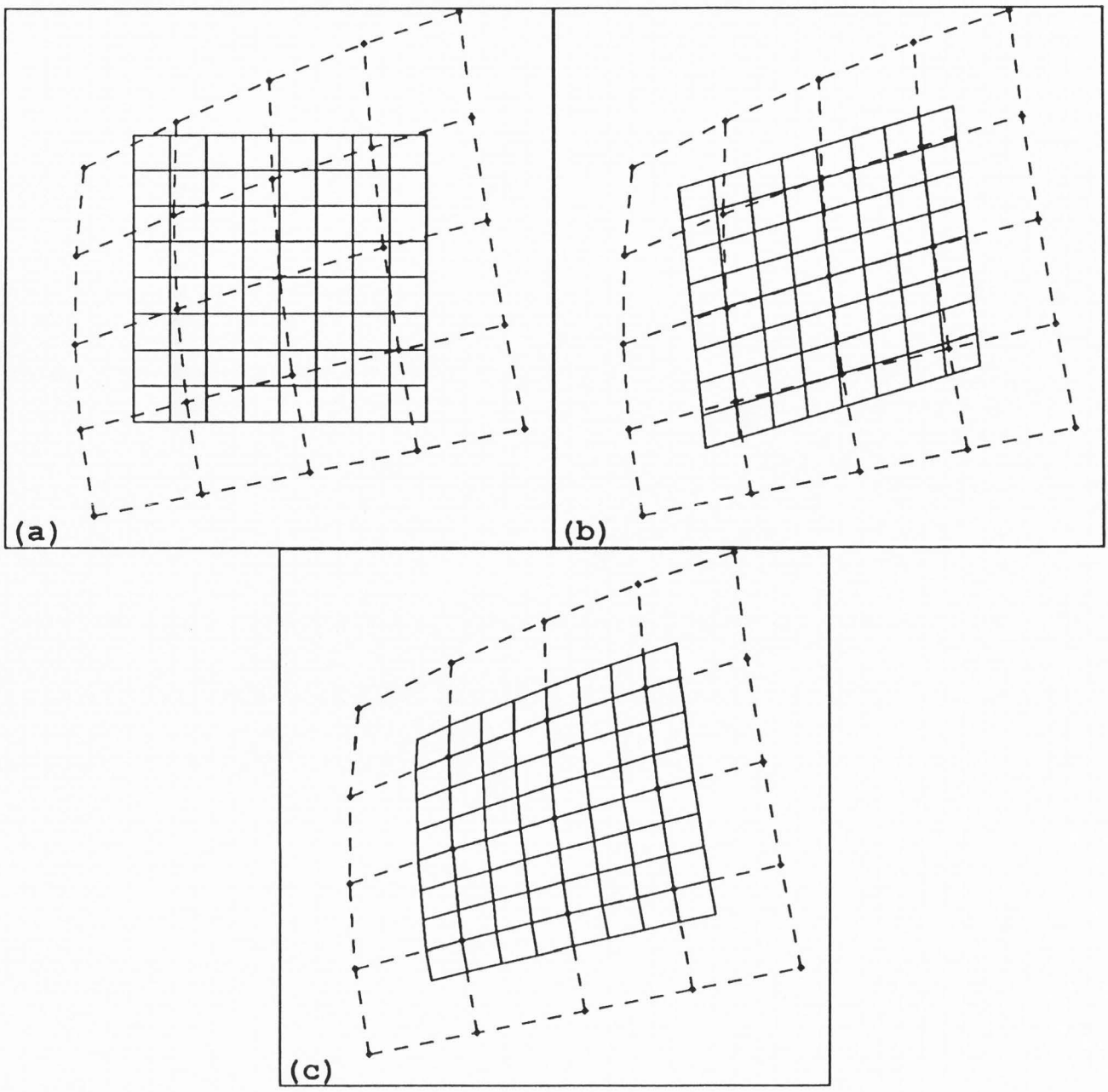

Fig.4 Resampling grids (solid lines) used in extracting regions for averaging in three ways from a distorted crystal lattice (broken lines). (a) conventional correlation averaging, correcting for displacement only; (b) constant distortion parameters over the region extracted; (c) locally varying distortion parameters - grid obtained by bilinear interpolation of the observed displacements.

others, for example, as reported by Saxton et al., 1991.

In these circumstances, it should be preferable to average only molecules in similar environments - and this is the real justification of all the effort devoted to *analysing* the distortions, which is unnecessary if the sole object is the production of distortion-compensated averages. With seriously distorted crystals, such an approach can be viewed entirely positively: clear averages of molecules in various states of substantial but known strain should show how the individual parts of the unit cell respond to the strain, providing information not otherwise available on the architecture of the crystal as a working layer - the curvature of bacterial surface layers for example alone implies substantial strain levels *in vivo* (several

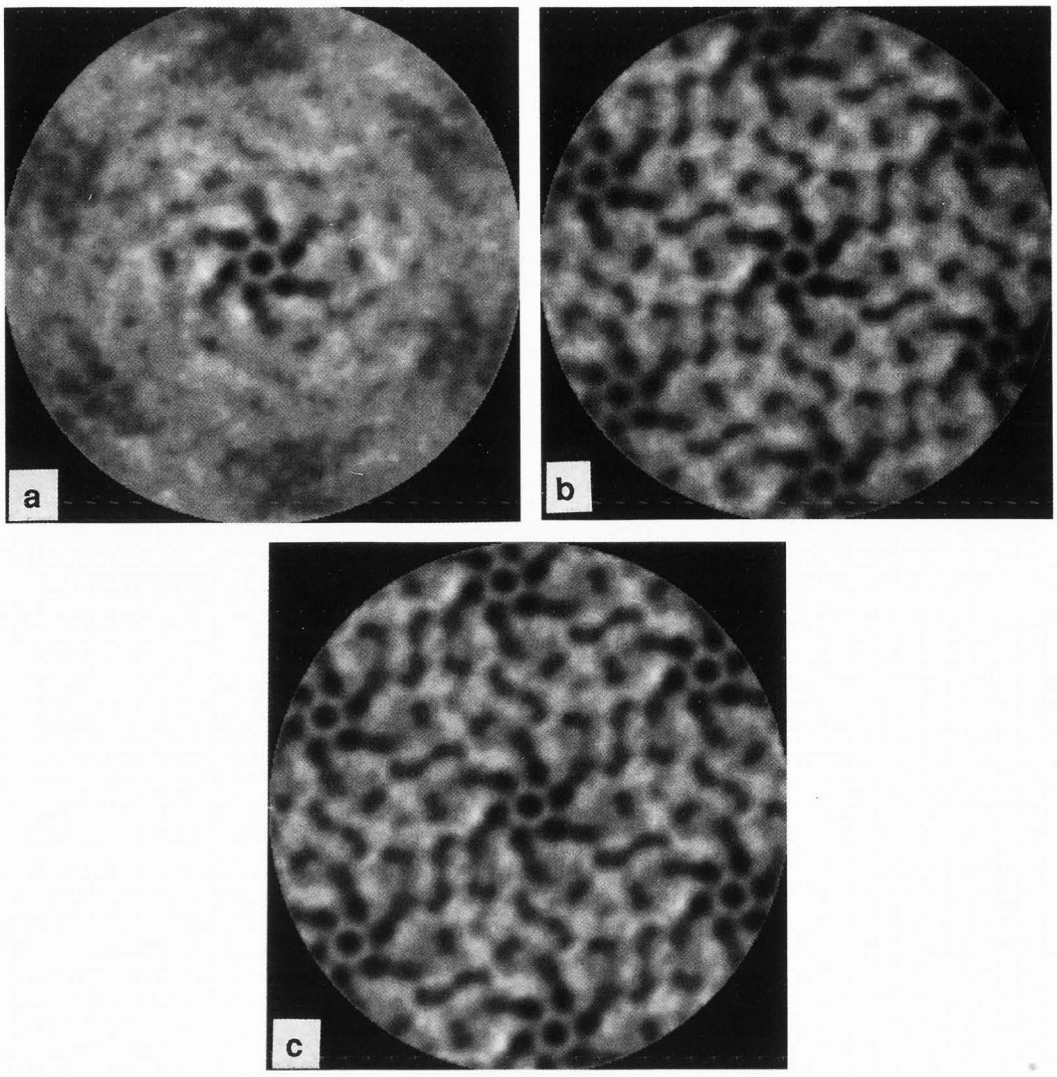

Fig.5 Averages made in three different ways of the most highly distorted region only of the crystal in Fig.1 (the region half way down the right hand side), containing about a quarter of the total unit cells; the three averages correspond to the three different resampling patterns shown in Fig.4.

per cent).

Selective averaging - which may be on other bases besides the distortion parameters - is of course the main reason why local averaging methods are generally to be preferred to complete lattice unbending and conventional Fourier filtration or transform peak profile fitting.

Two new Fortran-coded commands have been provided as tools for position selection, PLSORT and PLDELETE (Appendix G). The former simply reorders all the positions in a list, and all their associated parameters, so that any one chosen parameter is sorted into ascending or descending order; thus an average over the fifty least elongated regions (loosely, those with lowest shear strains) can be achieved by sorting the list with respect to the elongation and then averaging the first fifty points listed. The latter command, PLDELETE, provides a variety of mechanisms for selective deletion of positions from a list, from individual positions, groups within or outside indicated regions, and most usefully positions for which the parameters meet or satisfy some algebraic criterion. It is one of the felicitous results of the otherwise often frustrating way in which Semper handles text strings that facilities such as

```
PLDELETE IF MAG<.97 | MAG>1.03 | MOD(ROT)>.03
```

(i.e. "delete position if magnification factor is less than 0.97, if magnification is greater than 1.03, or if

the magnitude of the local rotation exceeds 0.03 rad") can be provided without any significant programming effort within the code implementing PLDELETE. In this way, it is simple to make separate lists of positions highly elongated in two different directions, or other similarly selective lists.

2.6 Effecting an average with local distortion compensation

In conventional correlation averaging, the only form of distortion compensated is the local displacement. Recent measurements for a variety of different specimens (Saxton et al., 1991) have confirmed that this is invariably the most serious in its effect on resolution when it is left uncorrected; displacements accumulate from unit cell to unit cell across the crystal, but once this is corrected features at the centre of the unit cells averaged are correctly registered regardless of the remaining forms of distortion present, and misregistration arises only in proportion to distance from the centre of the field averaged.

There seems little reason to average with any level of compensation intermediate between compensating for the displacement only, as in conventional correlation averaging, and compensating for all forms of distortion - for example, compensating for rotation as well without compensating for magnification and elongation, even though rotation is not uncommonly the next most serious form of distortion after local displacement. There are two ways in which a fuller degree of compensation can be incorporated easily in averages given the form of data analysis so far described. In the first, the distortion parameters are treated as constant throughout each region averaged: for each site listed, a re-sampling grid is determined either from the deformation tensor or from the distortion parameters deduced from it, as described in detail in Appendix C, a distortion-compensated region is extracted from the original picture on this grid, and the result is added to those already accumulated. The resampling grid is entirely specified by an origin and a pair of re-sampling lattice vectors (cf. Appendix C), and new Fortran code providing a command PLUVXY has been provided to recover these from the analysed position list (Appendix G).

The alternative approach is to use simple bilinear interpolation of the positions at which the distorted lattice sites were found to provide the necessary resampling grid; this allows the distortion parameters to vary within the regions averaged, and achieves a slightly better final result at relatively large distances from the centre of the field averaged. The interpolation can be achieved conveniently by exploiting the index-ordered list, reconstructed if necessary for the present purpose: the first two layers of this provide the x- and y-coordinates for the distorted lattice sites, tabulated in index order, so that interpolating each layer finely in turn provides the re-sampling grid needed to recover distortion-compensated regions sampled on a lattice-relative grid (i.e. one with base axes parallel to the lattice base vectors); a single final re-interpolation of the average is needed to restore normal cartesian sampling. A drawback to the approach is that any sites with missing neighbours cannot be included in the average.

Figure 4 shows the re-sampling grids used in the two approaches, together with one equivalent to what is achieved by conventional correlation averaging; Fig.5 shows averages of the surface layer in Fig.1 made in each of the three ways, with successive improvment evident at each stage; these are taken from Saxton et al. (1991). The user normally uses a Semper program to effect the averaging; all that is actually necessary however is a simple loop such as the following:

```
FOR N=0,100
   PLUVXY 2 NUMBER N
   EXTRACT 1 TO 3 @UVXY SIZE 80
   CALCULATE :4+:3

LOOP
```

which adds to picture 4 compensated regions around the first 100 positions of picture 1 listed in the plist 2

Although they are not provided for in the Semper code described here, two alternative approaches to the production of distortion-compensated averages should be mentioned. In both, the strategy is simply the interpolation of the observed unit cell displacements, with no distortion characterisation. Henderson et al. (1986) use a standard subroutine (from the NAG library) to fit bicubic splines to the displacement data; this effects some smoothing (which may or may not be desirable according to how noisy the data are), and interpolates values at unlisted sites; the process is unstable however where large gaps in the data exist, or where relatively accurate data demand close fitting with a large number of knots. Dürr et al. (1989) adopt a different approach, tabulating displacement values finely and adjusting them to minimise an estimate of the elastic deformation energy in the crystal subject to their matching the data available; this is very reliable in both interpolating and extrapolating the data in a physically realistic fashion.

Examples obtained using the above techniques of images of two bacterial surface layers, each in two different known high strain states, are given by Saxton et al. (1991).

3 External Stress at the Surface of a Crystal

The analysis presented above of the state of strain in a 2-D crystal; though it has already proved useful in allowing averages to be obtained in distinct known states of high strain, it is in fact an incomplete characterisation of the environment of a unit cell, if our purpose is to ensure that all unit cells averaged are (statistically at least) in identical configurations.

The factor remaining to be considered is the presence or otherwise of tangential forces applied to the crystal at its interface with the support film, as illustrated in Fig.6. Since they are applied asymmetrically - at the top or the bottom of the layer - any such forces are likely to encourage molecular subunits to tilt in an way not promoted by the in-plane forces associated with the strains we have considered so far. In fact, it proves possible to obtain good estimates of these surface stresses also from the strain fields already deduced.

To calculate the surface stress, we need first to know the relationship between stresses and strains in a 2-D crystal; these are summarised in Appendix E for general, p4 and isotropic sheets. The appendix shows that in the general case there are six independent elastic constants (stiffness coefficients) for a 2-D sheet; that a sheet with p4 symmetry has only 4 - but that is more than the seemingly analogous case of a cubic crystal in 3-D; and that an isotropic sheet in 2-D has only two, as in 3-D. The last case is tractable as a model, and Appendix F uses it to deduce the stress in a crystal from the strain with only one unknown multiplicative constant. The force applied to the surface of an element of the crystal is calculated from the difference in the in-plane stresses acting at opposite edges; the final expression for the surface stress involves both x- and y-derivatives of the three independent strain components.

Given this model, it is not difficult to supplement the strain parameters so far obtained by two more: the x- and y-components of the surface stress, and further Fortran code has been written recently to provide a new command PLXTRA for the purpose. The various derivatives are obtained in much the same way as the displacement derivatives (§2.3): the strain tensor for each site is recovered from the stored distortion parameters; the neighbours for each site are found, and the best-fitting value for the derivative of the three components is found as in Appendix B, after which the force is obtained directly from (F4). Figure 7 shows the surface stress field thus found for the *P. islandicum* crystal used in the earlier figures, displayed in the form of short vectors proportional to the local stress vector. Large stresses are found at some parts of the crystal periphery, making it

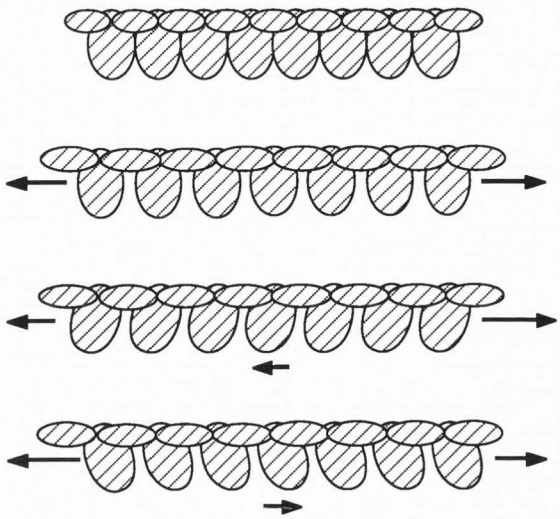

Fig.6 Surface stress on a crystal at the lower surface, where it is supposed to rest on a support film. Top: the unstressed crystal; second line: the crystal under tension, but without tension variations from one side to the other; third line: the crystal with the same mean tensile stress, but with a larger stress at the right; bottom: the same with the stress gradient reversed. In the last two cases, the imbalance in the forces at the side must be made up by forces at the lower surface, which distort the crystal in different ways as shown.

impossible simultaneously to show the stress clearly throughout the interior; the lower part of the figure accordingly shows the stress vectors enlarged further with the larger peripheral stresses suppressed. A reasonably smooth pattern of surface stress variations is discernible; for example, most unit cells near the highly distorted region half way down the right edge of the crystal are systematically driven towards the centre of this region by the surface forces. Selective averages allowing for variations in surface as well as internal stresses are expected to be the subject of later reports.

The results above are both welcome and unwelcome for different reasons. It can only be welcomed that a further factor significantly affecting the configuration of individual unit cells can be deduced from the observed strain fields; on the other hand, it is disappointing that there should be any such factor, especially one with two components, as there are already a rather large number of factors that must be similar before unit cells can safely be combined in an average with a high resolution target: the local displacement and rotation can be ignored, but magnification and elongation are both significant, quite apart from differences due to more familiar causes such as variations in stain level. It remains to be established where the most effective trade-off is to

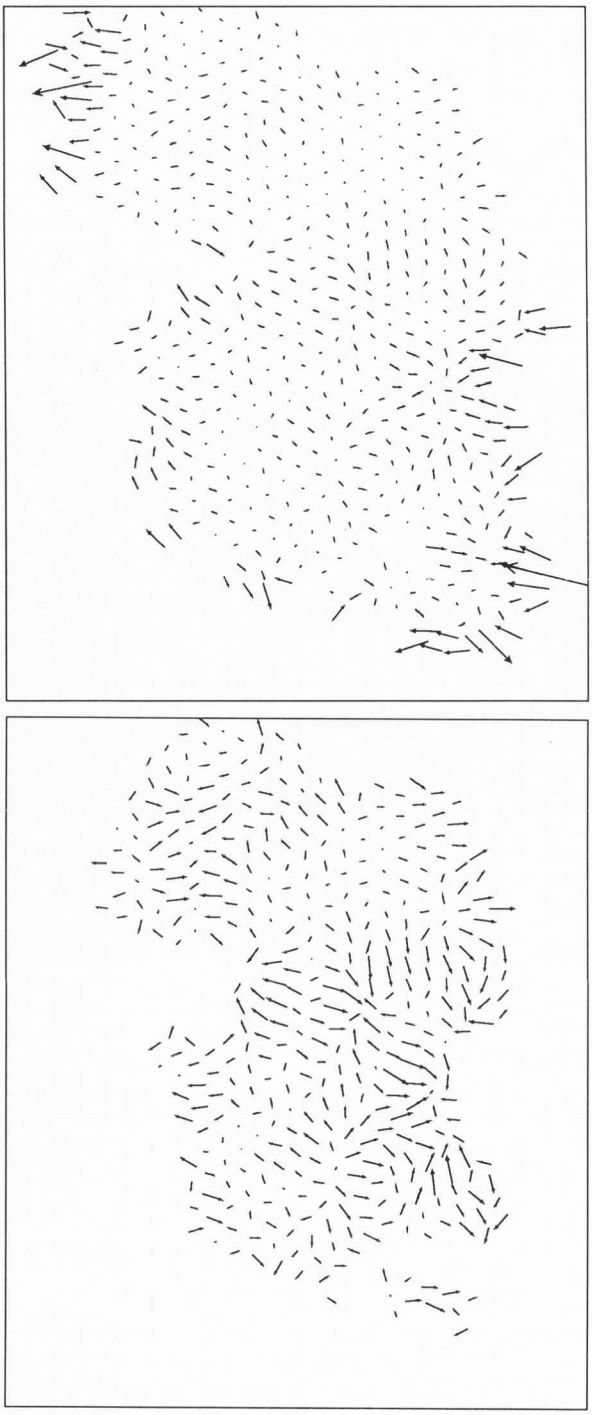

Fig.7 The surface stress field observed for the crystal in Fig.1, shown to an arbitrary scale. Above: stress at all recorded sites, showing relatively large values near the periphery; below: peripheral sites excluded, and stress vectors further magnified threefold.

be made between resolution loss due to inadequate characterisation and consequent misregistration, and high noise levels because too few unit cells have been averaged.

4 Acknowledgements

All but the most recent code described above was written on visits to (and the partial support of) the Max Planck Institute for Biochemistry, Martinsried, Germany. I am particularly grateful to R Dürr for introducing me to the problem addressed, for constant stimulation and frequent helpful suggestions.

Appendix A: Analysis of Finite Deformations in 2-D

A.1 Textbooks on elasticity theory or continuum mechanics abound, but deal almost entirely with 3-D materials and most commonly with infinitesimal displacements. Those by Billington & Tate (1981), Segel & Handelman (1977) and Spencer (1980) are useful in the present context. This appendix sets out how finite distortions are analysed in 2-D (see also Dürr, 1991); appendices E and F below deals with 2-D elasticity theory, where there are some significant differences from the standard 3-D case.

We suppose that a point of the 2-D medium originally at $\mathbf{r} = (x,y)$ is displaced to $\mathbf{r'} = (x',y')$, which we treat as a function of x and y - the *Lagrangian* formulation - and may call the *distortion* function. The difference between these, $\mathbf{u} = (v,w) = \mathbf{r'} - \mathbf{r}$, we call the displacement field. Any *small* vector $\delta\mathbf{r}$ in the original medium becomes $\delta\mathbf{r'}$ in the following linear way:

$$\begin{pmatrix}\delta x'\\ \delta y'\end{pmatrix} = \begin{pmatrix}\frac{\partial x'}{dx} & \frac{\partial x'}{\partial y}\\ \frac{\partial y'}{\partial x} & \frac{\partial y'}{\partial y}\end{pmatrix}\begin{pmatrix}\delta x\\ \delta y\end{pmatrix} = \begin{pmatrix}1+\frac{\partial v}{dx} & \frac{\partial v}{\partial y}\\ \frac{\partial w}{\partial x} & 1+\frac{\partial w}{\partial y}\end{pmatrix}\begin{pmatrix}\delta x\\ \delta y\end{pmatrix} \qquad (A1)$$

Defining $d_{ij} = \frac{\partial r_i'}{\partial r_j}$ and $u_{ij} = \frac{\partial u_i}{\partial r_j}$, this may be rewritten, with the usual suffix summation convention,

$$\delta r'_i = d_{ij}\,\delta r_j = (\delta_{ij} + u_{ij})\,\delta r_j. \qquad (A2)$$

D is called the *deformation* tensor, and U the *displacement* tensor. We now decompose the deformation D into a product RF, with R a (clockwise) rotation and F symmetric, i.e. into the application of a simpler deformation F followed by a rotation R; the decomposition is not unique without some such condition on F, and we shall see below why

this condition is useful. To find the appropriate angle of rotation θ, we note that $F = R^{-1} D$, i.e.

$$\begin{pmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{pmatrix}; \quad \text{(A3)}$$

our symmetry requirement $f_{12} = f_{21}$ therefore requires that

$$d_{12}\cos(\theta) - d_{22}\sin(\theta) = d_{11}\sin(\theta) + d_{21}\cos(\theta),$$

which allows us to obtain θ via

$$\theta = \tan^{-1}\{\frac{d_{12}-d_{21}}{d_{11}+d_{22}}\} = \tan^{-1}\{\frac{u_{12}-u_{21}}{2+u_{11}+u_{22}}\}. \quad \text{(A4)}$$

Once θ has been determined, F is obtained from (A3) as

$$F = \begin{pmatrix} \cos(\theta)(1+u_{11})-\sin(\theta)u_{21} & \cos(\theta)u_{12}-\sin(\theta)(1+u_{22}) \\ f_{12} & \sin(\theta)u_{12}+\cos(\theta)(1+u_{22}) \end{pmatrix} \quad \text{(A5)}$$

A.2 The *shape change* tensor F thus obtained describes all forms of local distortion other than rotation. Its diagonal components are said to measure *normal* or *tensile* change and its off-diagonal elements *shear*. Being symmetric, it takes the simpler form

$$\begin{pmatrix} f_1 & 0 \\ 0 & f_2 \end{pmatrix}, \quad \text{(A6)}$$

where f_1, f_2 are its eigenvalues, if the coordinate axes are rotated to be parallel to its eigenvectors; it describes stretching by a factor f_1 in the direction of the first eigenvector and by a factor f_2 at 90° to this. Any deformation can thus be regarded locally as pure stretching in these *principal* directions followed by pure rotation. It is the simplicity of this description that led us to seek a symmetric F when first decomposing D[3].

We find the actual values of the *principal magnifications* f_1, f_2 and the corresponding directions in the standard way, obtaining

$$f_{1,2} = \frac{1}{2}[(f_{11}+f_{22}) \pm \sqrt{(f_{11}-f_{22})^2 - 4f_{12}^2}] \quad \text{(A7)}$$

with the principal directions

$$\alpha_{1,2} = \tan^{-1}\{\frac{f_{1,2}-f_{11}}{f_{12}}\} \quad \text{(A8)}$$

anticlockwise from the positive x axis[4] It is convenient, though not essential, to interchange the principal magnifications if necessary so that f_1 is the larger, and to quote *its* direction α only; it also helpful to quote the direction after the local rotation has been applied too, i.e. α-θ (the rotation was *clockwise*).

To measure strain itself, we subtract from F the value it has for an undistorted medium, namely unity; the result, E = F - I, i.e. $e_{ij} = f_{ij} - \delta_{ij}$, is called the *strain* tensor. The principal directions of this are identical with those of E; its eigenvalues differ only in having the one subtracted, i.e. $e_{1,2} = f_{1,2} - 1$, and are called the *principal strain components*.

The stretching by f_1 and f_2 in two mutually perpendicular directions can obviously be rephrased in several ways. One such is as the product of an isotropic *magnification* by a factor $\sqrt{f_1 f_2}$ - the same in both directions - and an anisotropic magnification, or *elongation* by a factor $\sqrt{f_1/f_2}$ - magnification in one direction with simultaneous demagnification in the other; the former changes area without changing shape, and the latter changes shape without changing area; both parameters are *linear*, measuring factors by which distances change as a result of the distortion. Other factors that could be used include the *area magnification* (the factor by which areas change) $f_1 f_2$, the *area strain* (the fractional area change) $f_1 f_2 - 1$, and the maximum shear strain $|\frac{1}{2}(f_1 - f_2)|$ [5]. For low strain levels, the elongation approximates the maximum shear strain. [The term *stretch*, which suggested itself first for what is here called elongation, has been pre-empted to mean the ratio of distorted to undistorted lengths for small line elements in the medium. Terminology is however confused in the literature, and both of the terms stretch and elongation are also widely employed in loose senses.]

We chose above to decompose D into a deformation followed by a rotation; it is helpful to note that the opposite choice leads to no significant difference in the later analysis. If D is written as F'R instead, then the fact that F'R = RF means that F' is simply RFR^{-1}, i.e. F with respect to rotated axes, giving the same principal directions as before with respect to the medium itself.

A.3 For completeness, we note that there exist two *invariants* of the tensor F - invariants in the sense that they are unchanged by rotation of the coordinate axes - namely its trace α and its determinant β; by evaluating with axes along the principal directions, we can easily relate these to $f_{1,2}$:

$$\alpha = f_1 + f_2 \qquad \beta = f_1 f_2 \quad \text{(A9)}$$

which thus provides an alternative route to finding the principal magnifications if the orientation of the principal axes is not required.

In the same spirit, we note a route to the principal magnifications that evades the removal of the rotation R entirely. According to (A2), the scalar product between two small vectors $\delta\mathbf{r}_1$ and $\delta\mathbf{r}_2$ in the original medium becomes on distortion

$$\delta r'_{1i}\delta r'_{2i} = d_{ij}\delta r_{1j}d_{ik}\partial r_{2k} = c_{jk}\delta r_{1j}\delta r_{2k} \qquad (A10)$$

if we define $c_{jk} = d_{ij}d_{ik}$, i.e. $C = D^TD$. Any pure rotation preserves the lengths of and angles between such vectors as $\delta\mathbf{r}_1$ and $\delta\mathbf{r}_2$, and so preserves the scalar product; according to (A3) however, that is preserved for arbitrary $\delta\mathbf{r}_1$ and $\delta\mathbf{r}_2$ however only if $c_{ij} = \delta_{ij}$, i.e. if $C = I$. The departure of C from unity thus provides an alternative way of measuring distortions other than rotations, and its eigenvalues prove to be simply f_1^2 and f_2^2. Since $D = RF$, its transpose D^T is F^TR^T, so that $D^TD = F^TR^TRF = F^TF$ (since R is a rotation and its transpose equals its inverse). If we use as a coordinate basis the eigenvectors of F then, F has the form (A6) and F^T is identical, so that C has the form

$$\begin{pmatrix} f_1^2 & 0 \\ 0 & f_2^2 \end{pmatrix}$$

which reveals its eigenvalues at once. Calculation of C and its eigenvalues, via expression equivalent to those of (A7,A8), provide a convenient way of evading the factorisation of the rotation. It is insufficient however for our present purposes, for which the rotation and the orientation information is also required.

A.4 It is useful finally to note the simplifications that occur if displacements are assumed small - as is commonly the case in engineering applications of elasticity theory - since use is made of the resulting expressions in Appendix E. $D = RF$ becomes $I+U = (I+A)(I+S)$ with small antisymmetric A and small symmetric S, or simply $I + A + S$ to first order. A is then the antisymmetric part of U, and S the symmetric part:

$$a_{ij} = \tfrac{1}{2}(u_{ij}-u_{ji}) \qquad s_{ij} = \tfrac{1}{2}(u_{ij}+u_{ji});$$

the local rotation is

$$\theta = \tan^{-1}\tfrac{1}{2}(u_{12}-u_{21}),$$

and the strain components are

$$e_{11} = u_{11};\quad e_{22} = u_{22};\quad e_{12} = \tfrac{1}{2}(u_{12}+u_{21}). \qquad (A11)$$

Appendix B: Determination of Function Gradients

B.1 This appendix gives the tedious details of how the vector gradient of a scalar function can be estimated from a variable number of function samples irregularly positioned around the point of interest. Three successively more sophisticated fitting methods are given; vector fields are addressed by applying any one procedure to each component in turn.

We suppose the function value observed, with noise, at the point at which the derivatives are required - the *target* point -and also at a set of points with position vectors $\mathbf{r}_i$ relative to this. In the simplest approach, which needs a minimum of two neighbouring points only, we convert the observed function values into a set of increments $\Delta v_i = v(\mathbf{r}_i)-v(0)$ from the observed target point value and model these by a linear ramp function:

$$f_i = c_x x_i + c_y y_i. \qquad (B1)$$

The coefficients c_x and c_y are chosen so as to minimise the summed squared deviation

$$s = \sum_i (\Delta v_i - f_i)^2 = \sum_i (\Delta v_i - c_x x_i - c_y y_i)^2 .$$

The minimisation equations are

$$\begin{pmatrix} \frac{\partial s}{\partial c_x} \\ \frac{\partial s}{\partial c_y} \end{pmatrix} = 0 \Rightarrow \begin{pmatrix} \sum_i x_i^2 & \sum_i x_i y_i \\ \sum_i x_i y_i & \sum_i y_i^2 \end{pmatrix} \begin{pmatrix} c_x \\ c_y \end{pmatrix} = \begin{pmatrix} \sum_i x_i \Delta v_i \\ \sum_i y_i \Delta v_i \end{pmatrix} \qquad (B2)$$

which are readily solved analytically. The vector (c_x,c_y) - here and subsequently - is the required gradient.

B.2 While satisfactory if noise levels are low, this first approach in fact gives disproportionate weight to the target point itself, since the fitted function is forced to match the observed value there exactly. This weakness can be eliminated by fitting the function v itself, rather than incremental values, using a general bilinear function:

$$f_i = c + c_x x_i + c_y y_i. \qquad (B3)$$

The coefficients c, c_x and c_y are chosen to minimise

$$s = \sum_i (v_i - f_i)^2 = \sum_i (v_i - c - c_x x_i - c_y y_i)^2 ,$$

which leads to a set of *three* equations

$$\begin{pmatrix} \frac{\partial s}{\partial c} \\ \frac{\partial s}{\partial c_x} \\ \frac{\partial s}{\partial c_y} \end{pmatrix} = 0 \Rightarrow \begin{pmatrix} S_{00} & S_{10} & S_{01} \\ S_{10} & S_{20} & S_{11} \\ S_{01} & S_{11} & S_{02} \end{pmatrix} \begin{pmatrix} c \\ c_x \\ c_y \end{pmatrix} = \begin{pmatrix} V_{00} \\ V_{10} \\ V_{01} \end{pmatrix}, \quad \text{(B4)}$$

where S denotes $\sum_i x_i^m y_i^n$ and V_{mn} denotes $\sum_i v_i x_i^m y_i^n$, which are readily enough solved for c_x and c_y as part of a computer program. If no more than two neighbouring samples are available, this simply gives the same result as (B2).

If four or more neighbouring samples are available, a better result still may be obtained by fitting a bi-quadratic function:

$$f_i = c + c_x x_i + c_y y_i + c_{xx} x^2 + c_{xy} xy + c_{yy} y^2. \quad \text{(B5)}$$

The minimisation equations for this case are

$$\begin{pmatrix} S_{00} & S_{10} & S_{01} & S_{20} & S_{11} & S_{02} \\ S_{10} & S_{20} & S_{11} & S_{30} & S_{21} & S_{12} \\ S_{01} & S_{11} & S_{02} & S_{21} & S_{12} & S_{03} \\ S_{20} & S_{30} & S_{31} & S_{40} & S_{31} & S_{22} \\ S_{11} & S_{21} & S_{12} & S_{31} & S_{22} & S_{13} \\ S_{02} & S_{12} & S_{03} & S_{22} & S_{13} & S_{04} \end{pmatrix} \begin{pmatrix} c \\ c_x \\ c_y \\ c_{xx} \\ c_{xy} \\ c_{yy} \end{pmatrix} = \begin{pmatrix} V_{00} \\ V_{10} \\ V_{01} \\ V_{20} \\ V_{11} \\ V_{02} \end{pmatrix} \quad \text{(B6)}$$

with the same notation as before. This estimate follows local curvature; if the noise levels are high, it may be less desirable because it effects a lower degree of data smoothing.

Appendix C: Obtaining Compensating Re-sampling Grids

This appendix sets out how a re-sampling grid is determined that restores the region around a point with given distortion parameters to its original shape and orientation.

We want to know in effect into what the vectors **a** = (1,0) and **b** = (0,1) in the original crystal are transformed by the local deformation matrix D, since these define the increments at which the undistorted crystal would have been sampled. These are

$$\mathbf{a}' = \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} D_{11} \\ D_{21} \end{pmatrix}$$

and

$$\mathbf{b}' = \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} D_{12} \\ D_{22} \end{pmatrix} \quad \text{(C1)}$$

according to (A1) or (A2). If the matrix D is retained in its original form, no further effort is necessary to recover the corrective sampling grid.

If D has been analysed as in appendix A into parameters θ, f_1, f_2 and α however, it needs to be recovered from these. The second stage of distortion involves independent magnification of the medium by f_1 at polar angle α and by f_2 at 90° to this direction. This is described by a matrix

$$F = \begin{pmatrix} f_1 & 0 \\ 0 & f_2 \end{pmatrix}$$

relative to axes with those orientations, and so by $L_{-\alpha} F L_{-\alpha}^{-1}$, i.e. $L_{-\alpha} F L_{\alpha}$, relative to the original axes, where L_α describes an anticlockwise rotation of axes by α. This operation is preceded by a clockwise rotation of the medium by θ however, described by $L_{-\theta}$, so that $D = L_{-\alpha} F L_\alpha L_{-\theta} = L_{-\alpha} F L_{\alpha-\theta}$, i.e.

$$D = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{pmatrix} \begin{pmatrix} f_1 & 0 \\ 0 & f_2 \end{pmatrix} \begin{pmatrix} \cos(\alpha-\theta) & \sin(\alpha-\theta) \\ -\sin(\alpha-\theta) & \cos(\alpha-\theta) \end{pmatrix} \quad \text{(C2)}$$

Multiplying out the three matrices we thus obtain

$$\mathbf{a}' = \begin{pmatrix} f_1\cos(\alpha)\cos(\alpha-\theta)+f_2\sin(\alpha)\sin(\alpha-\theta) \\ f_1\sin(\alpha)\cos(\alpha-\theta)-f_2\cos(\alpha)\sin(\alpha-\theta) \end{pmatrix}$$

and

$$\mathbf{b}' = \begin{pmatrix} f_1\cos(\alpha)\sin(\alpha-\theta)-f_2\sin(\alpha)\cos(\alpha-\theta) \\ f_1\sin(\alpha)\sin(\alpha-\theta)-f_2\cos(\alpha)\cos(\alpha-\theta) \end{pmatrix}. \quad \text{(C3)}$$

Appendix D: Least-Squares Lattice Parameter Determination

To determine the lattice base vectors **a**, **b**, and origin **c** best fitting a set of observed lattice positions $\mathbf{r_i}$ with known indices (h_i,k_i), we can choose them to minimise the summed squared deviation function

$$s = \sum_i |\mathbf{r_i} - h_i\mathbf{a} - k_i\mathbf{b} - \mathbf{c}|^2 \quad \text{(D1)}$$
$$= \sum_i \{(x_i - h_i a_1 - k_i b_1 - c_1)^2 + (y_i - h_i a_2 - k_i b_2 - c_2)^2\}.$$

It is sufficient to solve the minimisation for one of the two sums, since they involve entirely independent coefficients. The first leads to minimisation equations

$$\begin{pmatrix} \frac{\partial s}{\partial a_1} \\ \frac{\partial s}{\partial b_1} \\ \frac{\partial s}{\partial c_1} \end{pmatrix} = 0 \Rightarrow \begin{pmatrix} S_{hh} & S_{hk} & S_h \\ S_{hk} & S_{kk} & S_k \\ S_h & S_k & S \end{pmatrix} \begin{pmatrix} a_1 \\ b_1 \\ c_1 \end{pmatrix} = \begin{pmatrix} X_h \\ X_k \\ X \end{pmatrix}, \quad \text{(D2)}$$

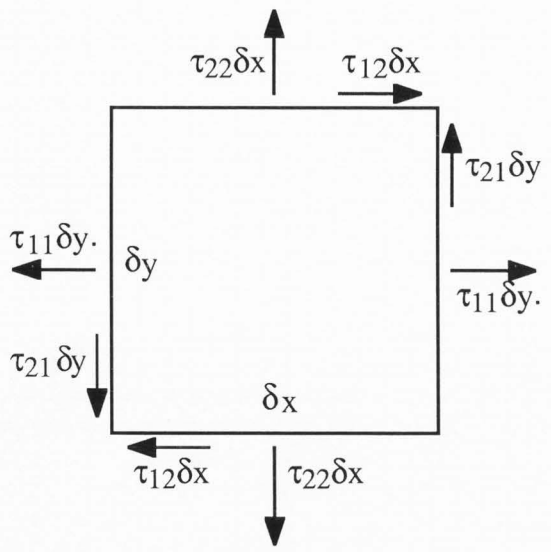

Fig.E1 Forces acting on the edges of an infinitesimal element δx by δy of a 2-D elastic medium.

where $S_{hh}=\sum_i h_i^2$, $S_{hk}=\sum_i h_i k_i$, $S_{hk}=\sum_i k_i^2$, $S_h=\sum_i h_i$, $S_k=\sum_i k_i$, $S=\sum_i 1$, $X_h=\sum_i x_i h_i$, $X_k=\sum_i x_i k_i$ and $X=\sum_i x_i$, which are easily solved as part of a computer program.

Note that indices for all lattice positions are needed before the minimisation can be carried out correctly. These are commonly obtained by requiring an initial estimate of the lattice vectors, and calculating approximate indices for each position from these:

$$\begin{pmatrix} h_i \\ k_i \end{pmatrix} = \frac{1}{a_1b_2-a_2b_1} \begin{pmatrix} b_2 & -b_1 \\ -a_2 & a_1 \end{pmatrix} \begin{pmatrix} x_i-c_1 \\ y_i-c_2 \end{pmatrix}; \qquad \text{(D3)}$$

these can be rounded to the nearest integer to provide the required indices, and any positions deviating markedly from integral indices can be detected and discarded if desired. A bootstrap approach can be used to the fitting of large area, with the area fitted initially being small enough to prevent misindexing even with very rough estimates of the lattice parameters, and the result being sufficiently accurate to index a larger area correctly.

Appendix E: The Elastic Behaviour of 2-D Crystals

E.1 Appendix A developed the strain tensor e_{ij} which measures deformation as opposed to rotation of the medium; we have now to introduce the *stress* tensor τ_{ij} which measures the internal forces acting on small regions, and which is related to the strain tensor by a set of elastic moduli.

We generalise the idea of pressure or stress as force per unit length in the medium by writing the force $\delta\mathbf{f}$ acting on the medium behind a small line element in the 2-D medium in the form

$$\delta f_i = \tau_{ij}\, \delta s_j, \qquad \text{(E1)}$$

where the vector $\delta\mathbf{s}$ is defined to lie perpendicular to the line element, with a modulus equal to its length. The tensor ratio allows $\delta\mathbf{f}$ to have different direction from $\delta\mathbf{s}$, thus accommodating shear as well as normal stresses; τ_{ij} is called the *stress* tensor. That τ_{ij} is symmetrical is easily seen by considering the net torque acting on a square element of the medium with side δa, which can be seen from Fig.E1 to be $(\tau_{12}-\tau_{21})\delta a^2$; since the moment of inertia is proportional to δa^4 however, the element will have an infinite angular acceleration in the limit $\delta a \to 0$ unless $\tau_{12} = \tau_{21}$. τ_{ij} thus has the general form

$$\begin{pmatrix} \tau_{11} & \tau_{12} \\ \tau_{12} & \tau_{22} \end{pmatrix}.$$

The diagonal elements are again called *normal* or *tensile* components, and the off-diagonal elements *shear* components. Since the tensor is symmetric, it will appear diagonal (without shear components) if the coordinate axes are suitably oriented - parallel to its eigenvectors. It is not at this stage obvious however whether these eigenvectors coincide with those of the strain tensor at the same point of the medium.

E.2 We can expect that for moderate levels of distortion at least, the crystal will obey Hooke's law, giving rise to a linear relationship between the stress and strain tensors:

$$\tau_{ij} = c_{ikjl}\, e_{kl}, \qquad \text{(E2)}$$

in which the stiffness coefficients c_{ikjl} form a fourth rank tensor with 16 components. In fact, only 6 of these components are independent in general, and for an isotropic crystal only 2; these assertions are now proved.

We simplify first by rewriting the equations in non-tensor (*Voigt*) form; since τ_{ij} and e_{ij} are symmetric, we write them in the form

$$T = \begin{pmatrix} \tau_1 & \tau_3 \\ \tau_3 & \tau_2 \end{pmatrix} \qquad E = \begin{pmatrix} e_1 & e_3 \\ e_3 & e_2 \end{pmatrix} \qquad \text{(E3)}$$

(in which the use of a single subscript does *not* now denote an eigenvalue,as it did in Appendix A), and so rewrite (E2) in the form

$$\begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix}. \qquad \text{(E4)}$$

with only nine independent coefficients. We next consider the work $\delta\Gamma$ done on a small element $(\delta x,\delta y)$ when a small additional displacement $\delta\mathbf{u}$ is effected. Considering Fig.E1 again, this can be seen to be

$$\begin{aligned}\delta\Gamma &= \tau_{11}\delta y\,\delta(\frac{\partial v}{\partial x}\delta x) + \tau_{12}\delta x\,\delta(\frac{\partial v}{\partial y}\delta y)\\ &\quad + \tau_{21}\delta y\,\delta(\frac{\partial w}{\delta x}\delta x) + \tau_{22}\delta x\delta(\frac{\partial w}{\partial y}\delta y)\\ &= (\tau_{11}\delta u_{11} + \tau_{12}\delta u_{12}\\ &\quad + \tau_{21}\delta u_{21} + \tau_{22}\delta u_{22})\delta x\delta y.\end{aligned}$$

For the purpose of determining independence of elasticity coefficients, we can consider the special case of small displacements (§A.4 above) without loss of generality; (A11) allows us to rewrite $\delta\Gamma$ in the form

$$\delta\Gamma = (\tau_{11}\delta e_{11} + 2\tau_{12}\delta e_{12} + \tau_{22}\delta e_{22})\delta x\delta y;$$

the work done per unit area $\delta\gamma = \delta\Gamma/(\delta x\delta y)$ can thus be written as

$$\delta\gamma = \tau_1\delta e_1 + \tau_2\delta e_2 + 2\tau_3\delta e_3. \qquad \text{(E5)}$$

It follows that

$$\tau_1 = \frac{\partial\gamma}{\partial e_1} \quad \text{and} \quad \tau_3 = \frac{1}{2}\frac{\partial\gamma}{\partial e_3},$$

and we can then write

$$c_{13} = \frac{\partial\tau_1}{\partial e_3} = \frac{\partial^2\varepsilon}{\partial e_1\partial e_3} = 2\frac{\partial\tau_3}{\partial e_1} = 2c_{31}.$$

In the same way, $c_{23} = 2c_{32}$ and $c_{12} = c_{21}$, so that the general form of (E4) is in fact no worse than

$$\begin{pmatrix}\tau_1\\ \tau_2\\ \tau_3\end{pmatrix} = \begin{pmatrix}c_{11} & c_{21} & 2c_{31}\\ c_{21} & c_{22} & 2c_{32}\\ c_{31} & c_{32} & c_{33}\end{pmatrix}\begin{pmatrix}e_1\\ e_2\\ e_3\end{pmatrix}. \qquad \text{(E6)}$$

with only six independent coefficients.

E.3 We consider now the special case of a p4 crystal and show that the number of independent components in the stiffness matrix is only four [6]. The crystal symmetry implies that all components of the fourth rank tensor c_{ikjl} must be unaltered when the coordinate axes are rotated by 90°. A rotation of 90° anticlockwise is described by the matrix

$$L = \begin{pmatrix}0 & 1\\ -1 & 0\end{pmatrix}$$

and the stiffness tensor transforms according to

$$c'_{ikjl} = l_{ip}l_{kr}l_{jq}l_{ls}\,c_{prqs}. \qquad \text{(E7)}$$

To make use of this, we need to note how the coefficients c_{ij} of (E6) are related to the tensor components: from (E2)

$$\begin{aligned}\tau_1 &= \tau_{11}\\ &= c_{1111}e_{11} + c_{1112}e_{12} + c_{1211}e_{21} + c_{1212}e_{22}\\ &= c_{1111}e_1 + (c_{1112} + c_{1211})e_3 + c_{1212}e_2\end{aligned}$$

so that comparison with (B6) gives

$$c_{11} = c_{1111};\quad c_{13} = c_{1112} + c_{1211};\quad c_{12} = c_{1212}.$$

Similarly, we find that

$$\begin{aligned}&c_{31} = c_{1121} = c_{2121};\\ &c_{32} = c_{1222} = c_{2212};\\ &c_{33} = c_{1122} + c_{1221} = c_{2112} + c_{2211};\\ &c_{21} = c_{2121}\\ &c_{23} = c_{2122} + c_{2221}\\ &c_{22} = c_{2222}.\end{aligned} \qquad \text{(E8)}$$

On evaluating the transform (E7) [7] and using (E8), we find that

$$\begin{aligned}&c'_{1111} = c_{2222} = c_{1111} && \Rightarrow\ c_{11} = c_{22};\\ &\left.\begin{aligned}c'_{1112} &= -\,c_{2221} = c_{1112}\\ c'_{1211} &= -\,c_{2122} = c_{1211}\end{aligned}\right\} && \Rightarrow\ c_{13} = -\,c_{23};\\ &c'_{1121} = -\,c_{2212} = c_{1121} && \Rightarrow\ c_{31} = -\,c_{32}.\end{aligned} \qquad \text{(E9)}$$

The remaining components lead to relations that are either trivial or already established. The elasticity equations thus have the form

$$\begin{pmatrix}\tau_1\\ \tau_2\\ \tau_3\end{pmatrix} = \begin{pmatrix}c_{11} & c_{21} & 2c_{31}\\ c_{21} & c_{11} & -2c_{31}\\ c_{31} & -c_{31} & c_{33}\end{pmatrix}\begin{pmatrix}e_1\\ e_2\\ e_3\end{pmatrix}, \qquad \text{(E10)}$$

which has only four independent components.

E.4 Trigonal and hexagonal crystals are obviously also special cases of practical interest, but have more rather than fewer independent elastic constants - the transformation (E7) is much more tedious to evaluate, and is not pursued further here. What is treated next here is the case of an *isotropic* sheet, whose elastic properties are assumed to be direction-independent. The form of the stiffness matrix can be conveniently deduced in this case from the initial

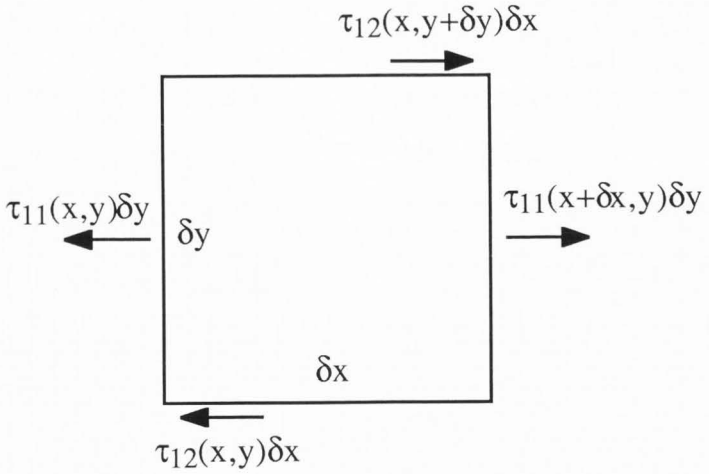

Fig.F1 Forces acting on the edges of an infinitesimal element δx by δy of a 2-D elastic medium, when the stress varies across the element. For clarity, only the x-components are shown.

result that the most general isotropic fourth rank tensor is

$$c_{ikjl} = \lambda\delta_{ik}\delta_{jl} + \mu\delta_{ij}\delta_{kl} + \nu\delta_{il}\delta kj \qquad (E11)$$

[e.g. Jeffreys & Jeffreys, 1956]. Using (E8) we can thus write

$$c_{11} = c_{1111} = \lambda + \mu + \nu$$

and evaluating the other coefficients in the same way, we find that the elasticity equations take the simple form

$$\begin{pmatrix}\tau_1\\ \tau_2\\ \tau_3\end{pmatrix} = \begin{pmatrix} c_{11} & c_{21} & 0 \\ c_{21} & c_{11} & 0 \\ 0 & 0 & c_{11}-c_{12}\end{pmatrix}\begin{pmatrix}e_1\\ e_2\\ e_3\end{pmatrix}, \qquad (E12)$$

having only three two independent coefficients $c_{11} = \lambda + \mu + \nu$ and $c_{21} = \mu$, so that the behaviour of the sheet is closely analogous to that of a three-dimensional istropic medium.

Appendix F: External Stress on a 2-D Medium

F.1 One respect in which a 2-D sheet differs fundamentally from a 3-D medium is in the possible application of an external force at one or both surfaces of the sheet. The requirement for the sheet to be in equilibrium means that we can deduce what surface forces are present from the imbalance in stresses at opposite edges of an elementary area δx by δy. Considering Fig.F1, we can see that the internal stresses within the medium exert the following net force **F** on the element

$$\begin{pmatrix}F_1\\ F_2\end{pmatrix} = \begin{pmatrix} \dfrac{\partial\tau_{11}}{\partial x}\delta x\delta y + \dfrac{\partial\tau_{12}}{\delta y}\delta y\delta x \\ \dfrac{\partial\tau_{21}}{\partial x}\delta x\delta y + \dfrac{\partial\tau_{22}}{\delta y}\delta y\delta x \end{pmatrix}$$

The force per unit area, or stress, that must be being applied at the surface(s) to counterbalance this is accordingly **f** with components

$$\begin{pmatrix}f_1\\ f_2\end{pmatrix} = -\begin{pmatrix} \dfrac{\partial\tau_1}{\partial x} + \dfrac{\partial\tau_3}{\partial y} \\ \dfrac{\partial\tau_3}{\partial x} + \dfrac{\partial\tau_2}{\partial y} \end{pmatrix} \qquad (F1)$$

The relations (E6), (E10) or (E12) might be used now to provide an expression for this external stress in terms of the strain tensor e_{ij}. We shall adopt the simplest model, with the fewest elastic constants, since we do not know their actual values, namely (E12); before we use it however, we need a reasonable basis for assigning relative sizes to c_{11} and c_{12}.

F.2 If we invert the relations (E12), we obtain

$$\begin{pmatrix}e_1\\ e_2\\ e_3\end{pmatrix} = \frac{1}{(c_{11}^2-c_{12}^2)}\begin{pmatrix} c_{11} & -c_{12} & 0 \\ -c_{12} & c_{11} & 0 \\ 0 & 0 & c_{11}+c_{12}\end{pmatrix}\begin{pmatrix}\tau_1\\ \tau_2\\ \tau_3\end{pmatrix} \qquad (F2)$$

By comparison with the form

$$E\begin{pmatrix}e_1\\e_2\\e_3\end{pmatrix}=\begin{pmatrix}1 & -\sigma & 0\\ -\sigma & 1 & 0\\ 0 & 0 & G\end{pmatrix}\begin{pmatrix}\tau_1\\\tau_2\\\tau_3\end{pmatrix},$$

in which E is the 2-D analogue of Young's modulus (τ_1/e_1 when $\tau_2 = 0$), σ that of the Poisson ratio ($-e_2/e_1$ when $\tau_2 = 0$), and G that of the shear modulus (τ_3/e_3), we can conclude that $c_{12} = \sigma c_{11}$. In the 3-D case, the Poisson ratio cannot exceed $\frac{1}{2}$ without resulting in a negative bulk modulus; in 2-D however, all moduli remain positive provided $-1<\sigma<1$, and we may take a typical value to be perhaps $\frac{1}{3}$. We therefore adopt the following final model for the stress/strain relationship in a 2-D crystal, with only one unknown parameter:

$$\begin{pmatrix}\tau_1\\\tau_2\\\tau_3\end{pmatrix}=c_{11}\begin{pmatrix}1 & \frac{1}{3} & 0\\ \frac{1}{3} & 1 & 0\\ 0 & 0 & \frac{2}{3}\end{pmatrix}\begin{pmatrix}e_1\\e_2\\e_3\end{pmatrix}. \qquad \text{(F3)}$$

Using this, we can evaluate (C1), obtaining

$$\begin{pmatrix}f_1\\f_2\end{pmatrix}=-c_{11}\begin{pmatrix}\frac{\partial e_1}{\partial x}+\frac{1}{3}\frac{\partial e_2}{\partial x}+\frac{2}{3}\frac{\partial e_3}{\partial y}\\ \frac{1}{3}\frac{\partial e_1}{\partial y}+\frac{\partial e_2}{\partial y}+\frac{2}{3}\frac{\partial e_3}{\partial x}\end{pmatrix}. \qquad \text{(F4)}$$

The strain components $e_1=e_{11}$, $e_2=e_{22}$, $e_3=e_{12}=e_{21}$ are recovered from stored strain parameters simply by applying a clockwise rotation by $\alpha'=\alpha-\theta$ to the strain tensor in diagonal form, i.e.

$$E=\begin{pmatrix}\cos(\alpha') & -\sin(\alpha')\\ \sin(\alpha') & \cos(\alpha')\end{pmatrix}\begin{pmatrix}f_1 & 0\\ 0 & f_2\end{pmatrix}\begin{pmatrix}\cos(\alpha') & \sin(\alpha')\\ -\sin(\alpha') & \cos(\alpha')\end{pmatrix}$$

Appendix G: New Semper Programs and Commands

Beginning at the highest level, several Semper *programs* (i.e. procedures defined in terms of existing Semper commands only) have been provided to support the procedures described in this paper, of which the important ones are the following.

INDPL, INDPLADD, INPLDOUT together effect the indexation of a position list describing lattice sites in a heavily distorted crystal, producing both *index-ordered* and *indexed* position lists from a raw list.

INDPLMARK, given an index-ordered position list, marks the distorted lattice on a display in the form of a mesh of lines joining lattice sites.

INDPLTEST, given an index-ordered position list, tests the validity of the indexation by seeking and reporting any instances of supposedly neighbouring sites in inappropriate relative positions.

UNBEND, given an analysed position list, produces a distortion-compensated average on the basis of constant distortion parameters across the regions averaged.

BILUNBEND, given an analysed position list, produces a distortion-compensated average with distortion parameters varying bilinearly across the regions averaged.

POSNVER verifies all parameters recorded for a given position.

Fortran modules have been written to provide the following new Semper *commands*.

PLANALYSE generates 14-layer *analysed* position lists from 4/5-layer *indexed* lists produced the program INDPL, recording local rotation / deformation parameters for each position.

PLDELETE deletes positions from lists in various alternative ways: singly; inside or outside a given circle, rectangle or polygon; or if the rotation/deformation parameters meet arbitrary conditions specified algebraically. For example,

```
XWIRES; PLDELETE @XY
XWIRES CLOSED CURVE: PLDELETE 1 WITH 999
PLDELETE IF ELONG > 1.02
```

PLFIND searches a position list and returns the Semper coordinates of the position nearest a requested point.

PLMARK marks position lists on the display in various ways: positions; original positions; displacement vectors; rotations; magnifications and elongation patterns; surface stress vectors; serial numbers; and positions with mark sizes varied according to the position weights - all for an optionally restricted range of position numbers and with optional exaggeration of the strain parameters. For example

```
PLMARK DVECTORS TIMES 2
```

PLSORT sorts position lists by any indicated layer, in ascending or descending order; it can also sort so as to order an externally supplied array, which allows other parameters besides those already stored to be calculated and used as the basis for sorting.

```
PLSORT @MAGN DESCENDING
```

PLUVXY given an analysed position list recovers distortion parameters for a single site indicated by number or approximate position, and sets extraction sampling variables (U,V,X,Y) so that EXTRACT @UVXY recovers locally unbent regions

for averaging; calls PLFIND if the site is indicated by position rather than by number.

PLXTRA adds a further two parameters to those generated by PLANALYSE, namely the x- and y-components of the tangential force exerted at the crystal surface(s) per unit area - estimated via (F4) with c_{11} simply taken to be 1. The output contains 16 layers.

Finally, the following Fortran modules provide lower level facilities for the commands described above:

FITUIJ expects data in the form of a set of displacment differences $\Delta u,\Delta v$ for increments $\Delta x,\Delta y$ from an initial position at which the module estimates the displacement derivatives by a least squares fit; called by PLANALYSE.

PSC2D analyses displacement tensor in terms of local rotation, magnification, elongation (and principle strain components); called by PLANALYSE.

PLCOND condenses position lists, omitting any marked for deletion (i.e. with recorded x-coordinates greater than 10^5); used by PLANALYSE, PLDELETE.

INPLYG given a closed curve definition (the x- and y-coordinates of a series of vertices), determines whether a given position lies inside it or outside; called by PLDELETE. The basic method used to scan all boundary segments counting intersections with a horizontal line from the far left to the point under scrutiny.

References

E W Billington & A Tate, 1981. The physics of deformation and flow. (McGraw-Hill)

A Crowther & U Sleytr, 1977. An analysis of the fine structure of the surface layers from two strains of *Clostridia*, including correction for distorted images. *J. Ultrastruct. Res.* 58:41-49.

R Dürr, 1991. Displacement field analysis: calculation of distortion measures from displacement maps. *Ultramicroscopy* **38**:135-141

R Dürr, E Peterhans, & R von der Heydt. 1989. Correction of distorted image pairs with elastic models. *Eur J. Cell Biol* 48 Suppl 25:85-88.

J Frank, 1982. New methods for averaging non-periodic objects and distorted crystals in biological electron microscopy. *Optik* 63:67-89.

R Hegerl & W Baumeister, 1988. Correlation averaging of a badly distorted lattice: the surface protein of *Pyrodictium occultum*. *J. Electron Microsc. Technique* 9:413-419.

R Henderson, J M Baldwin, K H Downing, J Lepault & F Zemlin, 1986. Structure of purple membrane from *Halobacterium halobium*: recording, measurement and evaluation of micrographs at 3.5Å resolution. *Ultramicroscopy* 19:147-178.

H Jeffreys & B S Jeffreys, 1956. Mathematical Physics. Cambridge University Press, §3.03

B M Phipps, H Engelhardt, R Huber & W Baumeister, 1990. Three-dimensional structure of the crystalline protein envelope layer of the hyperthermophilic archaebacterium *Pyrobaculum islandicum*. *J. Struct. Biol.* 103:152-163.

W O Saxton & W. Baumeister, 1982. The correlation averaging of a regularly arranged bacterial cell envelope protein. *J. Microsc* 127:127-138.

W O Saxton, R Dürr & W Baumeister, 1991. From lattice distortion to molecular distortion: characterising and exploiting crystal deformation. *Ultramicroscopy* 46:287-306.

W O Saxton, T J Pitt & M Horner, 1979. Digital image processing: the Semper system. *Ultramicroscopy* 4:343-354.

L A Segel & G H Handelman, 1977. Mathematics applied to continuum mechanics. (Macmillan)

A J M Spencer, 1980. Continuum mechanics. (Longman)

Notes

[1] The rotation is in fact reported in an anticlockwise sense, in contrast to that assumed in the appendix A.

[2] This assertion in fact holds good only if the specimen is untilted, of course.

[3 Rotating by 180° obviously reverses (negates) all components of D, so the ambiguity of $\pm\pi$ in θ left by (A4) is resolved below by requiring s_1 to be positive.

[4] The ambiguity of π this time is fundamental but trivial.

[5] The shear strain obviously depends on the orientation of the coordinate axes, being zero when they lie along the principal directions; its maximum value is found by considering f'_{12} following a rotation and maximising w.r.t. the rotation angle; this proves to be $\sqrt{(\frac{1}{4}(f_{11}-f_{22})^2+f_{12}^2)}$ for arbitrary axes, and therefore $\frac{1}{2}|(f_1-f_2)|$ on evaluation with respect to the principal axes.

[6] The case of a cubic crystal in three dimensions, which has only three independent components, seems equivalent at first, but has in fact more symmetry elements, and in particular rotation axes in the crystal plane as well as normal to it.

[7] The evaluation is not in fact very tedious, since the vanishing of l_{ij} whenever i=j means that the only non-zero term in the sum is that in which all subscripts have changed.