Volume Fractions of Texture Components

A. D. Rollett

27-750

Texture, Microstructure & Anisotropy

Last revised: 28th Jan. '14

Lecture Objectives

- Explain how to compute intensities in a discrete OD from counts of grains, points or volumes.

- Define *volume fraction* as the fraction of material whose orientation lies within a specified range of orientations.

- Explain how to calculate volume fractions given a discrete orientation distribution.

- Describe the calculation of *orientation distance* as a subset of the calculation of *misorientations*. Also discuss how to apply symmetry, and some of the pitfalls.

Notation

g - orientation

$f(g)$ - orientation distribution

O - symmetry operator

O_{432} - 432 point group

O_C - crystal symmetry operator

O_S - sample symmetry operator

tr - trace of a matrix

Δg - misorientation

V_f – physical volume fraction of grain, material, orientation

ϕ_1, Φ, ϕ_2 - Euler angles

$\Omega, \Delta\Omega$ - Volume of orientation space, increment of volume

popLA – preferred orientation package Los Alamos, a software package written in the early 1990s to analyze texture data, based on the DOS operating system.

In Class Questions

- How do we compute intensities from volume fractions?

- What is the difference between cell-edge and cell-centered coordinates?

- What do we mean by the *volume fraction* of a *texture component*?

- How do we compute the volume fraction of a component?

- What is a *capture angle* or *tolerance angle* or *capture radius*?

- What is a *partition map*?

- Why is the density of randomly chosen orientations (in Euler space) not uniform?

- Describe two ways to compute *orientation distance*.

- Why is *misorientation* useful for computing orientation distance?

- How do we apply symmetry to compute misorientation?

- Why do different components (e.g. cube vs. S) have different volume fractions in a random texture?

Intensity from Volume Fractions

Objective: given information on volume fractions (e.g. numbers of grains of a given orientation), how do we calculate the intensity in the OD?

• General relationships:

$$V_f(g) = \int f(g)\,dg$$

$$f(g) = \frac{1}{V}\frac{dV(g)}{dg} = \Omega\frac{\Delta V_f}{\Delta\Omega}\bigg|_g$$

Reminder on units of the OD: the way that normalization is performed means that the units of the OD are Multiples of a Random Density (MRD). If there is no texture or preferred orientation then the value of "f" is one everywhere.

Intensity from V_f, contd.

- For each cell, we assign an intensity equal to the volume fraction in that cell, ΔV, divided by the volume of orientation space associated with that cell, $\Delta \Omega$ and multiplied by the total volume of orientation space .

- If we have points with equal area or volume (e.g. EBSD data) the volume fraction, ΔV_i, is simply the number of points in the ith cell, n_i, divided by the total number of points, N: $\Delta V_i = n_i / N.$

- For 5x5x5° discretization in a 90x90x90° space, we particularize to:

$$V_f(g) = \frac{1}{8100^{°2}} \int f(g) \sin \Phi \, d\Phi \, d\varphi_1 \, d\varphi_2$$

$$f(g) = \frac{1}{V} \frac{dV(g)}{dg} = \Omega \frac{\Delta V_f}{\Delta \Omega}\bigg|_g$$

$$= 8100^{°2} \frac{\Delta V_f}{25^{°2}\left(\cos\left[\Phi - 2.5°\right] - \cos\left[\Phi + 2.5°\right]\right)}$$

Discrete OD

- Normalization also required for discrete OD

- Normalization depends on the size of the sub-space of orientation, and on the measure used (radians versus degrees).

- Sum the intensities over all the cells.

- $0 \leq \phi_1 \leq 2\pi, \ 0 \leq \Phi \leq \pi, \ 0 \leq \phi_2 \leq 2\pi$

$$1 = \frac{1}{8\pi^2} \sum_{\phi_1} \sum_{\Phi} \sum_{\phi_2} f\left(\phi_1, \Phi_i, \phi_2\right) \Delta\phi_1 \Delta\phi_2 \left(\cos\left(\Phi_i - \frac{\Delta\Phi}{2} \right) - \cos\left(\Phi_i + \frac{\Delta\Phi}{2} \right) \right)$$

$0 \leq \phi_1 \leq 90°, \ 0 \leq \Phi \leq 90°, \ 0 \leq \phi_2 \leq 90°$

$$1 = \frac{1}{8100} \sum_{\phi_1} \sum_{\Phi} \sum_{\phi_2} f\left(\phi_1, \Phi_i, \phi_2\right) \Delta\phi_1 \Delta\phi_2 \left(\cos\left(\Phi_i - \frac{\Delta\Phi}{2} \right) - \cos\left(\Phi_i + \frac{\Delta\Phi}{2} \right) \right)$$
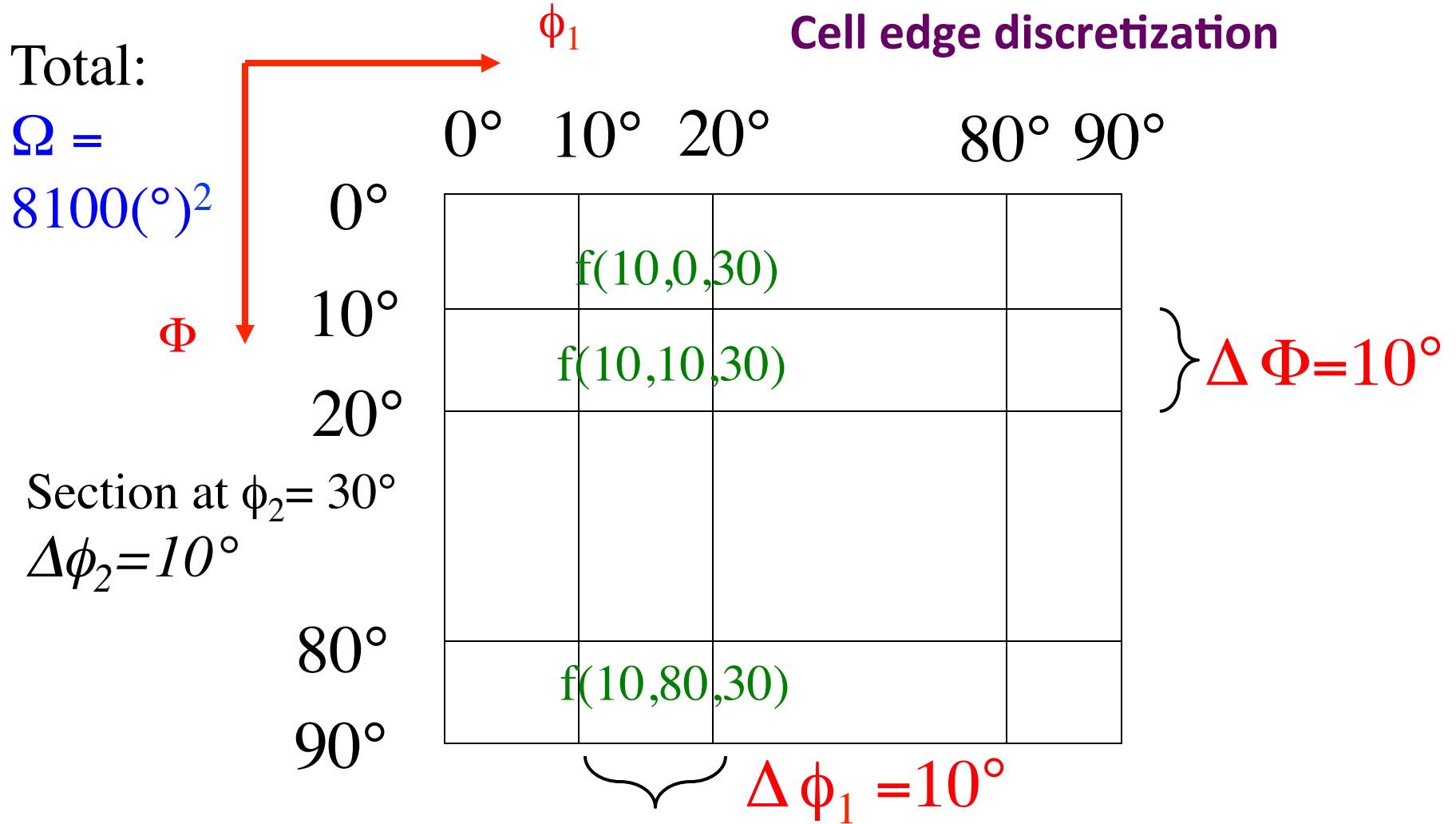
Volume fraction calculations

- Choice of cell size determines size of the volume increment, which depends on the value of the second angle (Φ or Θ).

- Some grids start at the specified value.

- More typical for the specified value to be in the center of the cell.

- popLA: grids are cell-centered.

Discrete ODs

$$dA = \sin\Phi\, d\Phi\, d\phi_1:$$

Each layer: $\Delta\Omega = \Sigma \Delta A\, \Delta\phi_2 = 90(°)$
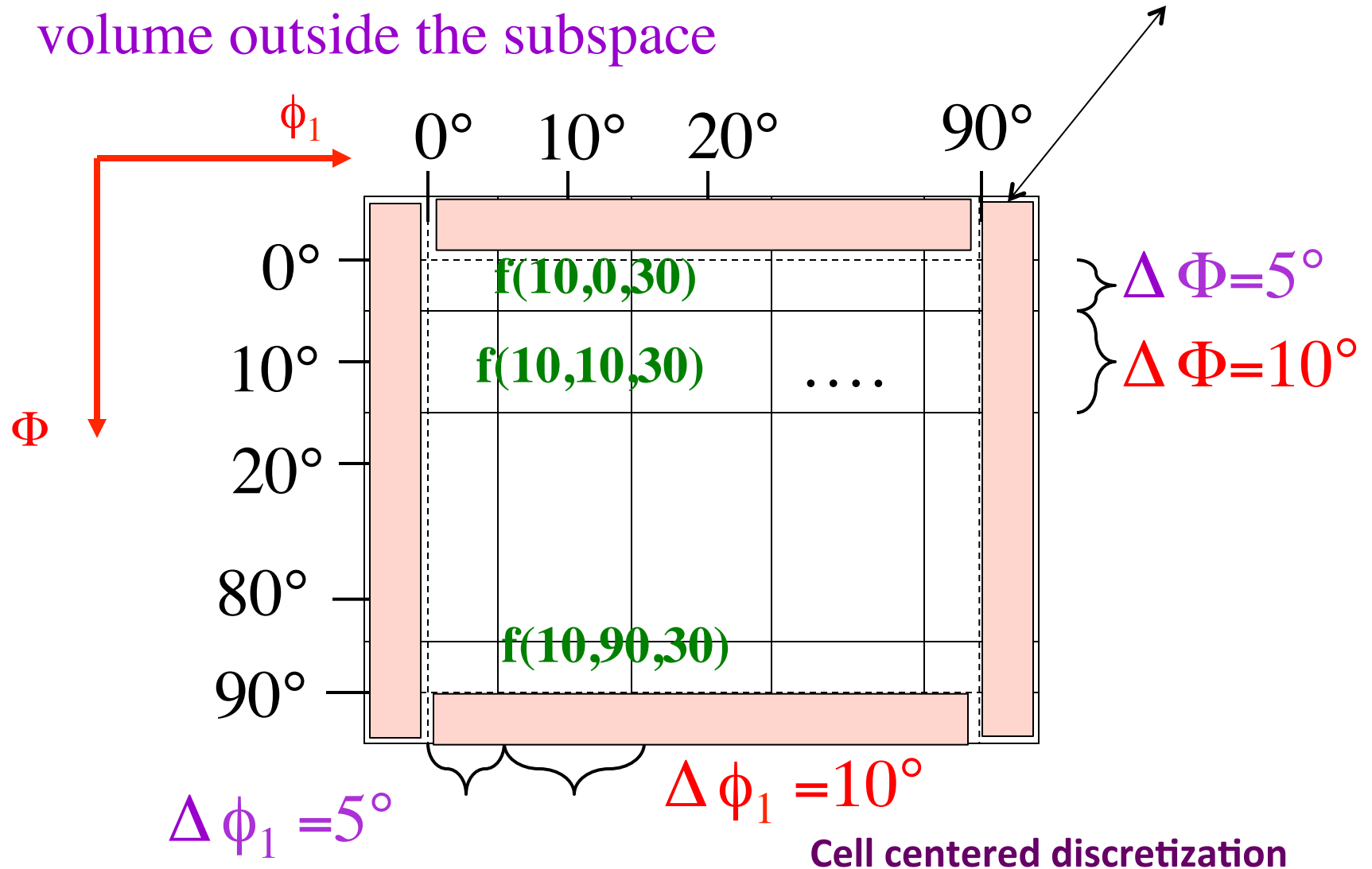
$$\Delta A = \Delta(\cos\Phi)\Delta\phi_1$$

ϕ_1

Cell edge discretization

Total:

$\Omega = 8100(°)^2$

	0°	10°	20°		80°	90°
0°						
		f(10,0,30)				
10°						
		f(10,10,30)				
20°						

Φ

$\Delta\Phi = 10°$

Section at $\phi_2 = 30°$

$\Delta\phi_2 = 10°$

| 80° | | f(10,80,30) | | | | |
| 90° | | | | | | |

$\Delta\phi_1 = 10°$

Centered Cells

$$dA = sin\Phi d\Phi d\phi_1:$$
$$\Delta A = \Delta(cos\Phi)\Delta\phi_1$$

Different treatment of end and corner cells to exclude volume outside the subspace

ϕ_1

$0°$ $10°$ $20°$ $90°$

Φ

$0°$ f(10,0,30) $\Delta\Phi=5°$

$10°$ f(10,10,30) .... $\Delta\Phi=10°$

$20°$

$80°$

$90°$ f(10,90,30)

$\Delta\phi_1 = 5°$ $\Delta\phi_1 = 10°$

Cell centered discretization

Discrete orientation information

Typical text data.ANG file from TSL EBSD system:

WorkDirectory   /usr/OIM/rollett   *Each dataset may*
# OIMDirectory   /usr/OIM   *have millions of points*

…………..

# ϕ_1	Φ	ϕ_2	x	y				
4.724	0.234	4.904	0.500	0.866	1.0	1.000	0	0
4.491	0.024	5.132	7.500	0.866	1.0	1.000	0	0
4.932	0.040	4.698	19.500	0.866	1.0	1.000	0	0
4.491	0.024	5.132	20.500	0.866	1.0	1.000	0	0
4.491	0.024	5.132	21.500	0.866	1.0	1.000	0	0
4.932	0.040	4.698	22.500	0.866	1.0	1.000	0	0
4.932	0.040	4.698	23.500	0.866	1.0	1.000	0	0
4.932	0.040	4.698	24.500	0.866	1.0	1.000	0	0

(Euler angles, radians)   (spatial coordinates, microns)

Binning individual orientations in a discrete OD

ϕ_1

0° 10° 20° 80° 90°

Φ

0°
10°
20°

$\Delta\Phi=10°$

Section at $\phi_2= 30°$

individual orientation

80°
90°

$\Delta\phi_1 =10°$
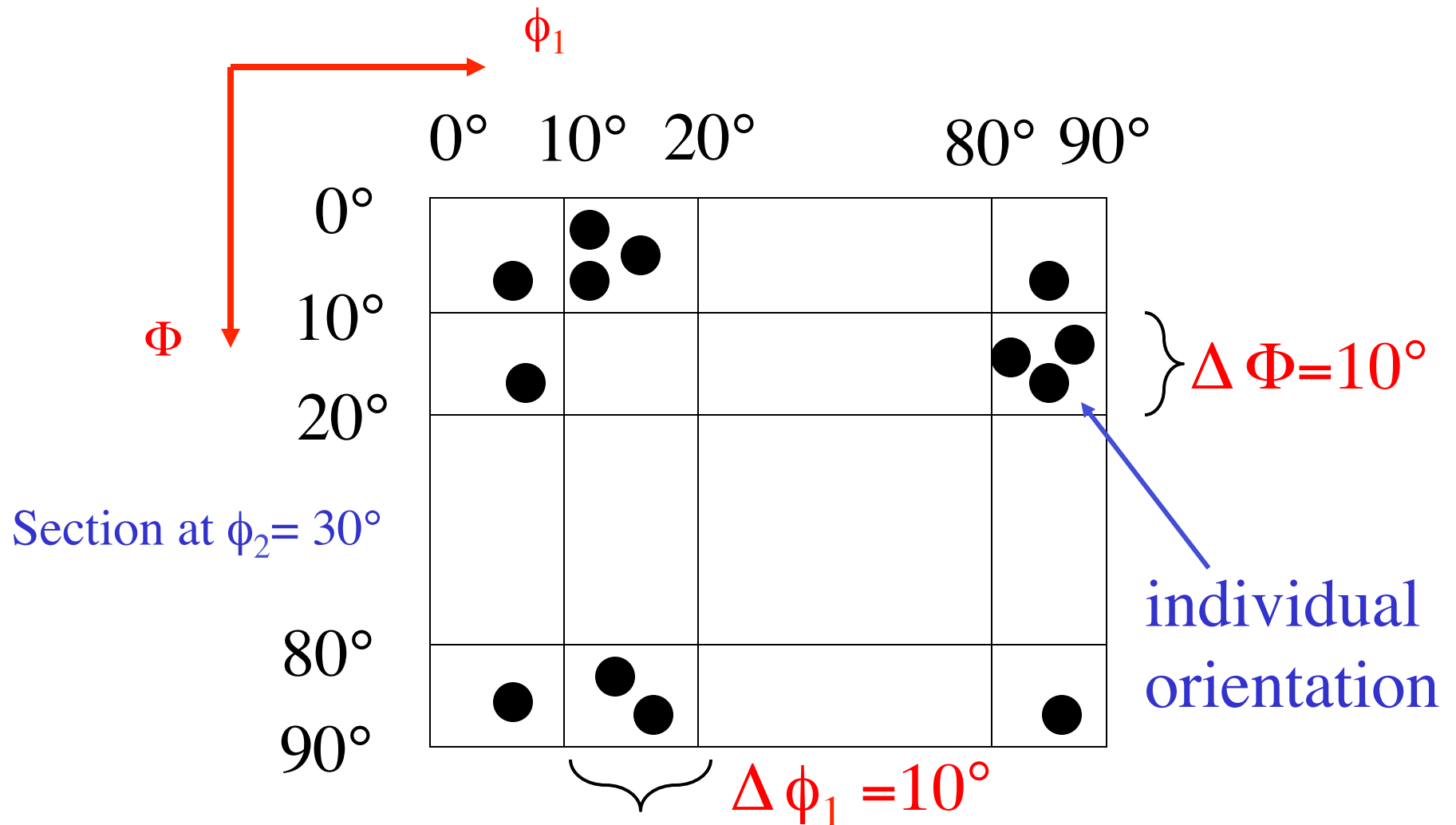
Example of random orientation distribution in Euler space
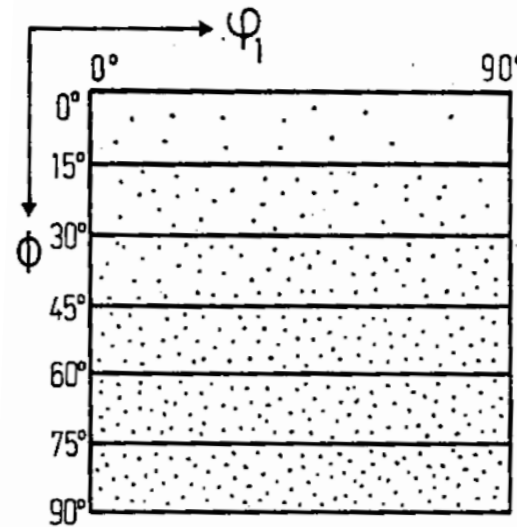
[Bunge]

Figure 2.25 On the definition of the invariant measures by a random distribution in the EULER space $\varphi_1 \Phi \varphi_2$

Note the smaller densities of *points* (arbitrary scale) near Φ = 0°. When converted to *intensities*, however, then the result is a uniform, constant value of the OD (because of the effect of the volume element size, $\sin\Phi d\Phi d\phi_1 d\phi_2$). If a material had randomly oriented grains all of the same size then this is how they would appear, as individual points in orientation space.

OD from discrete points: pseudo-code

1. Compute the volume of the chosen orientation space, e.g. Euler space, e.g. 90° x 90° x 90° $\Rightarrow \Omega = \int dg = 8100$ °². Also compute the volume of each cell; in this case
    $$d\Omega(\phi_1, \Phi, \phi_2) = \Delta(\cos\Phi)\Delta\phi_1\Delta\phi_2$$

2. Bin each orientation into a cell in the OD

3. Sum number (or weight, if each orientation represents a different physical volume) in each cell

4. Divide the number (or physical volume) in each cell by the total number of grains (or total physical volume) to obtain V_f

5. Convert from
    V_f to $f(g)$:
    $$f(g) = V_f * \Omega/d\Omega = 8100\ V_f / \{\Delta(\cos\Phi)\Delta\phi_1\Delta\phi_2\}$$

    $$\Omega = \int dg \qquad d\Omega = \text{cell volume}$$

Discrete OD from points

- The same V_f near Φ=0° will have much larger $f(g)$ than cells near Φ=90°.

- Unless large number (>10^4, texture dependent) of grains are measured, the resulting OD will be noisy, i.e. large variations in intensity between cells.

- Typically, smoothing is used to facilitate presentation of results: always do this last and as a visual aid only!

- An alternative to smoothing an ODF plot is to replace individual points by Gaussians and then evaluate the texture. This is particularly helpful (and commonly applied) when performing a series expansion fit to a set of individual orientation measurements, such as OIM data.

Volume fraction calculation

- In its simplest form: sum up the intensities multiplied by the value of the volume increment (invariant measure) for each cell.

- Check that when you compute this sum for the entire space the result is equal to one (else the normalization is not correct).

Simple results

- What if all the grains fall in one cell?
- Answer: the intensity (in units of Multiples of a Random Density, MRD) in that cell depends on its location in the space (assuming uniformly divided space by angle), and the intensity in all other cells is exactly zero.
- Example: cell-edge coordinates with 10° increments in a 90x90x90 space (total volume, Ω = 8100 (°²);
  each cell has volume
  $\Delta\Omega(g) = 100 \times \Delta\cos(\Phi))$.
  Note that the sum of the second column is one, as it must be for the integral of $\sin(\Phi)$ over the interval $[0..\pi/2]$.
- More exactly: $f(g) = \Omega / \Delta\Omega(g)$
- The magnitude of the peak depends on the cell size in relation to the volume, not on the measure (*degrees* versus *radians*).

Φ (°)	$\Delta\cos\Phi$(g)	Intensity (MRD)
0	0.0152	5331
10	0.0451	1795
20	0.0737	1100
30	0.1	810
40	0.1233	657
50	0.1428	567
60	0.1580	512
70	0.1684	481
80	0.1736	466

Simple results: 2

- What if all, say, 9/10ths of the grains fall in one cell and the rest are randomly distributed?
- Answer: the intensity in that cell once again depends on its location in the space (assuming uniformly divided space by angle), and the intensity in all other cells is fixed at 0.1/{remaining volume}, where the "remaining volume" is the total volume minus the volume of the individual cell that contains 9/10ths of the volume.
- Example: cell-edge coordinates with 10° increments in a 90x90x90 space (total volume = 8100; each cell has volume 100xΔcos(Φ)). Note that the sum of the second column is one.
- More exactly (in the single cell):
  $f(g) = 0.9 \ \Omega \ / \ \Delta\Omega(g)$, - "peak"
  or elsewhere, since $\int f(g) = \Omega$
  $f(g) = 0.1 \ \Omega \ / \ (\Omega - \Delta\Omega(g))$ - "random"

Φ (°)	$\Delta\cos\Phi(g)$	Intensity in Cell (MRD)	$f(g)$ elsewhere (10⁻⁵ MRD)
0	0.0152	4779	0.10002
10	0.0451	1615	0.10006
20	0.0737	990	0.10009
30	0.1	729	0.10012
40	0.1233	591	0.10015
50	0.1428	510	0.10017
60	0.1580	461	0.10019
70	0.1684	433	0.10020
80	0.1736	420	0.10021

Texture Component Fractions

- Now we discuss how to compute the **volume fraction of material associated with a particular texture component**.

- The physical analogy is, how many (equal sized) grains will we find in a material that correspond to a particular texture component?

- The simplest way to think about volume fractions is to consider that all cells within a certain angle of the location of the position of the texture component of interest belong to that component.

- Although we will need to use the concept of orientation distance (equivalent to misorientation), for now we can use a fixed angular distance or *acceptance angle* to decide which component a particular cell belongs to.

Acceptance Angle Schematic

- In principle, one might want to weight the intensity in each cell as a function of distance from the component location.
- For now, however, we will assign equal weight to all cells included in the volume fraction estimate.
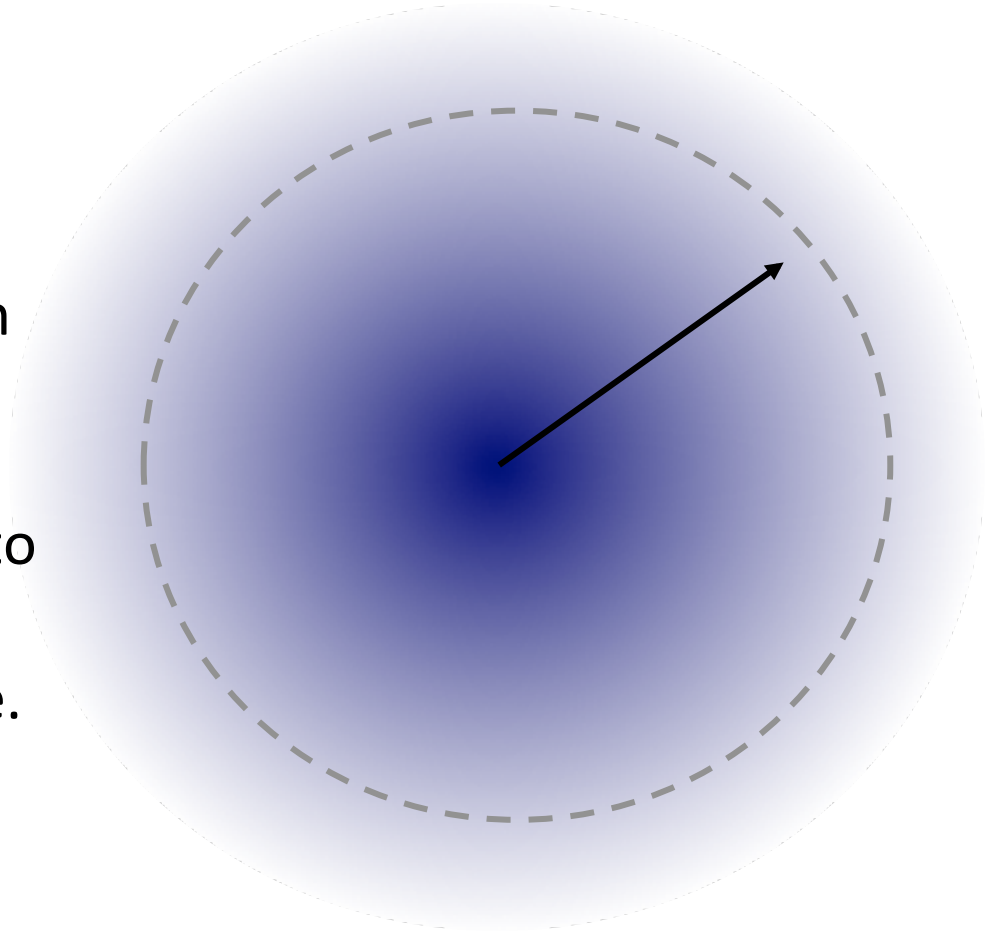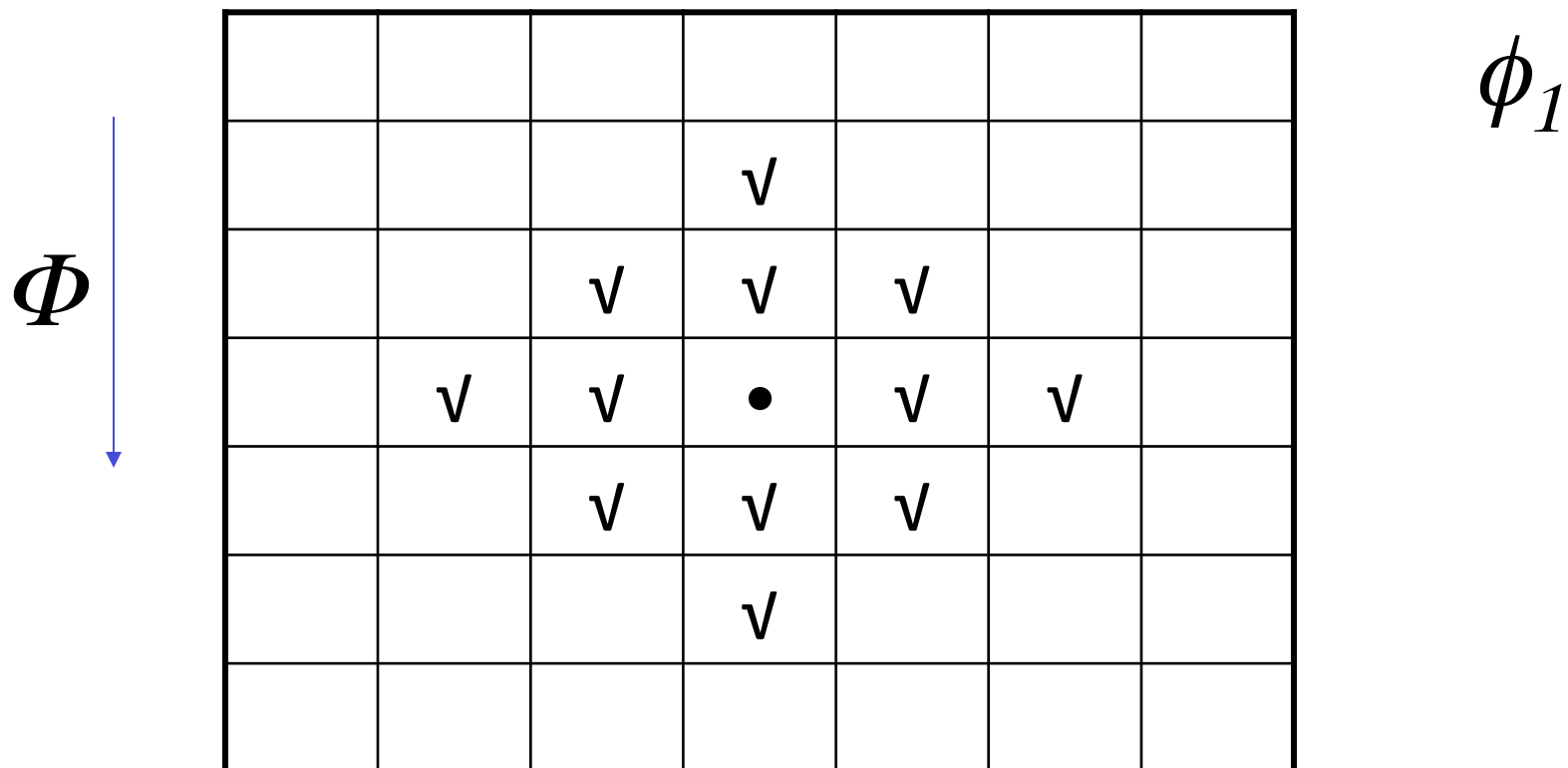
Illustration of Acceptance Angle

- As a basic approach, include all cells within 10° of a central location.

ϕ_1

Φ

Copper component example

15° acceptance angle; location of
maximum intensity 5° off **ideal position**

```
CUR80-2 6/13/88              35 Bwimv iter: 2.0%FON=  0 13-APR-** strength= 2.43
 CODK  5.0 90.0  5.0 90.0 1 1 1 2 3  100       phi= 45.0
   15   12    8    3    3    6   14   42   89   89   89   42   14    6    3    3    8   12   15
    5    5    5    6    8   20   43   53   57   65   65   45   21   14   12   10    8    9    7
   12   11   10   14   20   30   60  118  136   84   49   16    2    1    1    1    2    4    5
   22   21   32   49   68   81  100  123  132  108   37   12    6    3    3    3    3    2    1
  321  284  228  185  172  190  207  178  109   48   19    7    5    5    4    3    3    1    1
  955  899  770  575  389  293  223  131   55   12    3    2    2    1    1    1    0    0    0
 1730 1547 1100  652  382  233  132   62   23    7    2    1    1    1    1    0    1    0    0
 1513 1342  881  436  191   90   53   29   17    6    2    1    0    0    1    0    0    0    0
  137  135  109   77   59   41   24   10    4    2    1    0    0    0    0    0    0    0    0
    1    0    1    3    5   10   13   14   10    3    1    1    0    0    0    0    0    0    0
    0    1    1    1    1    1    1    1    1    0    0    0    0    0    0    0    0    0    0
    0    0    0    1    1    1    1    1    1    1    0    0    0    0    0    1    1    1    1
    0    0    0    0    1    0    1    2    2    1    1    1    2    2    3    4    5    6    7
    0    0    0    0    1    1    2    4    5    5    5    4    3    6    8    6    6    7    5
    2    2    2    2    2    2    2    2    4    3    3    3    4    7    9    6   12   17   16
    3    4    4    4    4    7   33   80   86   66   42   29   29   31   33   40   51   46   40
    7    7    9   14   31   71  144  179  145   81   31   11    7    7   10   17   25   23   23
  203  190  188  193  224  304  417  486  410  249  109   51   36   26   16   12   12   12    8
  301  315  404  559  752 1005 1180 1140  861  494  237  132   65   42   29   26   31   30   30
```
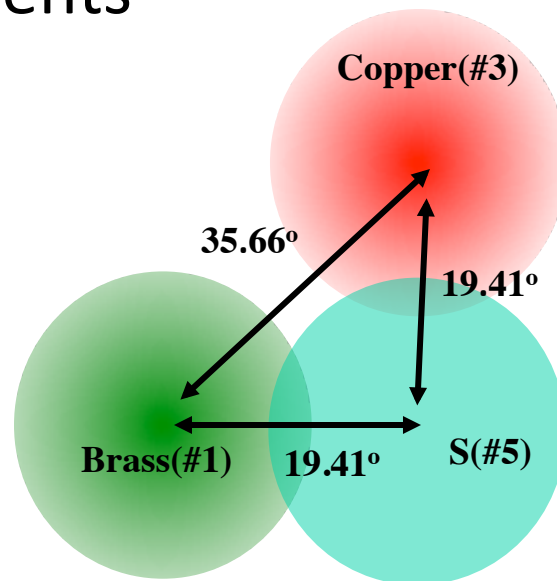
ϕ_1

Φ

Partitioning Orientation Space

- Problem!
- If one chooses too large an acceptance angle, overlap occurs between different components

- Solution:
- It is necessary to go through the entire space and *partition* the space into separate regions with one subregion for each component. Each cell is assigned to the "nearest" component.

Copper(#3)

35.66°

19.41°

Brass(#1)   **19.41°**   **S(#5)**

Distance in Orientation Space

- What does "distance" mean in orientation space?

- Note: distance is *not* the Cartesian distance (Pythagorean, $\sqrt{\{\Delta x^2 + \Delta y^2 + \Delta z^2\}}$)

- This is an issue because the volume increment varies with [the *sine* of the] the 2nd Euler angle.

- Answer:

- Distance in orientation space is measured by *misorientation*.

- This provides a better method for partitioning the space.

- Misorientation distance is the *minimum available rotation angle* between a pair of orientations.

Partitioning by Misorientation

- Compute misorientation by "reversing" one orientation and then applying the other orientation. More precisely stated, compose the inverse of one orientation with the other orientation.

- $|\Delta g| = min_{ij}\{\ cos^{-1}(\ \{\ tr([O_i^{xtal}g_A O_j^{sample}]g_B^{\ T})\ -1\}/2\ )\}$

- The symbol "*tr()*" means the trace of (sum of leading diagonal entries) of the matrix within the parentheses. The *minimum* function indicates that one chooses the particular combination of crystal symmetry operator, $O_i \in O_{432}$, and sample symmetry operator, $O_j \in O_{222}$, that results in the smallest angle (for cubic crystals, computed for all 24 proper rotations in the crystal symmetry point group). Thus $i=1..24$ and $j=1..4$.

- Superscript *T* indicates (*matrix) transpose* which gives the inverse rotation. Subscripts *A* and *B* denote first and second component. For this purpose, the order of the rotations does not matter (but it will matter when the rotation axis is important!).

- Note that including the symmetry operators allows points near the edges of orientation space to be close to each other, even though they may be at opposite edges of the space.

- More details provided in later slides.

Partitioning by Misorientation, contd.

- For each point (cell) in the orientation space, compute the misorientation of that point with every component of interest (including all 3 variants of that component within the space); this gives a list of, say, six misorientation values between the cell and each of the six components of interest.

- Assign the point (cell) to the component with which it has the smallest misorientation, provided that it is less than the acceptance angle.

- If a point (cell) does not belong to a particular component (because it is not close enough), label it as "other" or "random".

Partition Map, COD, $\phi_2 = 0°$

```
 Acceptance angle (degrees) =   15.
AL       3/08/02               99 WIMV iter: 1.2%,Fon=  0 20-MAY-** strength= 3.88
 CODB  5.0 90.0  5.0 90.0 1 1 1 2 3    0 6859Phi2=  0.0
```

Cube

Cube

```
  1 1 1  4  4  4  4  0  0  0  0  0  4  4  4  4  1  1  1
  1 1 1  4  4  4  4  0  0  0  0  0  4  4  4  4  1  1  1
  2 2 2  4  4  4  4  0  0  0  0  0  4  4  4  4  5  5  5
  2 2 2  2  4  0  0  0  0  0  0  0  0  0  0  0  5  5  5
  2 2 2  0  0  0  0  0  0  0  0  0  0  0  0  0  5  5  5
  2 2 2  0  0  0  9  9  9  0  0  0  0  0  0  0  0  0  0
  3 3 3  7  0  9  9  9  9  9  0  0  0  0  0  0  0  0  0
  3 3 3  7  7  9  9  9  9  9  0  0  0  0  0  0  0  0  0
  3 3 7  7  7  7  8  8  8  8  0  0  0  0  0  0  0  0  0
  3 7 7  7  7  7  8  8  8  8  0  0  0  0  0  0  0  0  0
  3 3 7  7  7  7  8  8  8  8  0  0  0  0  0  0  0  0  0
  3 3 3  7  7  9  9  9  9  9  0  0  0  0  0  0  0  0  0
  3 3 3  7  0  9  9  9  9  9  0  0  0  0  0  0  0  0  0
  2 2 2  0  0  0  9  9  9  0  0  0  0  0  0  0  0  0  5
  2 2 2  0  0  0  0  0  0  0  0  0  0  0  0  0  5  5  5
  2 2 2  0  4  0  0  0  0  0  0  0  0  0  0  0  5  5  5
  2 2 2  4  4  4  4  0  0  0  0  0  4  4  4  4  5  5  5
  1 1 1  4  4  4  4  0  0  0  0  0  4  4  4  4  1  1  1
  1 1 1  4  4  4  4  0  0  0  0  0  4  4  4  4  1  1  1
```

Brass

Cube

Cube

The number in each cell indicates which component it belongs to. 0 = "random"; 8 = Brass; 1 = Cube.

Partition Map, COD, $\phi_2 = 45°$

```
AL        3/08/02              99 WIMV iter: 1.2%,Fon=   0 20-MAY-** strength= 3.88
 CODB   5.0 90.0   5.0 90.0 1 1 1 2 3     0 6859Phi2= 45.0
      0    0    4    4    4    4    4    1    1    1    1    1    4    4    4    4    4    0    0
      0    0    0    4    4    4    4    1    1    1    1    1    4    4    4    4    0    0    0
      0    0    0    4    4    4    4    6    6    6    6    6    4    4    4    4    0    0    0
      0    0    0    0    0    0    6    6    6    6    6    6    6    0   11   11   12   12   12
      0    0    0    0    0    0    0    6    6    6    6    6    0   11   11   11   12   12   12
      0    0    0    0    0    0    0    6    6    6    6    6    0   11   11   11   12   12   12
      0    0    0    0    0    0    0    0    0    6    0    0    0   11   11   11   12   12   12
      0    0    0    0    0    0    0    0    0    0    0    0    0   11   11   11   12   12   12
      0    0    0    0    0    0    0    0    0    0    0    0    0    0   11   11   12   12   12
      0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
      0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
      0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
      0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
      0    0    0    0    0    0    0    0    0   10   10    0    0    0    0    0    0    0    0
      0    0    0    0    0    0    0    0   10   10   10   10    0    0    0    0    0    0    0
      0    0    0    0    0    0    0    0    9    9    9    9    9    7    7    7    7    3
      0    0    0    0    0    0    0    0    8    8    8    8    7    7    7    7    7    3
      0    0    0    0    0    0    0    0    8    8    8    8    7    7    7    7    7    3
```
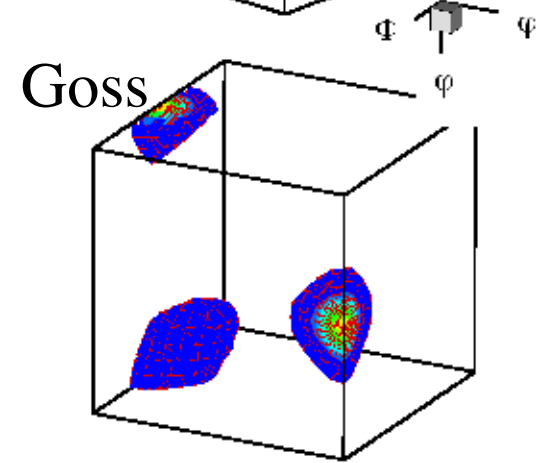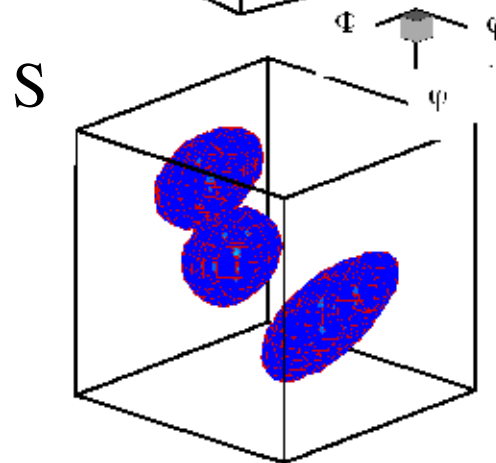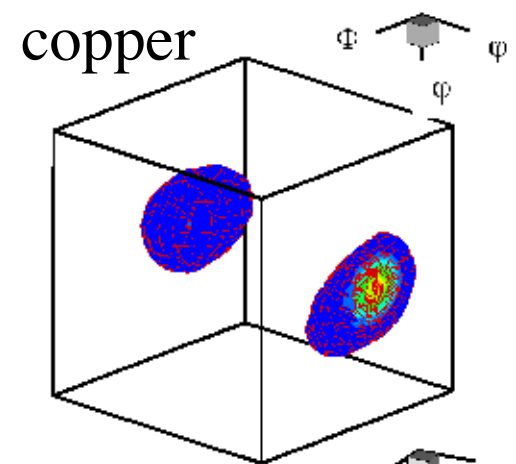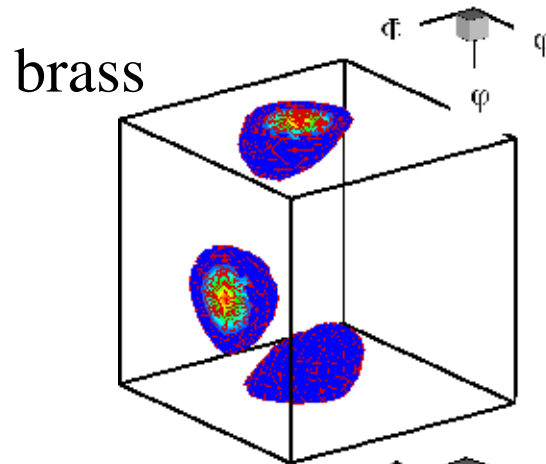
Copper

Brass

Component numbers: 0:=random; 8:=Brass; 11:= Dillamore; 12:=Copper.

Component Volumes: fcc rolling texture

- These contour maps of individual components in Euler space are drawn for an acceptance angle of ~12°.

brass

copper

S

Goss

Cube
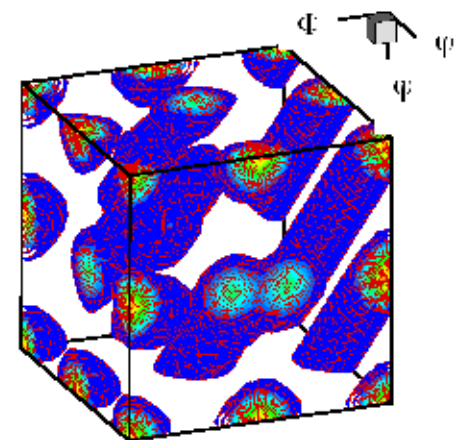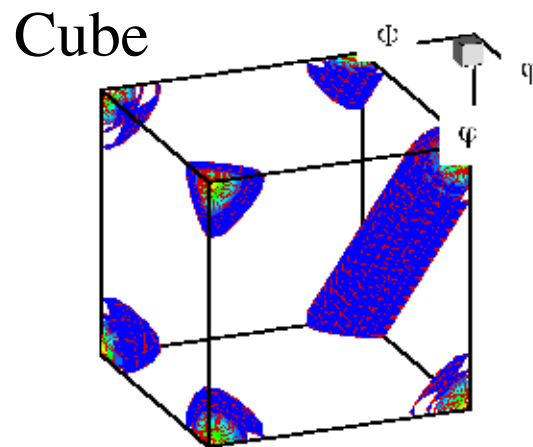
How to calculate misorientation?

- The next set of slides describe how to calculate misorientations, how to deal with crystal symmetry and sample symmetry, and some of the pitfalls that can arise.

- For *orientation distance*, only the magnitude of the difference in orientation needs to be calculated. Therefore some of the details that follow go beyond what you need for volume fraction. Nevertheless, you need to be aware of these issues so that you do not become confused in subsequent exercises.

- This misorientation calculation is *not* available in popLA but *is* available in TSL/HKL software. It is completely reliable but does not allow you to control the application of symmetry.

Objective

- To make clear how it is possible to express a misorientation in more than (physically) equivalent fashion.

- To allow researchers to apply symmetry correctly; mistakes are easy to make!

- It is *essential* to know how a rotation/orientation/texture component is expressed in order to know how to apply symmetry operations.

Worked Example

- In this example, we take a pair of orientations that were chosen to have a 60°<111> misorientation between them (rotation axis expressed in crystal coordinates). In fact the pair of orientations are the two sample symmetry related Copper components. The Copper component is nominally (112)[11-1].

- We calculate the 3x3 Rotation matrix for each orientation, g_A and g_B, and then form the misorientation matrix, $\Delta g = g_B g_A^{-1}$.

- From the misorientation matrix, we calculate the angle, $= \cos^{-1}(\mathrm{trace}(\Delta g) - 1)/2)$, and the rotation axis.

- In order to find the smallest possible misorientation angle, we have to apply crystal symmetry operators, O, to the misorientation matrix, $O\Delta g$, and recalculate the angle and axis.

- First, let's examine the result….

Worked Example

```
angles..      90.   35.2599983   45.
angles..      270.  35.2599983   45.


1st Grain: Euler angles:   90.   35.2599983   45.
2nd Grain: Euler angles:   270.  35.2599983   45.


1st matrix:
[     -0.577        0.707        0.408 ]
[     -0.577       -0.707        0.408 ]
[      0.577        0.000        0.817 ]


2nd matrix:
[      0.577       -0.707        0.408 ]
[      0.577        0.707        0.408 ]
[     -0.577        0.000        0.817 ]


Product matrix for gA X gB^-1:
[     -0.667        0.333        0.667 ]
[      0.333       -0.667        0.667 ]
[      0.667        0.667        0.333 ]
 MISORI: angle=    60. axis=   1 1 -1
```
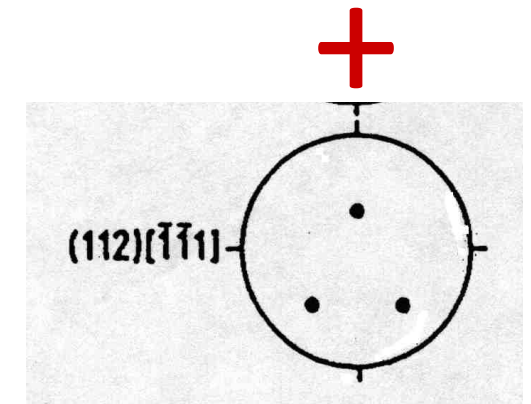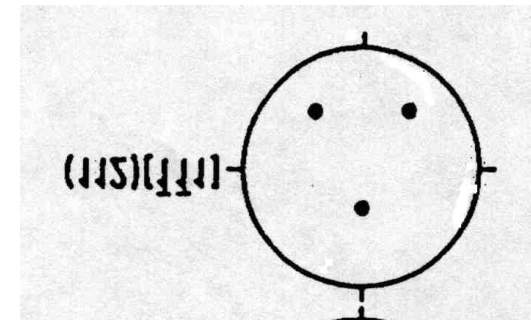
{100} pole figures

(1̄1S)[1̄1̄1]

+

(112)[1̄1̄1]

As it happens, the result is 60°[11-1], which looks reasonable, but is it, in fact, the smallest angle?

Output with Symmetry Applied

1st matrix:
[-0.691   0.596   0.408]
[-0.446   -0.797   0.408]
[0.569   0.100   0.817]

2nd matrix:
[0.691   -0.596   0.408]
[0.446   0.797   0.408]
[-0.569   -0.100   0.817]

Symmetry operator number 1
Product matrix for gA X gB^-1:
[-0.667   0.333   0.667]
[0.333   -0.667   0.667]
[0.667   0.667   0.333]
Trace = -1.
 angle =   180.

Symmetry operator number 2
Product matrix for gA X gB^-1:
[-0.667   0.333   0.667]
[-0.667   -0.667   -0.333]
[0.333   -0.667   0.667]
Trace = -0.666738808
 angle =   146.446426

Symmetry operator number 3
Product matrix for gA X gB^-1:
[-0.667   0.333   0.667]
[-0.333   0.667   -0.667]
[-0.667   -0.667   -0.333]
Trace = -0.333477736
 angle =   131.815857

Symmetry operator number 4
Product matrix for gA X gB^-1:
[-0.667   0.333   0.667]
[0.667   0.667   0.333]
[-0.333   0.667   -0.667]
Trace = -0.666738927
 angle =   146.446442

Symmetry operator number 5
Product matrix for gA X gB^-1:
[-0.667   -0.667   -0.333]
[0.333   -0.667   0.667]
[-0.667   0.333   0.667]
Trace = -0.666738987
 angle =   146.446442

Symmetry operator number 6
Product matrix for gA X gB^-1:
[0.667   0.667   0.333]
[0.333   -0.667   0.667]
[0.667   -0.333   -0.667]
Trace = -0.666738987
 angle =   146.446442

Symmetry operator number 7
Product matrix for gA X gB^-1:
[0.667   -0.333   -0.667]
[0.333   -0.667   0.667]
[-0.667   -0.667   -0.333]
Trace = -0.333477974
 angle =   131.815872

Symmetry operator number 8
Product matrix for gA X gB^-1:
[0.667   -0.333   -0.667]
[-0.333   0.667   -0.667]
[0.667   0.667   0.333]
Trace =   1.66695571
 angle =   70.5199966

Symmetry operator number 9
Product matrix for gA X gB^-1:
[0.333   -0.667   0.667]
[0.667   -0.333   -0.667]
[0.667   0.667   0.333]
Trace =   0.333477855
 angle =   109.46682

Symmetry operator number 10
Product matrix for gA X gB^-1:
[-0.333   0.667   -0.667]
[-0.667   0.333   0.667]
[0.667   0.667   0.333]
Trace =   0.333477855
 angle =   109.46682

Symmetry operator number 11
Product matrix for gA X gB^-1:
[-0.333   0.667   -0.667]
[0.667   0.667   0.333]
[0.667   -0.333   -0.667]
Trace = -0.333261013
 angle =   131.807526

Symmetry operator number 12
Product matrix for gA X gB^-1:
[0.667   0.667   0.333]
[0.667   -0.333   -0.667]
[-0.333   0.667   -0.667]
Trace = -0.333261073
 angle =   131.807526

Symmetry operator number 13
Product matrix for gA X gB^-1:
[-0.333   0.667   -0.667]
[-0.667   -0.667   -0.333]
[-0.667   0.333   0.667]
Trace = -0.333261013
 angle =   131.807526

Symmetry operator number 14
Product matrix for gA X gB^-1:
[-0.667   -0.667   -0.333]
[-0.667   0.333   0.667]
[-0.333   0.667   -0.667]
Trace = -1.
 angle =   180.

Symmetry operator number 15
Product matrix for gA X gB^-1:
[0.333   -0.667   0.667]
[-0.667   -0.667   -0.333]
[0.667   -0.333   -0.667]
Trace = -1.
 angle =   180.

Symmetry operator number 16
Product matrix for gA X gB^-1:
[-0.667   -0.667   -0.333]
[0.667   -0.333   -0.667]
[0.333   -0.667   0.667]
Trace = -0.333260953
 angle =   131.807526

Symmetry operator number 17
Product matrix for gA X gB^-1:
[0.333   -0.667   0.667]
[0.667   0.667   0.333]
[-0.667   0.333   0.667]
Trace =   1.66652203
 angle =   70.533165

Symmetry operator number 18
Product matrix for gA X gB^-1:
[0.667   0.667   0.333]
[-0.667   0.333   0.667]
[0.333   -0.667   0.667]
Trace =   1.66652203
 angle =   70.533165

Symmetry operator number 19
Product matrix for gA X gB^-1:
[0.333   -0.667   0.667]
[-0.667   0.333   0.667]
[-0.667   -0.667   -0.333]
Trace =   0.333044171
 angle =   109.480003

Symmetry operator number 20
Product matrix for gA X gB^-1:
[0.667   -0.333   -0.667]
[0.667   0.667   0.333]
[0.333   -0.667   0.667]
Trace =   2.
 angle =   60.

Symmetry operator number 21
Product matrix for gA X gB^-1:
[0.667   0.667   0.333]
[-0.333   0.667   -0.667]
[-0.667   0.333   0.667]
Trace =   2.
 angle =   60.

Symmetry operator number 22
Product matrix for gA X gB^-1:
[0.667   -0.333   -0.667]
[-0.667   -0.667   -0.333]
[-0.333   0.667   -0.667]
Trace = -0.666522205
 angle =   146.435211

Symmetry operator number 23
Product matrix for gA X gB^-1:
[-0.667   -0.667   -0.333]
[-0.333   0.667   -0.667]
[0.667   -0.333   -0.667]
Trace = -0.666522026
 angle =   146.435196

Symmetry operator number 24
Product matrix for gA X gB^-1:
[-0.333   0.667   -0.667]
[0.667   -0.333   -0.667]
[-0.667   -0.667   -0.333]
Trace = -0.999999881
 angle =   179.980209
 MISORI: angle= 60. axis= 1 1

 MISORI: angle= 60. axis= 1 1 -1-1

This set of tables shows each successive result as a different symmetry operator is applied to *Δg*. Note how the angle and the axis varies in each case! Note that #20 is the one that gives a 60° angle.

Misorientations

- Misorientations:

$$\Delta g = g_B g_A^{-1}$$

  transform from crystal axes of grain A back to the reference axes, and then transform *to* the axes of grain B.

- Note that this use of "*g*" is based on the standard Bunge definition (transformation of axes)

- In some texts, misorientation formed from axis transformations is written with a tilde.

$$\Delta \tilde{g}$$

- Standard A->B transformation is expressed in *crystal* axes. The reason for this is that we generally want to know the common axis between the two crystals in terms of crystal coordinates.

Misorientation +Symmetry

- $\Delta g =$
  $(O_c\,g_B)(O_c\,g_A)^{-1}$
  $= O_c g_B g_A^{-1} O_c^{-1}.$

- Note the presence of symmetry operators pre- & post-multiplying

Symmetry: how many equivalent representations of misorientation?

- Axis transformations:
  24 independent operators (for cubic) present on either side of the misorientation. Two equivalents from switching symmetry, i.e. the fact that there is no (physical) difference between passing from grain A to grain B, versus passing from grain B to grain A.

- Number of equivalents = 24x24x2=1152.

When to include Sample Symmetry?

- The rule is simple:
- For calculating orientation distances for the purpose of partitioning orientation space, you *do* include sample symmetry. You only have to apply the sample symmetry, however, to either the component or the cell being tested but not both.
- For calculating misorientations for the purpose of characterizing grain boundaries, you do *not* include sample symmetry.

Practical Help with Volume Fractions

- To calculate volume fractions directly from popLA .SOD files (orientation distributions in popLA format), use sod2vf.f (a Fortran 77 code)

- To calculate volume fractions from a list of discrete orientations in .WTS format, use wts2pop[-latest_revision_date].f, which also bins the orientations into an SOD as well as pole figures and inverse pole figures. Look at any .WTS file to learn about the format (or read the popLA manual). You can find these programs at **neon.materials.cmu.edu/rollett/texture_subroutines**

- If your data source is a *.ANG orientation map (from TSL, or, a *.CTF from HKL) then first use OIM2WTS.f to convert it to the .WTS format. If your material has hexagonal symmetry be very careful about how the Cartesian x-axis is aligned with the crystal axes (TSL and HKL are, typically, different).

Volume fractions from Random?

- Based on a list of 20,000 random orientations and a **15°** acceptance angle, you should expect this set of volume fractions:

- ```
  {001}<100> cube      vol. frac.=      2.175 %
  {001}<110> NDcube   vol. frac.=      2.144 %
  {011}<100> Goss      vol. frac.=      2.310 %
  {110}<112> brass     vol. frac.=      4.116 %
            Dillamore  vol. frac.=      3.030 %
  {211}<111> Copper    vol. frac.=      3.721 %
  {231}<124> S          vol. frac.=      8.475 %
  ```

Volume fractions from Random?

- Based on a list of 20,000 random orientations and a **10°** acceptance angle, you should expect this set of volume fractions:

```
For component cube      vol. frac.=      0.575 %
For component NDcube  vol. frac.=      0.690 %
For component Goss      vol. frac.=      0.715 %
For component brass     vol. frac.=      1.310 %
For component Dillam    vol. frac.=      1.103 %
For component Copper  vol. frac.=      1.382 %
For component 231124  vol. frac.=      2.775 %
```

- Note how the volume fractions have decreased markedly with the decrease in acceptance angle.

- Eliminating the Dillamore component, which is only about 10° from Copper, the following set is found: note that Copper has increased but not by a factor of 2.

```
For component cube      vol. frac.=      0.575 %
For component NDcube  vol. frac.=      0.690 %
For component Goss      vol. frac.=      0.715 %
For component brass     vol. frac.=      1.310 %
For component Copper  vol. frac.=      1.710 %
For component 231124  vol. frac.=      2.775 %
```

More Random/Uniform Volume Fractions by Component

Table 1: Orientations, Multiplicities, and Volume Fractions of 5 Orientation Poles

Component	{hkl}	⟨uvw⟩	rst	(ϕ_1, Φ, ϕ_2)	Multiplicity	5° Code A	1° Code A	5° Code B	1° Code B
Cube	001	100	010	$(0,0,0)$	4	0.65%	0.68%	0.54%	0.67%
Goss	011	100	01$\bar{1}$	$(0,45,0)$	4	0.64%	0.67%	0.59%	0.67%
Brass	101	1$\bar{2}$1	10$\bar{1}$	$(35.26, 45, 90)$	2	1.19%	1.35%	1.19%	1.35%
$\eta - \text{fiber}_1$	012	100	02$\bar{1}$	$(0, 26.57, 0)$	2	1.34%	1.35%	1.34%	1.35%
S	213	3$\bar{6}$4	22 17 9	$(58.98, 36.70, 63.43)$	1	2.76%	2.70%	2.78%	2.70%

Taken from unpublished work by Creuziger, Hu & Rollett (2010).

Variations in Random V_f

- Why do the volume fractions vary with component location?

- Answer: mainly because of variations in how close they lie to symmetry planes in orientation space.

- Assume cubic-orthorhombic (crystal+sample) symmetry

- An orientation such as Goss lies on one edge, so despite including 3 symmetry-related locations, its volume is only about 1/4th of, say, the S component.

- Similarly the Copper component, only includes 1/2th of the space of the S component.

- The rest of the variation is related to location with respect to the second Euler angle. See the next slide for illustrations of the above points.

3D Views

a) Brass   b) Copper   c) S   d)
Goss   e) Cube
f) combined texture
1: {35, 45, 90}, Brass,
2: {55, 90, 45}, Brass
3: {90, 35, 45}, Copper,
4: {39, 66, 27}, Copper
5: {59, 37, 63}, S,
6: {27, 58, 18}, S,
7: {53, 75, 34}, S
8: {90, 90, 45}, Goss
9: {0, 0, 0}, cube*
10: {45, 0, 0}, rotated cube

* Note that the cube exists as a line between (0,0,90) and (90,0,0) because of the linear dependence of the 1st and 3rd angles when the 2nd angle = 0.
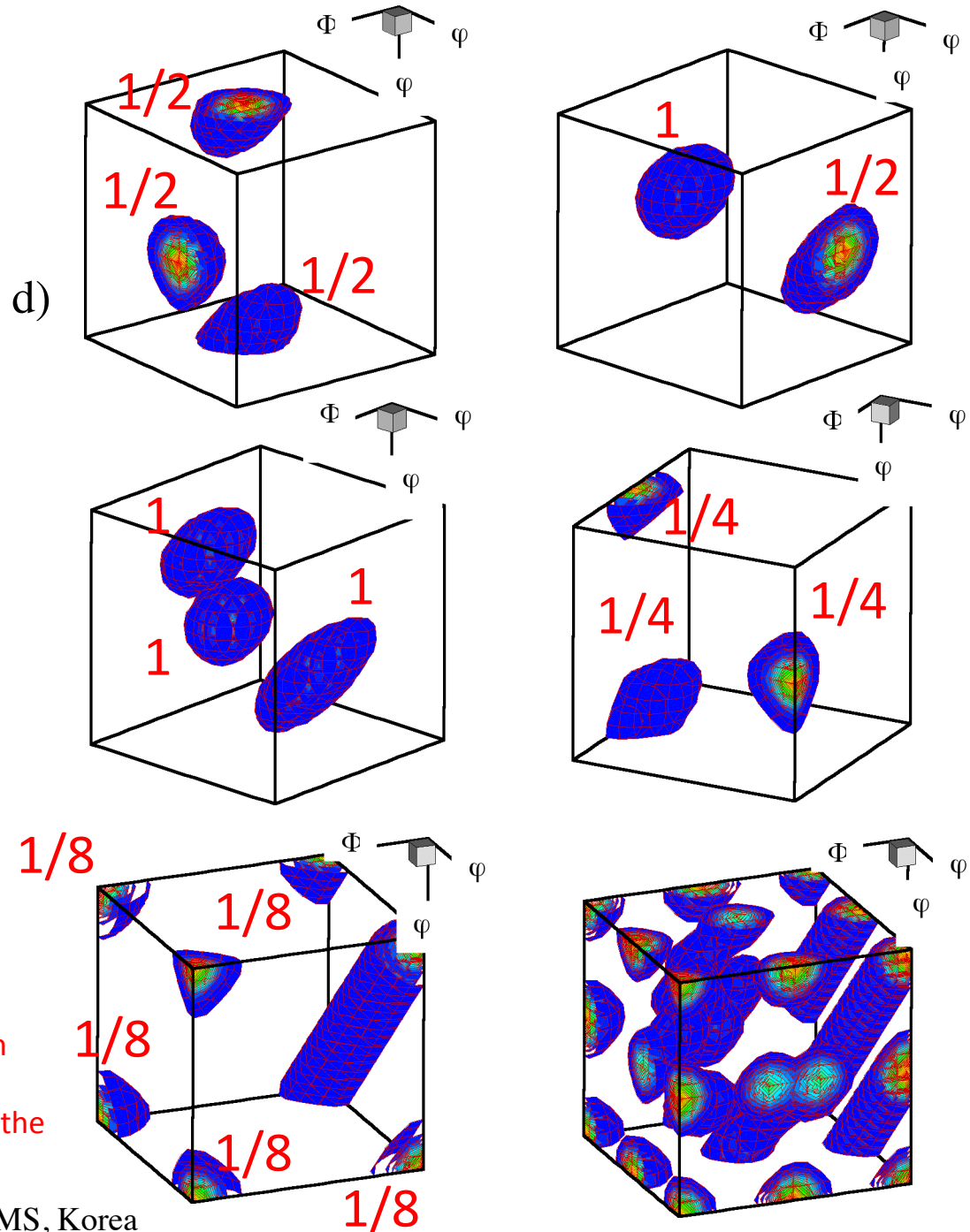
Figure courtesy of Jae-hyung Cho, KIMS, Korea

Scaling by Random V_f

- It has been argued that volume fractions are more reliable than intensities partly because they reflect the physical makeup of the material more accurately. For example, the intensity at the cube position rises to very high values once the volume fraction of orientations near cube rises much above 25%, which is not true of other orientations.

- Given that the volume fraction varies significantly with position in the space, especially for components near symmetry planes, it has also been argued that volume fractions should be reported as a multiple of the fraction associated with a random (uniform) texture.

Summary

- Methods for calculating volume fractions from discrete orientation distributions reviewed.

- Complementary method of calculating the OD from information on discrete orientations (e.g. OIM) provided.

- Method for calculating *orientation distance* (equivalent to *misorientation*) given, with illustrations of the importance of how to apply symmetry operators.

- For further discussion: in some cases, it is useful to compare volume fractions in a textured material to the volume fractions that would be expected in a randomly oriented material.

- Different programs may well yield different volume fraction values because of differences in the procedure (e.g. how the space is partitioned).

Supplemental Slides

- The following slides illustrate what happens with misorientations if you deal with *active rotations*, instead of the standard *axis transformations* (passive rotations) used in materials science.

- This material is useful in case you have experience with solid mechanics, or you cannot get a misorientation calculation to work properly.

- Note: it does *not* matter whether you use passive or active rotations for computing the rotation angle; it only makes a difference to the rotation axis, i.e. the skew-symmetric part of the misorientation matrix.

Passive vs. Active Rotations

These next few slides describe the differences between dealing with passive rotations (= transformations of axes) and active rotations (fixed coordinate system)

- Passive Rotations
- Materials Science
- g describes an *axis transformation* from sample to crystal axes

- Active Rotations
- Solid mechanics
- g describes a *rotation* of a crystal from ref. position to its orientation.

Passive Rotations (Axis Transformations)   Active (Vector) Rotations

Matrices

$$\boldsymbol{g = Z_2 X Z_1} =$$

$$\begin{pmatrix} \cos\varphi_1\cos\varphi_2 & \sin\varphi_1\cos\varphi_2 & \sin\varphi_2\sin\Phi \\ -\sin\varphi_1\sin\varphi_2\cos\Phi & +\cos\varphi_1\sin\varphi_2\cos\Phi & \\ & & \\ -\cos\varphi_1\sin\varphi_2 & -\sin\varphi_1\sin\varphi_2 & \cos\varphi_2\sin\Phi \\ -\sin\varphi_1\cos\varphi_2\cos\Phi & +\cos\varphi_1\cos\varphi_2\cos\Phi & \\ & & \\ \sin\varphi_1\sin\Phi & -\cos\varphi_1\sin\Phi & \cos\Phi \end{pmatrix}$$

Note transpose relationship between the two matrices.

$$\boldsymbol{g = g_{\phi_{1001}} g_{\Phi_{100}} g_{\phi_{2001}}} =$$

$$\begin{pmatrix} \cos\varphi_1\cos\varphi_2 & -\cos\varphi_1\sin\varphi_2 & \sin\varphi_1\sin\Phi \\ -\sin\varphi_1\sin\varphi_2\cos\Phi & -\sin\varphi_1\cos\varphi_2\cos\Phi & \\ & & \\ \sin\varphi_1\cos\varphi_2 & -\sin\varphi_1\sin\varphi_2 & -\cos\varphi_1\sin\Phi \\ +\cos\varphi_1\sin\varphi_2\cos\Phi & +\cos\varphi_1\cos\varphi_2\cos\Phi & \\ & & \\ \sin\varphi_2\sin\Phi & \cos\varphi_2\sin\Phi & \cos\Phi \end{pmatrix}$$

Passive Rotations (Axis Transformations)   Active (Vector) Rotations

Texture +Symmetry

Symmetry Operators:

$$O_{sample} \equiv O_s$$

$$O_{crystal} \equiv O_c$$

$$g' = O_c g O_s$$

$$g' \xleftarrow{=}\longrightarrow g$$

Note that the crystal symmetry post-multiplies, and the sample symmetry pre-multiplies.

$$g' = O_s g O_c$$

$$g' \xleftarrow{=}\longrightarrow g$$

Note the reversal in order of application of symmetry operators!

Passive Rotations (Axis Transformations)   Active (Vector) Rotations

Groups: Sample +Crystal Symmetry

- $O_c \in O(432)$; proper rotations of the cubic point group.

- $O_s \in O(222)$; proper rotations of the orthorhombic point group.

- Think of applying the symmetry operator in the appropriate frame: thus for active rotations, apply symmetry to the crystal *before* you rotate it.

Passive Rotations (Axis Transformations)

Active (Vector) Rotations

Misorientations

- Misorientations:

  $$\Delta g = g_B g_A^{-1};$$

  transform from crystal axes of grain A back to the reference axes, and then transform *to* the axes of grain B.

- Note that this use of "*g*" is based on the standard Bunge definition (transformation of axes)

- Misorientations:

  $$\Delta g = g_B g_A^{-1};$$

  the net rotation *from* A *to* B is: rotate first *back from* the position of grain A and then rotate *to* the position of grain B.

- Note that this use of "*g*" is based on the a definition in terms of an active rotation (the "*g*" is the inverse, or transpose of the one on the left).

Passive Rotations (Axis Transformations)   Active (Vector) Rotations

Notation

- In some texts, misorientation formed from axis transformations is written with a tilde.

$$\Delta \tilde{g}$$

- Standard A->B transformation is expressed in *crystal* axes.

- You must verify from the context which type of misorientation is discussed in a text!

- Standard A->B rotation is expressed in *sample* axes.

Misorientation +Symmetry

- $\Delta g =$
  $(O_c\, g_B)(O_c\, g_A)^{-1}$
  $= O_c g_B g_A^{-1} O_c^{-1}.$

- Note the presence of symmetry operators pre- & post-multiplying

- $\Delta g = g_B g_A^{-1};$
  $(g_B O_c)(g_A O_c)^{-1}$
  $= g_B O_c O_c^{-1} g_A^{-1}$
  $= g_B O_c' g_A^{-1}.$

- Note the reduction to a single symmetry operator because the symmetry operators belong to the same group!

Symmetry: how many equivalent representations of misorientation?

- Axis transformations: 24 independent operators (for cubic) present on either side of the misorientation. Two equivalents from switching symmetry.

- Number of equivalents= 24x24x2=1152.

- Active rotations: Only 24 independent operators present "inside" the misorientation. 2 from switching symmetry.

- Number of equivalents= 24x2=48.

Passive Rotations (Axis Transformations)   Active (Vector) Rotations

Passive ↔ *Active*

Just as is the case for rotations, and texture components,

$$g_{passive}(\theta, \boldsymbol{n}) = g^{T}_{active}(\theta, \boldsymbol{n}),$$

so too for misorientations,

$$\Delta g_{passive}(\theta, \boldsymbol{n}) = \Delta g^{T}_{active}(\theta, \boldsymbol{n}).$$

However, please be careful about the frame. The discussion given here (with the exception of the example that illustrated how the misorientation axis moved with the bi-crystal) is based on using the "local" or "crystal" frame, *not* the reference frame.

The relationship between the misorientation calculated in the local frame and the misorientation calculated in the reference frame is not at all simple. For dealing with grain boundaries, I strongly suggest that you stick to the local/crystal frame.

Worked example: active rotations

- So what happens when we express orientations as active rotations in the sample reference frame?

- The result is similar (same minimum rotation angle) but the axis is different!

- The rotation axis is the sample [100] axis, or *x-axis,* which happens to be parallel to a *crystal* <111> direction because the Copper component is (112)[11-1].

{100} pole figures
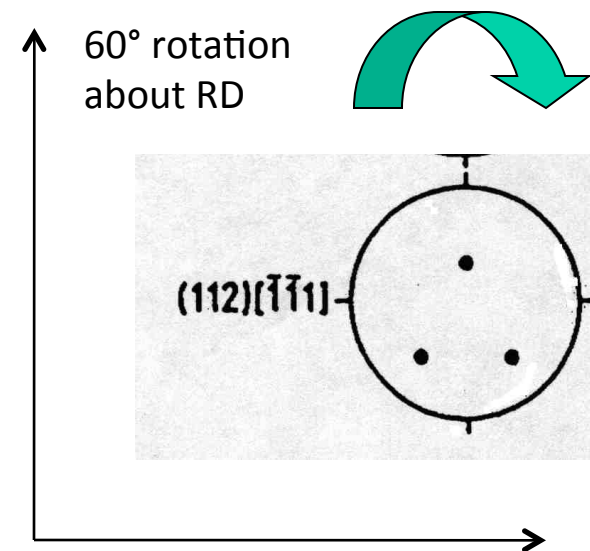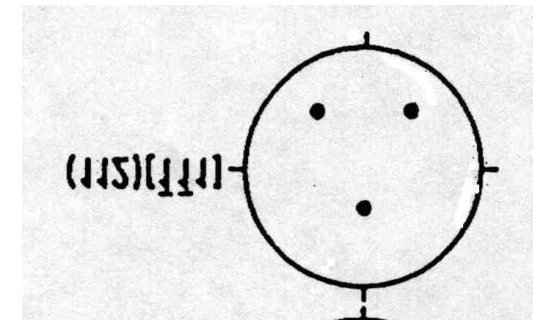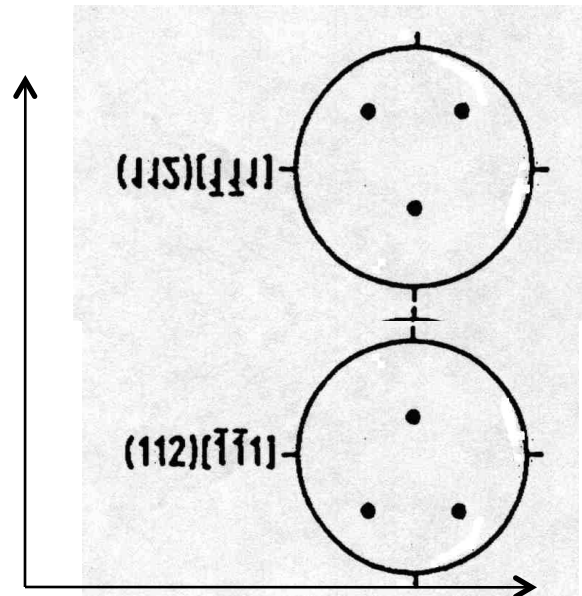
60° rotation about RD

Active rotations example

angles..   90.   35.2599983   45.
angles..   270.   35.2599983   45.

**1st Grain: Euler angles: 90.   35.2599983
   45.**
**2nd Grain: Euler angles: 270.
   35.2599983   45.**

 1st matrix:
[-0.577   0.707   0.408]
[-0.577   -0.707   0.408]
[0.577   0.000   0.817]

 2nd matrix:
[0.577   -0.707   0.408]
[0.577   0.707   0.408]
[-0.577   0.000   0.817]

MISORInv: angle=   60. axis= 1 0 0

Symmetry operator number 1
Product matrix for gB X gA^-1:
[-1.000   0.000   0.000]
[0.000   -1.000   0.000]
[0.000   0.000   1.000]
 Trace = -1.
  angle =   180.

 Symmetry operator number 2
Product matrix for gB X gA^-1:
[-0.333   0.000   0.943]
[0.816   -0.500   0.289]
[0.471   0.866   0.167]
 Trace = -0.666738927
  angle =   146.446442

 Symmetry operator number 3
Product matrix for gB X gA^-1:
[0.333   0.817   0.471]
[0.817   0.000   -0.577]
[-0.471   0.577   -0.667]
 Trace = -0.333477914
  angle =   131.815872
....

Note: same angle, different axis, now in sample frame

Active rotations

- What is stranger, at first sight, is that, as you rotate the two orientations together in the sample frame, the misorientation axis moves with them, if expressed in the reference frame (active rotations).

- On the other hand, if one uses passive rotations, so that the result is in crystal coordinates, then the misorientation axis remains *un*changed, as you rotate the pair of cr.

Active rotations example

(115)[[1̄1̄1]

(112)[1̄1̄1]

Add 10° to the first Euler angle so that both crystals move together:

angles..   100.   35.2599983   45.
angles..   280.   35.2599983   45.

1st matrix:
```
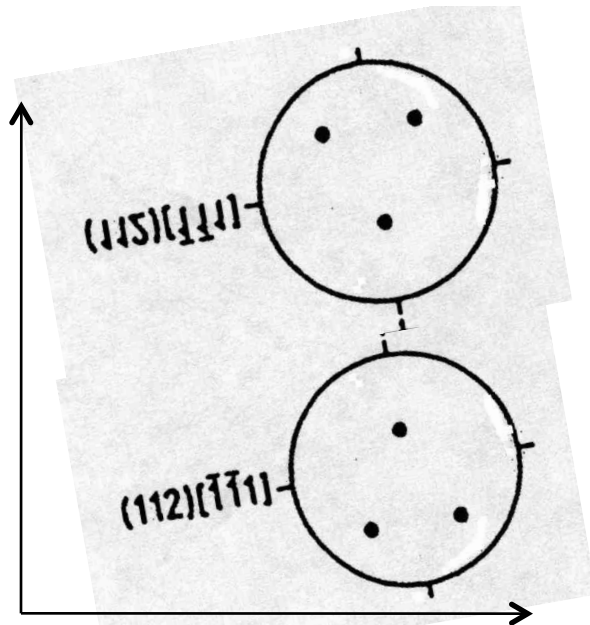[    -0.691      0.596      0.408 ]
[    -0.446     -0.797      0.408 ]
[     0.569      0.100      0.817 ]
```

2nd matrix:
```
[     0.691     -0.596      0.408 ]
[     0.446      0.797      0.408 ]
[    -0.569     -0.100      0.817 ]
```

MISORInv: angle=   60. axis= 6 1 0

Note the change in the misorientation axis from 100 to 610!

Symmetry operator number 1
Product matrix for gB X gA^-1:
```
[    -1.000      0.000      0.000 ]
[     0.000     -1.000      0.000 ]
[     0.000      0.000      1.000 ]
```
Trace = -1.
 angle =   180.

 Symmetry operator number 2
Product matrix for gB X gA^-1:
```
[    -0.478      0.004      0.878 ]
[     0.820     -0.355      0.448 ]
[     0.314      0.935      0.167 ]
```
Trace = -0.666738808
 angle =   146.446426

 Symmetry operator number 3
Product matrix for gB X gA^-1:
```
[     0.044      0.824      0.564 ]
[     0.824      0.289     -0.487 ]
[    -0.564      0.487     -0.667 ]
```
Trace = -0.333477765
 angle =   131.815857
…………..