27-731 Texture & Anisotropy
(mini version of 27-750 Advanced Characterization and Microstructural Analysis),
A.D. Rollett, J.V. Gordon

Homework 2; **different representations of orientations (as rotations), effect of crystal and sample symmetry**. Solution Set as of 18th Jan '20, corrected 29 Jan '20

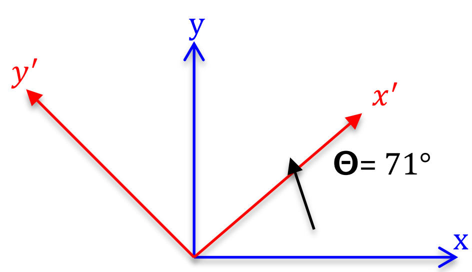
Points: 110

Due: 11:59 pm, Sat., Jan. 25th.

You are expected to do the numerical exercises in Python/Matlab. For each answer, provide your Python/Matlab script along with the values and/or graphs. As usual, there is no problem about working together but you have to provide individual answers (i.e., there had better not be any identical Python/Matlab scripts!).

1. [45] In this exercise, we learn how to apply an axis transformation.

*a*. Draw a pair of orthogonal x-y axes (hint: the arrow.m package for Python/Matlab is helpful, which is posted on Canvas), with x pointing to the right, and y up the page (i.e., the standard arrangement);
*b*. Draw a second pair of x'-y' axes that are rotated with respect to the first pair by 71° (hint: a positive angle corresponds to an anti-clockwise rotation, looking down the rotation axis, which is equivalent to the right-hand rule);




*c*. Calculate the coefficients of the transformation matrix;

$$a_{ij} = \begin{bmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.819 & 0.574 & 0 \\ -0.574 & 0.819 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

*d*. Compute the coefficients of the unit vector parallel to [410] in the first frame, assuming that x and y are parallel to [100] and [010], respectively;

$v_i$ = [410] , |v|=$\sqrt{4*4+1*1+0}$ = $\sqrt{17}$ = 4.123

$\hat{v}_1 = \frac{4}{4.123} = 0.970$ $\hat{v}_2 = \frac{1}{4.123} = 0.243$ $\hat{v}_3 = \frac{0}{4.123} = 0$

*e*. Compute the coefficients of this same vector in the new, primed frame;

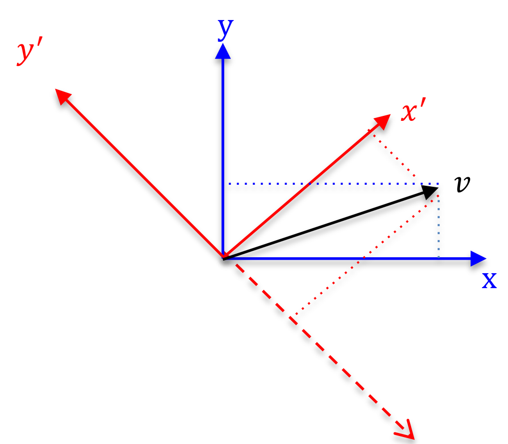
$$\acute{v}_i = a_{ij}v_j$$

$\acute{v}_1 = a_{11}v_1 + a_{12}v_2 + a_{13}v_3$ = 0.82*0.970+ 0.57*0.243+ 0*0= 0.934

$\acute{v}_2 = a_{21}v_1 + a_{22}v_2 + a_{23}v_3$ = -0.57*0.970+ 0.82*0.243+ 0*0= -0.357

$\acute{v}_3 = a_{31}v_1 + a_{32}v_2 + a_{33}v_3$ = 0*0.970+ 0*0.243+ 0*1=0

*f*. Draw the vector in the same diagram;




*g*. A stress tensor cannot be drawn as easily as a vector. Therefore this part asks you to compute the coefficients in the new frame from the following stress tensor.

$$\sigma_{ij} = \begin{bmatrix} 10*\sqrt{2} & 5*\sqrt{2} \\ 5*\sqrt{2} & 15*\sqrt{2} \end{bmatrix}$$

$$\acute{\sigma}_{ij} = a_{ik}a_{jl}\sigma_{kl}$$

$\acute{\sigma}_{11} = a_{1k}a_{1l}\sigma_{kl} = a_{11}a_{11}\sigma_{11} + a_{11}a_{12}\sigma_{12} + a_{12}a_{11}\sigma_{21} + a_{12}a_{12}\sigma_{22}$= (0.82*0.82*10*$\sqrt{2}$ )+(0.82*0.57*5*$\sqrt{2}$)+(0.57*0.82*5*$\sqrt{2}$ )+(0.57*0.57*15*$\sqrt{2}$ )=23.01

$\acute{\sigma}_{12} = a_{1k}a_{2l}\sigma_{kl} = a_{11}a_{21}\sigma_{11} + a_{11}a_{22}\sigma_{12} + a_{12}a_{21}\sigma_{21} + a_{12}a_{22}\sigma_{22}$ = (0.82*-0.57*10*$\sqrt{2}$ )+(0.82*0.82*5*$\sqrt{2}$ )+(0.57*-0.57*5*$\sqrt{2}$ )+(0.57*0.82*15*$\sqrt{2}$ )=5.76

$\acute{\sigma}_{21} = a_{2k}a_{1l}\sigma_{kl} = a_{21}a_{11}\sigma_{11} + a_{21}a_{12}\sigma_{12} + a_{22}a_{11}\sigma_{21} + a_{22}a_{12}\sigma_{22}$ = (-0.57*0.82*10*$\sqrt{2}$ )+(-0.57*0.57*5*$\sqrt{2}$)+(0.82*0.82*5*$\sqrt{2}$)+(0.82*0.57*15*$\sqrt{2}$)= 5.76

$\acute{\sigma}_{22} = a_{2k}a_{2l}\sigma_{ll} = a_{21}a_{21}\sigma_{11} + a_{21}a_{22}\sigma_{12} + a_{22}a_{21}\sigma_{21} + a_{22}a_{22}\sigma_{22}$ = (-0.57*-0.57*10*$\sqrt{2}$)+(-0.57*0.82*5*$\sqrt{2}$)+(0.82*-0.57*5*$\sqrt{2}$)+(0.82*0.82*15*$\sqrt{2}$)=12.248

$$\acute{\sigma}_{ij} = \begin{bmatrix} 23.01 & 5.76 \\ 5.76 & 12.248 \end{bmatrix}$$

Python/Matlab Hint: in a python/Matlab script, the commands "w2 = figure(1);" and "print(w2,'-dpng', fileOut);", where fileOut is the filename of the graph to be saved, and the plotting commands are between these two lines, are helpful.
Another Python/Matlab hint: this set of lines can be helpful for saving plots in your current directory.

```
"path = mfilename('fullpath');
[folder, name, ext] = fileparts(path);
fprintf(' current directory =  %s  \n',folder);
fileName = input('Enter a name for the Output file (.png will be added): ','s');
fileOut = strcat(folder,'/',fileName,'.png');"
```

*h*. Relate what you just did to the concept of Mohr's Circle (which you can look up in Wikipedia).

Mohr's Circle is simply a graphical method for accomplishing the same transformation as was performed numerically above.

i. Explain the equivalence of the index notation that we use for the transformation of a 2nd rank tensor with this version:

$$\sigma' = O\sigma O^T$$

This equation is the vector-matrix version of the axis transformation and is convenient to write because it does not depend on the reference frame.
\sigma' = O \sigma O^T

2. [20] In this exercise, we see how to perform a vector (active) rotation.

*a*. Draw the same x-y frame;
*b*. Draw the same unit vector parallel to [410] as above;
*c*. Write out the transpose of the transformation matrix that you obtained above, which is the matrix that represents the corresponding rotation.

$$b_{ij} = a_{ij}^T = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} = \begin{bmatrix} 0.82 & -0.57 \\ 0.57 & 0.82 \end{bmatrix}$$

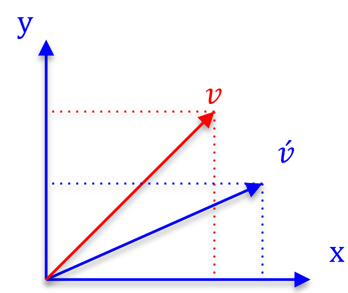
*d*. Compute and draw the new rotated vector.

$$\acute{v}_i = b_{ij} v_j$$

$\acute{v}_1 = b_{11}v_1 + b_{12}v_2 + b_{13}v_3$ = 0.82*0.970+ -0.57*0.243+ 0*0= 0.656

$\acute{v}_2 = b_{21}v_1 + b_{22}v_2 + b_{23}v_3$ = 0.57*0.970+ 0.82*0.243+ 0*0= 0.755

$\acute{v}_3 = b_{31}v_1 + b_{32}v_2 + b_{33}v_3$ = 0*0.943+ 0*0.236+ 0*1=0




3. [30 points]
The exact location of the "S" texture component in Euler space is somewhat debated in the literature, so we will (somewhat arbitrarily) assign it Euler angle values of (32°, 58°, 18°).

*a*. Convert the Euler angles to a matrix.

Here we can use SteveSintay_Pole_Figures etc., although the `variants_ADR.m` script also provides it.

| Rotation Matrix, g (Passive Axis transformation) | | |
|---|---|---|
| 0.719765238 | 0.642854487 | 0.262061274 |
| -0.529131664 | 0.263647909 | 0.806541668 |

| 0.449397023 | -0.719185573 | 0.529919264 |
|---|---|---|

*b*. Apply crystal symmetry and generate the 24 equivalent points. Convert each result from a matrix to Euler angles and list the result. Try to make as compact a table of values as you can (i.e. include 4 columns with the 1st being the number of the symmetry operator and the other 3 columns containing the 3 Euler angles).

Used variants_ADR.m (in Python/Matlab)

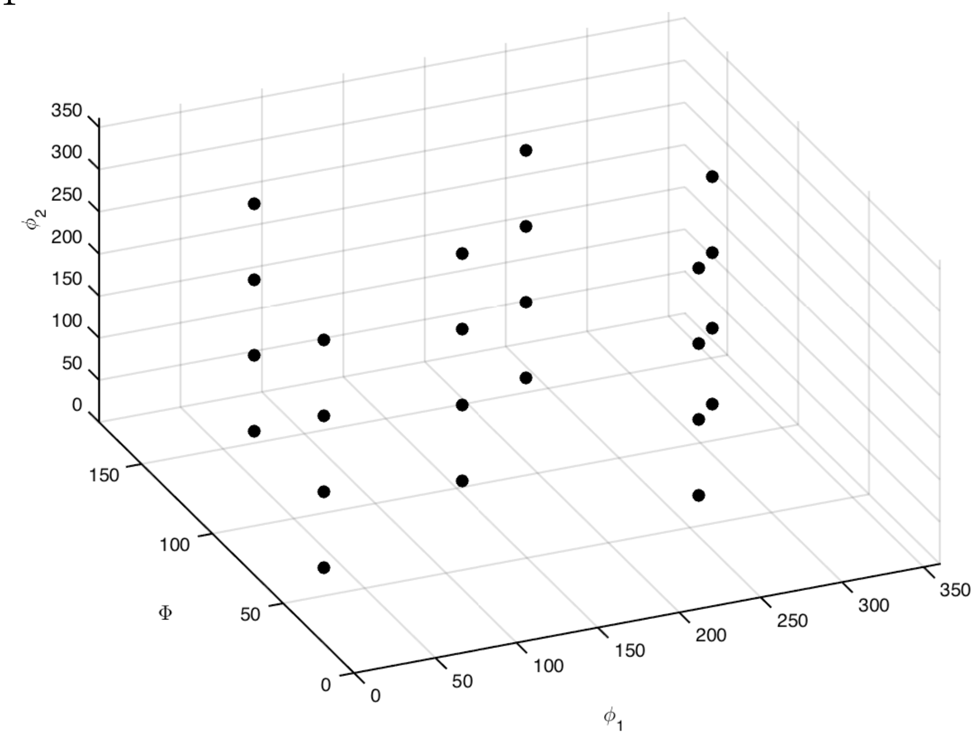
| loop_number | phi1 | PHI | phi2 (degrees) |
|---|---|---|---|
| 1 | 32.0000 | 58.0000 | 18.0000 |
| 2 | 212.0000 | 122.0000 | 162.0000 |
| 3 | 243.5145 | 36.2406 | 153.6862 |
| 4 | 63.5145 | 143.7594 | 26.3138 |
| 5 | 212.0000 | 122.0000 | 342.0000 |
| 6 | 32.0000 | 58.0000 | 198.0000 |
| 7 | 63.5145 | 143.7594 | 206.3138 |
| 8 | 243.5145 | 36.2406 | 333.6862 |
| 9 | 32.0000 | 58.0000 | 108.0000 |
| 10 | 311.7695 | 105.1924 | 123.3060 |
| 11 | 212.0000 | 122.0000 | 72.0000 |
| 12 | 131.7695 | 74.8076 | 56.6940 |
| 13 | 32.0000 | 58.0000 | 288.0000 |
| 14 | 131.7695 | 74.8076 | 236.6940 |
| 15 | 212.0000 | 122.0000 | 252.0000 |
| 16 | 311.7695 | 105.1924 | 303.3060 |
| 17 | 311.7695 | 105.1924 | 33.3060 |
| 18 | 243.5145 | 36.2406 | 63.6862 |
| 19 | 131.7695 | 74.8076 | 146.6940 |
| 20 | 63.5145 | 143.7594 | 116.3138 |
| 21 | 131.7695 | 74.8076 | 326.6940 |
| 22 | 243.5145 | 36.2406 | 243.6862 |
| 23 | 311.7695 | 105.1924 | 213.3060 |
| 24 | 63.5145 | 143.7594 | 296.3138 |

*c*. Plot the points as a 3D plot (Euler angle space) and show 2 or 3 different views. It is good practice to control the axes so that the 1st and 3rd angles range from 0 to 360° and the 2nd 0 to 180°. The following Python/Matlab commands worked for me, where {x,y,z} are vectors of equal length containing the Euler angle values:

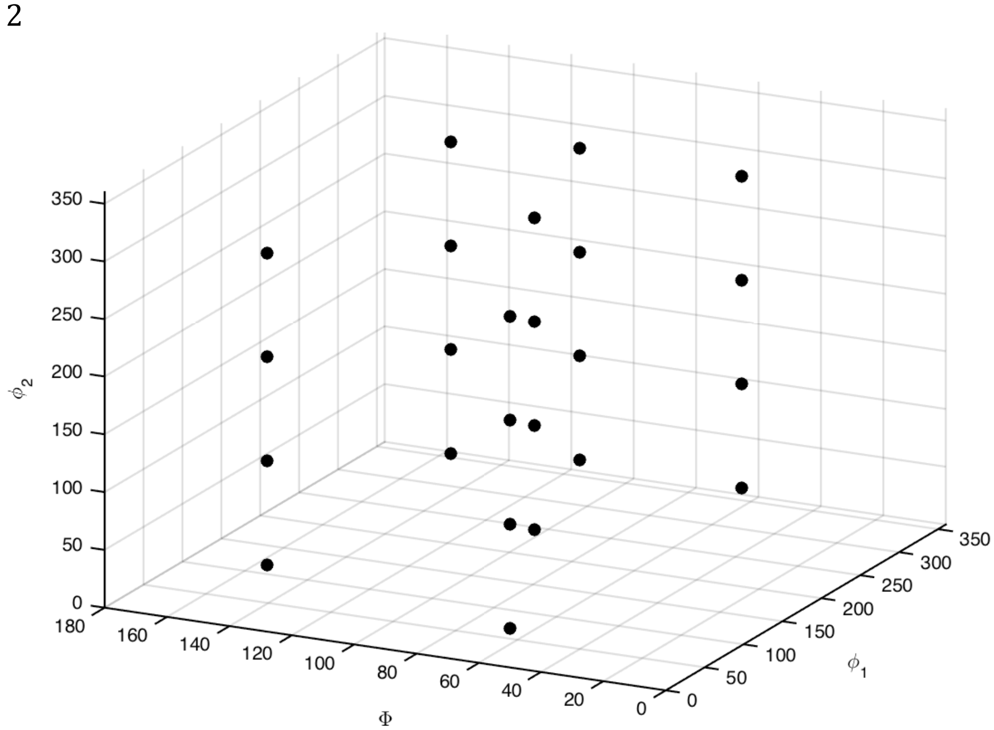
```
scatter3(x,y,z,'MarkerFaceColor','r');
axis([0 360 0 180 0 360]);
xlabel('\phi_1');
ylabel('\Phi');
zlabel('\phi_2');
```

The backslash ("\") in front of e.g. "Phi" tells Python/Matlab to treat the following letters as a symbol.

View 1




View 2



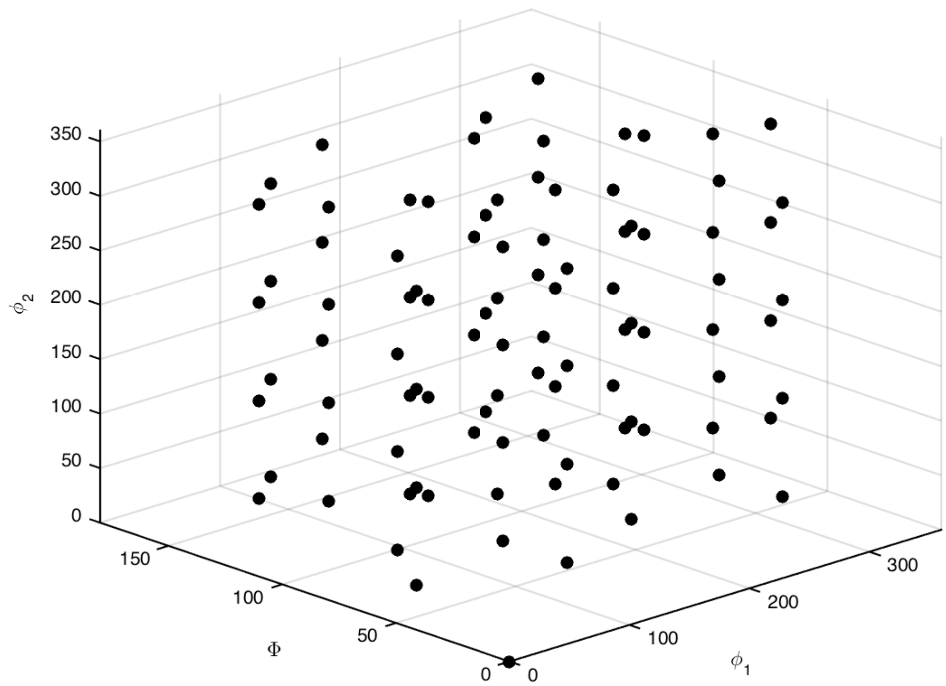
*d*. Now apply orthorhombic sample symmetry (222 point group) in addition to cubic crystal symmetry (432 point group) and re-draw. Make two views, one with the full range of Euler angles, and a second one with the range limited to 0-90° for all 3 angles.

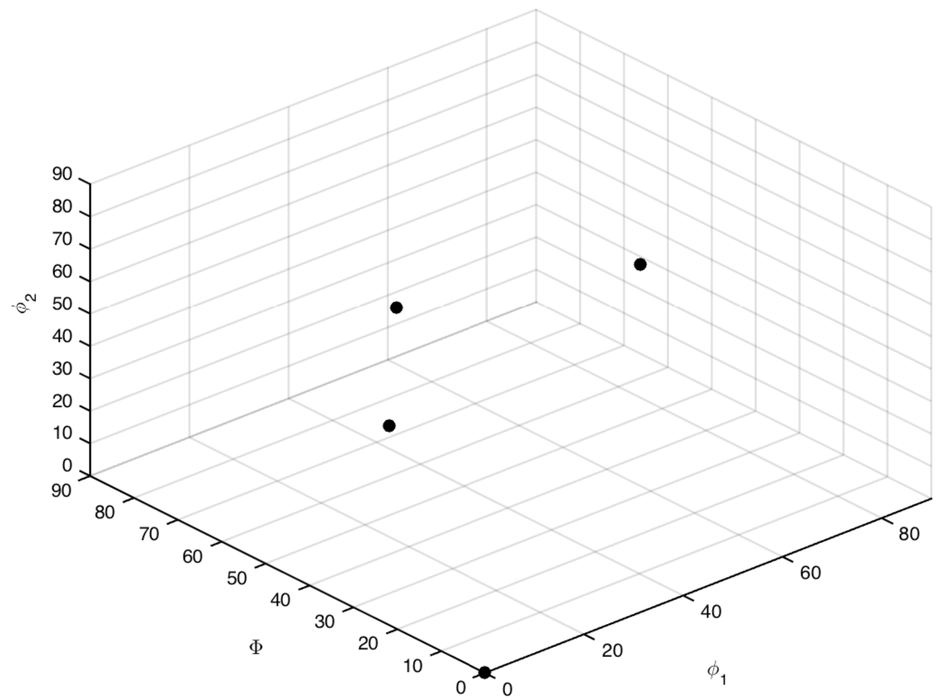
Used variants_xtal_sample.m (in Python/Matlab)

| i | j | phi1 | PHI | phi2 (degrees) |
|---|---|---|---|---|
| 1 | 1 | 32.0000 | 58.0000 | 18.0000 |
| 1 | 2 | 148.0000 | 122.0000 | 198.0000 |
| 1 | 3 | 328.0000 | 122.0000 | 198.0000 |
| 1 | 4 | 212.0000 | 58.0000 | 18.0000 |
| 2 | 1 | 212.0000 | 122.0000 | 162.0000 |
| 2 | 2 | 328.0000 | 58.0000 | 342.0000 |
| 2 | 3 | 148.0000 | 58.0000 | 342.0000 |
| 2 | 4 | 32.0000 | 122.0000 | 162.0000 |
| 3 | 1 | 243.5145 | 36.2406 | 153.6862 |
| 3 | 2 | 296.4855 | 143.7594 | 333.6862 |
| 3 | 3 | 116.4855 | 143.7594 | 333.6862 |
| 3 | 4 | 63.5145 | 36.2406 | 153.6862 |
| 4 | 1 | 63.5145 | 143.7594 | 26.3138 |
| 4 | 2 | 116.4855 | 36.2406 | 206.3138 |
| 4 | 3 | 296.4855 | 36.2406 | 206.3138 |
| 4 | 4 | 243.5145 | 143.7594 | 26.3138 |
| 5 | 1 | 212.0000 | 122.0000 | 342.0000 |
| 5 | 2 | 328.0000 | 58.0000 | 162.0000 |
| 5 | 3 | 148.0000 | 58.0000 | 162.0000 |
| 5 | 4 | 32.0000 | 122.0000 | 342.0000 |
| 6 | 1 | 32.0000 | 58.0000 | 198.0000 |
| 6 | 2 | 148.0000 | 122.0000 | 18.0000 |
| 6 | 3 | 328.0000 | 122.0000 | 18.0000 |
| 6 | 4 | 212.0000 | 58.0000 | 198.0000 |
| 7 | 1 | 63.5145 | 143.7594 | 206.3138 |
| 7 | 2 | 116.4855 | 36.2406 | 26.3138 |
| 7 | 3 | 296.4855 | 36.2406 | 26.3138 |
| 7 | 4 | 243.5145 | 143.7594 | 206.3138 |
| 8 | 1 | 243.5145 | 36.2406 | 333.6862 |
| 8 | 2 | 296.4855 | 143.7594 | 153.6862 |
| 8 | 3 | 116.4855 | 143.7594 | 153.6862 |
| 8 | 4 | 63.5145 | 36.2406 | 333.6862 |
| 9 | 1 | 32.0000 | 58.0000 | 108.0000 |
| 9 | 2 | 148.0000 | 122.0000 | 288.0000 |
| 9 | 3 | 328.0000 | 122.0000 | 288.0000 |
| 9 | 4 | 212.0000 | 58.0000 | 108.0000 |
| 10 | 1 | 311.7695 | 105.1924 | 123.3060 |
| 10 | 2 | 228.2305 | 74.8076 | 303.3060 |
| 10 | 3 | 48.2305 | 74.8076 | 303.3060 |
| 10 | 4 | 131.7695 | 105.1924 | 123.3060 |

| | | | | |
|---|---|---|---|---|
| 11 | 1 | 212.0000 | 122.0000 | 72.0000 |
| 11 | 2 | 328.0000 | 58.0000 | 252.0000 |
| 11 | 3 | 148.0000 | 58.0000 | 252.0000 |
| 11 | 4 | 32.0000 | 122.0000 | 72.0000 |
| 12 | 1 | 131.7695 | 74.8076 | 56.6940 |
| 12 | 2 | 48.2305 | 105.1924 | 236.6940 |
| 12 | 3 | 228.2305 | 105.1924 | 236.6940 |
| 12 | 4 | 311.7695 | 74.8076 | 56.6940 |
| 13 | 1 | 32.0000 | 58.0000 | 288.0000 |
| 13 | 2 | 148.0000 | 122.0000 | 108.0000 |
| 13 | 3 | 328.0000 | 122.0000 | 108.0000 |
| 13 | 4 | 212.0000 | 58.0000 | 288.0000 |
| 14 | 1 | 131.7695 | 74.8076 | 236.6940 |
| 14 | 2 | 48.2305 | 105.1924 | 56.6940 |
| 14 | 3 | 228.2305 | 105.1924 | 56.6940 |
| 14 | 4 | 311.7695 | 74.8076 | 236.6940 |
| 15 | 1 | 212.0000 | 122.0000 | 252.0000 |
| 15 | 2 | 328.0000 | 58.0000 | 72.0000 |
| 15 | 3 | 148.0000 | 58.0000 | 72.0000 |
| 15 | 4 | 32.0000 | 122.0000 | 252.0000 |
| 16 | 1 | 311.7695 | 105.1924 | 303.3060 |
| 16 | 2 | 228.2305 | 74.8076 | 123.3060 |
| 16 | 3 | 48.2305 | 74.8076 | 123.3060 |
| 16 | 4 | 131.7695 | 105.1924 | 303.3060 |
| 17 | 1 | 311.7695 | 105.1924 | 33.3060 |
| 17 | 2 | 228.2305 | 74.8076 | 213.3060 |
| 17 | 3 | 48.2305 | 74.8076 | 213.3060 |
| 17 | 4 | 131.7695 | 105.1924 | 33.3060 |
| 18 | 1 | 243.5145 | 36.2406 | 63.6862 |
| 18 | 2 | 296.4855 | 143.7594 | 243.6862 |
| 18 | 3 | 116.4855 | 143.7594 | 243.6862 |
| 18 | 4 | 63.5145 | 36.2406 | 63.6862 |
| 19 | 1 | 131.7695 | 74.8076 | 146.6940 |
| 19 | 2 | 48.2305 | 105.1924 | 326.6940 |
| 19 | 3 | 228.2305 | 105.1924 | 326.6940 |
| 19 | 4 | 311.7695 | 74.8076 | 146.6940 |
| 20 | 1 | 63.5145 | 143.7594 | 116.3138 |
| 20 | 2 | 116.4855 | 36.2406 | 296.3138 |
| 20 | 3 | 296.4855 | 36.2406 | 296.3138 |
| 20 | 4 | 243.5145 | 143.7594 | 116.3138 |
| 21 | 1 | 131.7695 | 74.8076 | 326.6940 |
| 21 | 2 | 48.2305 | 105.1924 | 146.6940 |
| 21 | 3 | 228.2305 | 105.1924 | 146.6940 |
| 21 | 4 | 311.7695 | 74.8076 | 326.6940 |
| 22 | 1 | 243.5145 | 36.2406 | 243.6862 |
| 22 | 2 | 296.4855 | 143.7594 | 63.6862 |
| 22 | 3 | 116.4855 | 143.7594 | 63.6862 |
| 22 | 4 | 63.5145 | 36.2406 | 243.6862 |

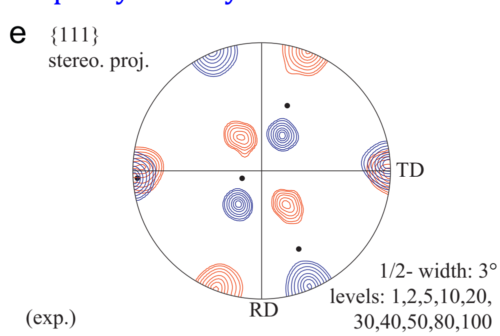
| | | | | |
|---|---|---|---|---|
| 23 | 1 | 311.7695 | 105.1924 | 213.3060 |
| 23 | 2 | 228.2305 | 74.8076 | 33.3060 |
| 23 | 3 | 48.2305 | 74.8076 | 33.3060 |
| 23 | 4 | 131.7695 | 105.1924 | 213.3060 |
| 24 | 1 | 63.5145 | 143.7594 | 296.3138 |
| 24 | 2 | 116.4855 | 36.2406 | 116.3138 |
| 24 | 3 | 296.4855 | 36.2406 | 116.3138 |
| 24 | 4 | 243.5145 | 143.7594 | 296.3138 |






*e.* How many points are listed in the new table?
24*4 = 96.
*f*. How many of the points lie within the range 0-90° for all three angles?
Three.

4. [15 points]
*This question, and the one following are intended to develop your skills for interpreting pole figures.*
*a*. [10] What sample symmetry does the following pole figure have, which is taken from a paper on texture (Quey *et al. J. Mech. Phys. Solids* **60** (2012) 509–524), if you include both the red and blue peaks?
*b*. [5] What sample symmetry is present if we only consider, say, the red peaks?
(a) The red and blue peaks taken together define an orthorhombic sample symmetry to high accuracy. If (b), we take only the red peaks then the only remaining symmetry is a diad (180° rotation) on the sample Z-axis, i.e. monoclinic sample symmetry.



Appendix

The following fortran90 code shows how to generate the 24 symmetry operators that belong to the (432) point group and that describe cubic crystal symmetry. Note that the first 6 lines of code specify the entries are all zero; the code that follows then changes the appropriate entries in each matrix to 1. This is more reliable than writing out (or assigning) each and every value. An exclamation point means that all characters after that (to the right of the !) are treated as a comment.
Note that all 24 matrices are listed in the Kocks-Tomé-Wenk book and in the lecture notes (and can be copied directly from there although be careful about corrections that are given in the lecture notes).

```
DO I=1,3
     DO J=1,3
        DO K=1,24
           SYM(I,J,K)=0.
        end do
     end do
  end do

  !   1
  SYM(1,1,1)=1.
  SYM(2,2,1)=1.
  SYM(3,3,1)=1.
  !   5
  SYM(1,1,2)=1.
  SYM(2,3,2)=-1.
  SYM(3,2,2)=1.
  !   2
  SYM(1,1,3)=1.
  SYM(2,2,3)=-1.
  SYM(3,3,3)=-1.
  !   11
  SYM(1,1,4)=1.
  SYM(2,3,4)=1.
  SYM(3,2,4)=-1.
  !   7
  SYM(1,3,5)=-1.
  SYM(2,2,5)=1.
  SYM(3,1,5)=1.
  !   12
```

```
SYM(1,3,6)=1.
SYM(2,2,6)=1.
SYM(3,1,6)=-1.
!   3
SYM(1,1,7)=-1.
SYM(2,2,7)=1.
SYM(3,3,7)=-1.
!   4
SYM(1,1,8)=-1.
SYM(2,2,8)=-1.
SYM(3,3,8)=1.
!   13
SYM(1,2,9)=1.
SYM(2,1,9)=-1.
SYM(3,3,9)=1.
!   6
SYM(1,2,10)=-1.
SYM(2,1,10)=1.
SYM(3,3,10)=1.
!   20
SYM(1,2,11)=-1.
SYM(2,3,11)=1.
SYM(3,1,11)=-1.
!   23
SYM(1,3,12)=1.
SYM(2,1,12)=-1.
SYM(3,2,12)=-1.
!   19
SYM(1,2,13)=-1.
SYM(2,3,13)=-1.
SYM(3,1,13)=1.
!   21
SYM(1,3,14)=-1.
SYM(2,1,14)=1.
SYM(3,2,14)=-1.
!   18
SYM(1,2,15)=1.
SYM(2,3,15)=-1.
SYM(3,1,15)=-1.
!   22
SYM(1,3,16)=-1.
SYM(2,1,16)=-1.
SYM(3,2,16)=1.
!   9
SYM(1,2,17)=1.
SYM(2,3,17)=1.
SYM(3,1,17)=1.
!   8
```

```
SYM(1,3,18)=1.
SYM(2,1,18)=1.
SYM(3,2,18)=1.
!   17
SYM(1,2,19)=1.
SYM(2,1,19)=1.
SYM(3,3,19)=-1.
!   24
SYM(1,1,20)=-1.
SYM(2,3,20)=1.
SYM(3,2,20)=1.
!   10
SYM(1,3,21)=1.
SYM(2,2,21)=-1.
SYM(3,1,21)=1.
!   15
SYM(1,1,22)=-1.
SYM(2,3,22)=-1.
SYM(3,2,22)=-1.
!   16
SYM(1,3,23)=-1.
SYM(2,2,23)=-1.
SYM(3,1,23)=-1.
!   14
SYM(1,2,24)=-1.
SYM(2,1,24)=-1.
SYM(3,3,24)=-1.
```