PolyLX – the MATLAB ™ toolbox for quantitative microstructural analyses

LEXA Ondrej
Institute of Petrology and Structural Geology
Charles University, Prague, CZ

Why new toolbox ?
There are already tools available...

- YES... but....
- We need tool which allowed us explore data and patterns in very different ways
- We need tool which is easily extendible to meet our desires...

Image analysis tools

- NIH Image – one of the best freeware (MAC)
- ImageJ software – multiplatform
- Plenty of commercial software e.g. Image-Pro Plus, OPTIMAS (MediaCybernetics)

Automated processing of grabbed images

- can markedly speed-up analyses

- quick, but can easily lead to erroneous results

- automatically vectorized objects are not always suitable for analysis that we need

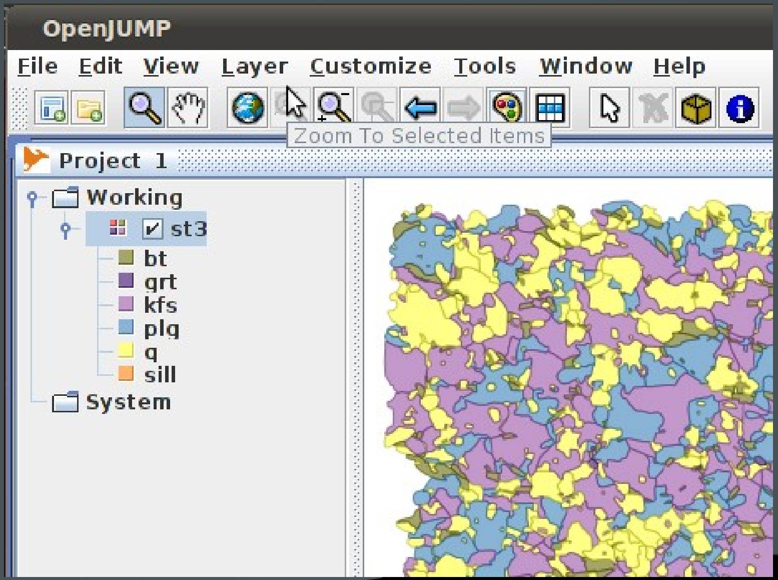
Semi-automated digitizing

- Manual digitizing of outlines
- Possibility to mosaic images
- Easy scale calibration
- Automatically controlled topology of grains
  - unavoidable for correct extraction of boundaries

• we use a desktop GIS - ESRI ArcGIS (commercial), QGIS or OpenJUMP (free)

• easy to do mosaic and calibration of images to be vectorized

• Various strategies to digitize microstructure

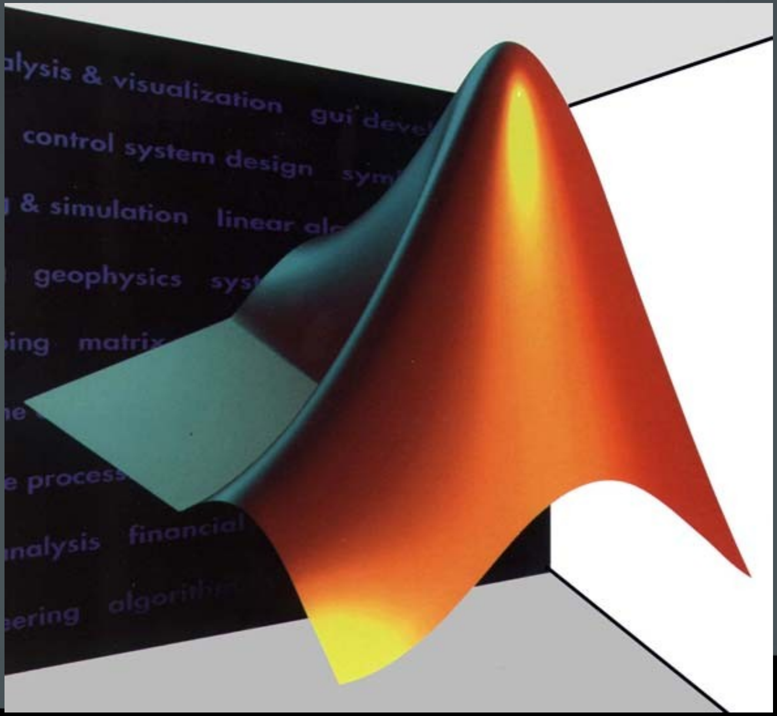


Why we use MATLAB ?

- Simple scripting language
- Good visualization and graph annotation
- Many ready-to-use statistical and numerical routines
- Accessible on many academic institutions
- Lot of existing literature
- Easily extensible

… but MATLAB is not cheap

and not free either.



What is PolyLX exactly ?

Object-oriented MATLAB™ toolbox i.e. set of routines that covers:

- Data exchange
  - input from various formats (Shapefiles, JML, SXM, Elle, DXF)
- Flexible data manipulation
  - Queries (data based or topology based)
- Visualization of microstructural data
  - Predefined graphs, easy to develop own one...
- Analysis of microstructural data
  - Predefined and actively developed routines, easily

What analysis could be done with PolyLX?

Various techniques to describe:

- grain and grain boundary shape
- grain and grain boundary preferred orientation
- spatial statistics of grains and grain boundaries
- strain analysis
- and more...

Object-oriented model

Microstructure

Grains
(polygons)
`polylxgrain`

Boundaries
(polylines)
`polylxboundary`

ID
Phase
X,Y coordinates
Centroid
[Out]Area
[Out]Perimeter
Length
LogLength
Width
Orientation

AxialRatio
LogAxialRatio
[Out]Elongation
[Out]Ferret
[Out]Roundness
[Out]Circularity
[Out]Ellipticity
[Out]Compactness
[Out]GSI
[Out]GSF

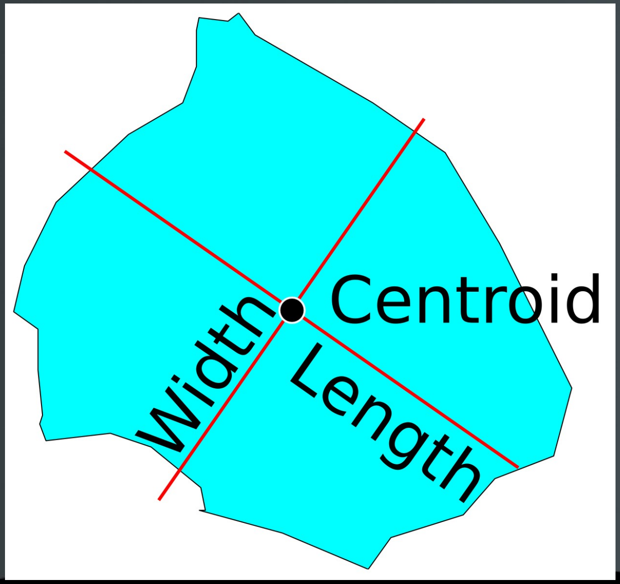
ID
ID’s of grains
Phase’s of grains
Type
X,Y coordinates
Centroid

CumLength
Length
Width
Orientation
AxialRatio
Straightness

Properties are calculated when object is created

Object geometry

- Length and Width grain properties are calculated as projection of grain on principal axes of area moments of inertia
  - Works well with angular shapes



Grain and grain boundary shape

Shape is extremely difficult property to measure, or even to define in a precise manner. Perhaps this is why there are so many proposed shape measures, none of which has been proved as entirely satisfactory.

A shape measure should possess several desirable properties. Obviously, objects with different shapes should yield different measures, and similarly shapes should yield similar values regardless of the size or orientation of the object. Unfortunately, a shape measure possessing these properties may be a chimera; it has been proven that no single measure can be unique to only one shape.

Therefore, there is a wide spectrum of single value measuring methods available in PolyLX toolbox.

Grain and grain boundary shape

AxialRatio	$AR = \frac{L}{W}$
Elongation	$E = \frac{\pi L^2}{4A}$
Ferret diameter	$F = 2\sqrt{\frac{A}{\pi}}$
Roundness	$R = \frac{4A}{\pi L^2} = \frac{1}{E}$
Circularity	$C = \frac{4A}{PL}$
Ellipticity	$El = \frac{\pi L^2}{2A}$
Compactness	$Cp = \frac{P^2}{4\pi A}$
Grain shape index	$GSI = \frac{2\pi\sqrt{A/\pi}}{L}$
Grain shape factor	$GSF = \left(L/W\right)^{0.318} \cdot \frac{P}{2\sqrt{A}}$
Straightness	$S = \frac{Lt}{L}$

Grain and grain boundary preferred orientation

Several techniques are implemented in PolyLX toolbox

- second moments based ellipse fitting
- coordinate covariance matrix

  This method can be applied on individual grains or boundaries as well as on a set of grains or grain boundaries. The result in letter case is weighted by size of objects, which is welcome in specific tasks and differs from the results obtained from orientation analysis based on histograms/rose diagrams or Fisher distribution.

- mean matrix of inertia (Harvey & Ferguson (1981))
- routines using approach of direction dependent projection of grain or grain boundaries, like PAROR, SURFOR and PARIS of Pannozo (1983) and fully implemented.

Moments based ellipse fitting

$$a = \iint\limits_R 1\, dx\, dy$$

zero moment i.e. area

$$\begin{aligned} \alpha_x &= \frac{1}{a}\iint\limits_R x\, dx\, dy \\ \alpha_y &= \frac{1}{a}\iint\limits_R y\, dx\, dy \end{aligned}$$

first moments ie. centroid of area

Second moments

$$\begin{aligned} \alpha_{xx} &= \frac{1}{a}\iint\limits_R x^2\, dx\, dy \\ \alpha_{xy} &= \frac{1}{a}\iint\limits_R xy\, dx\, dy \\ \alpha_{yy} &= \frac{1}{a}\iint\limits_R y^2\, dx\, dy \end{aligned}$$

Centralized second order moments

$$\begin{aligned} \mu_{xx} &= \alpha_{xx} - \alpha_x^2 \\ \mu_{xy} &= \alpha_{xy} - \alpha_x\alpha_y \\ \mu_{yy} &= \alpha_{yy} - \alpha_y^2 \ . \end{aligned}$$

We can obtain parameters of ellipse by calculating eigenvalues and eigenvectors of matrix

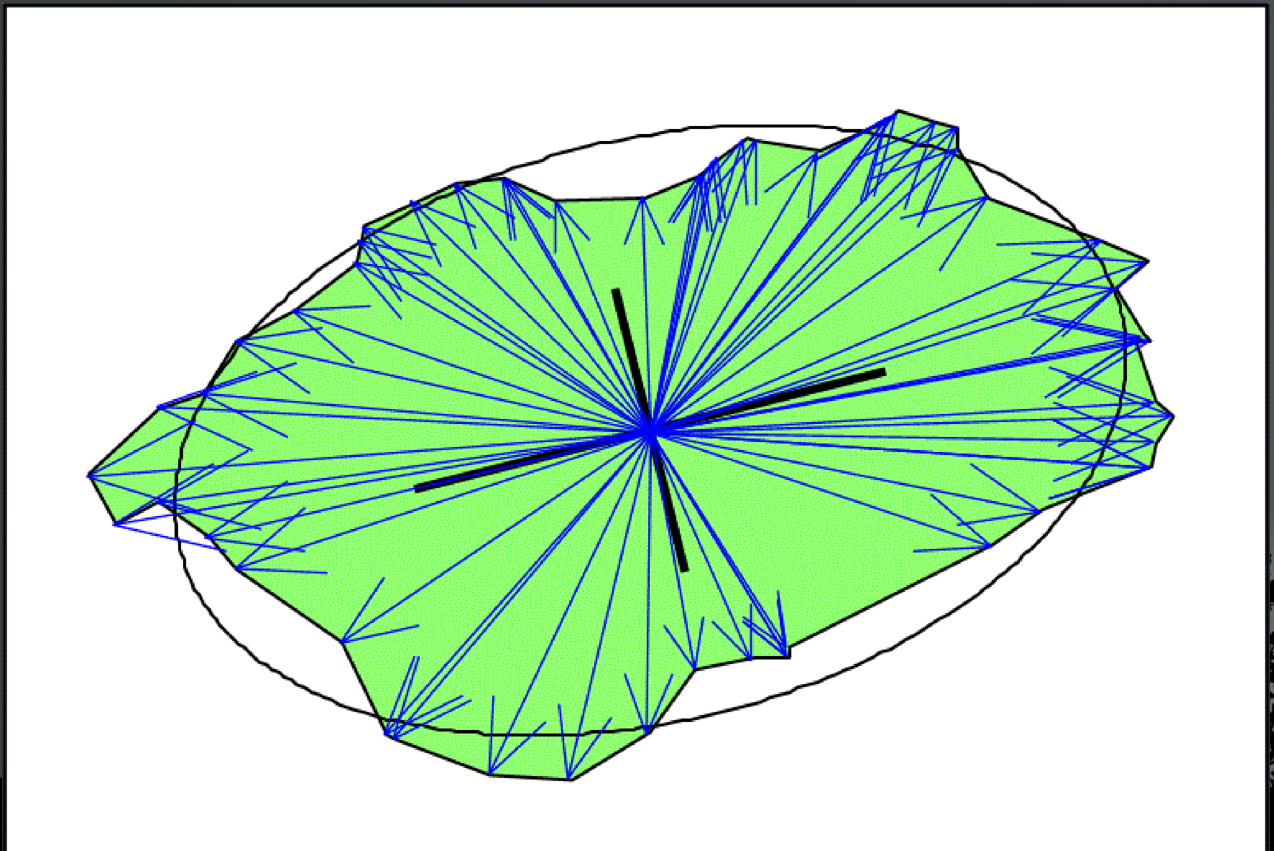
$$\frac{1}{4(\mu_{xx}\mu_{yy} - \mu_{xy}^2)} \begin{pmatrix} \mu_{yy} & -\mu_{xy} \\ -\mu_{xy} & \mu_{xx} \end{pmatrix}$$

To obtain these integrals, Green's theorem is applied to reduce them to curve integrals over boundary represented as piecewise linear function.

Coordinate covariance matrix

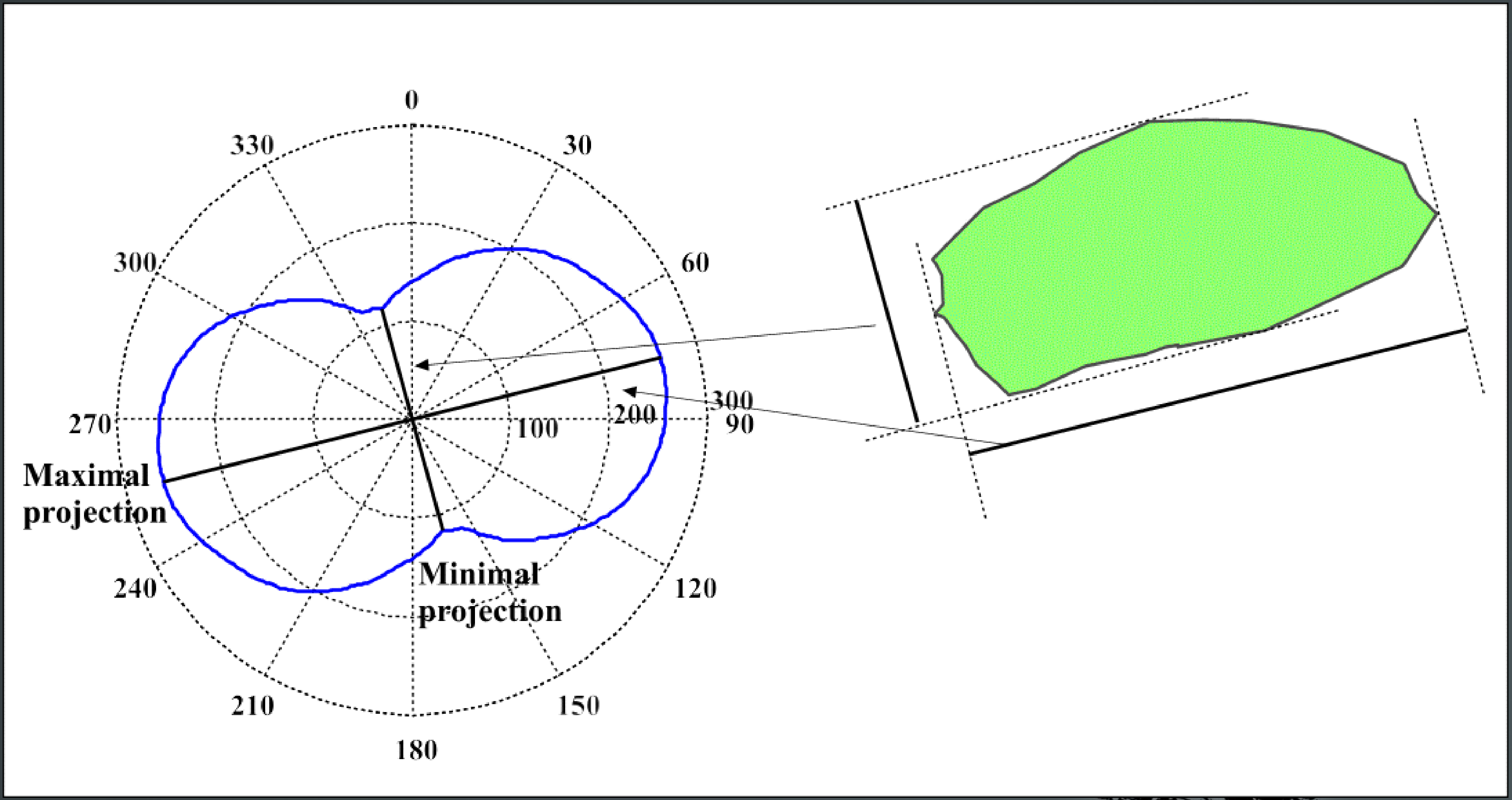
$$\mathbf{O} = \frac{1}{N}\begin{bmatrix} \sum dx^2 & \sum dxdy \\ \sum dxdy & \sum dy^2 \end{bmatrix}$$

$$dx = x_i - x_c$$
$$dy = y_i - y_c$$



Projection based methods

PAROR, SURFOR and PARIS



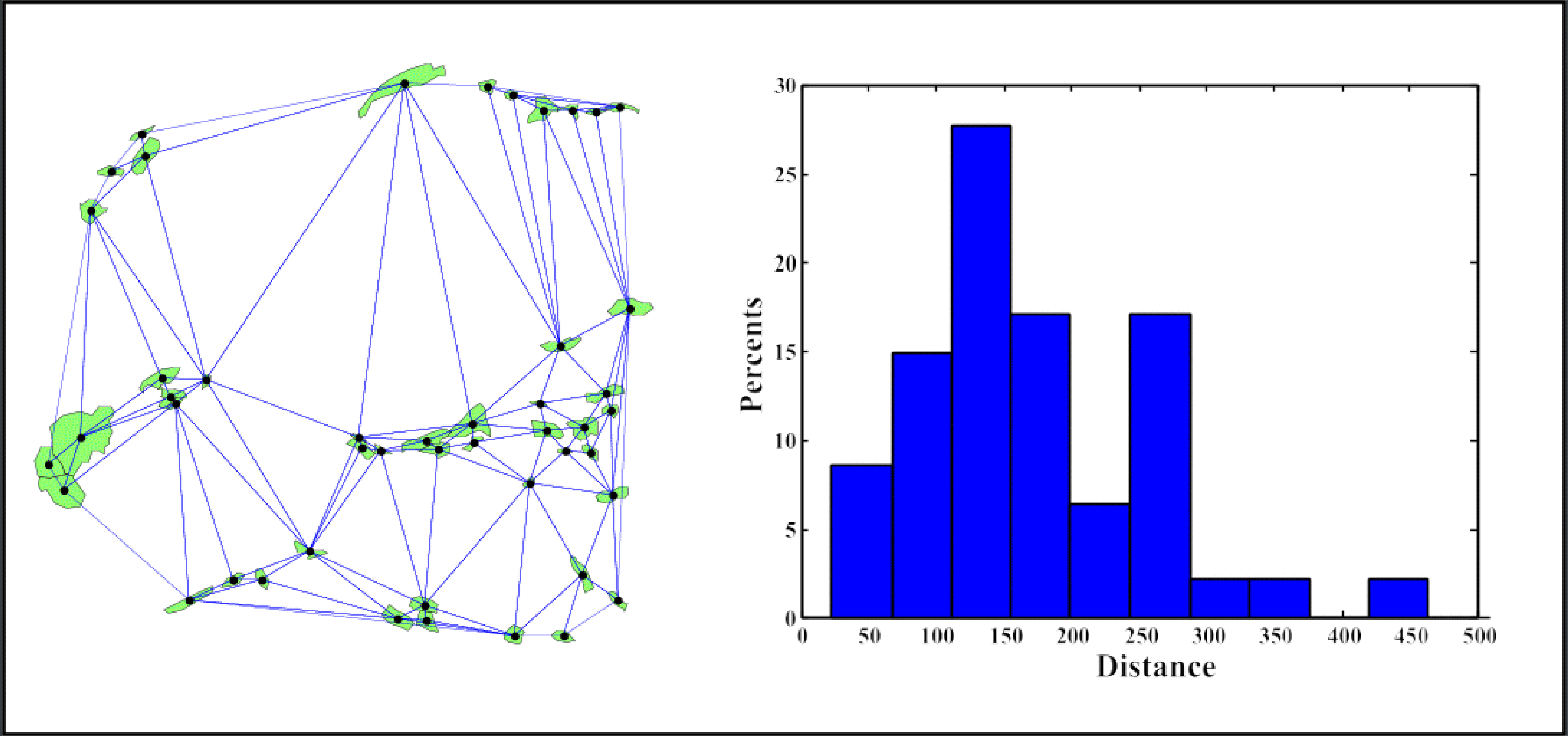
Spatial statistics

- grain density method - test of homogeneity is calculated
- nearest neighbour analysis (NNA – spatial pattern index R)
- phase connectivity
- Evaluation of deviation from random/expected distribution (Kretz, 1969)
    - Contact frequency method
    - Contact length method
    - Contact average method

Nearest neighbour analysis

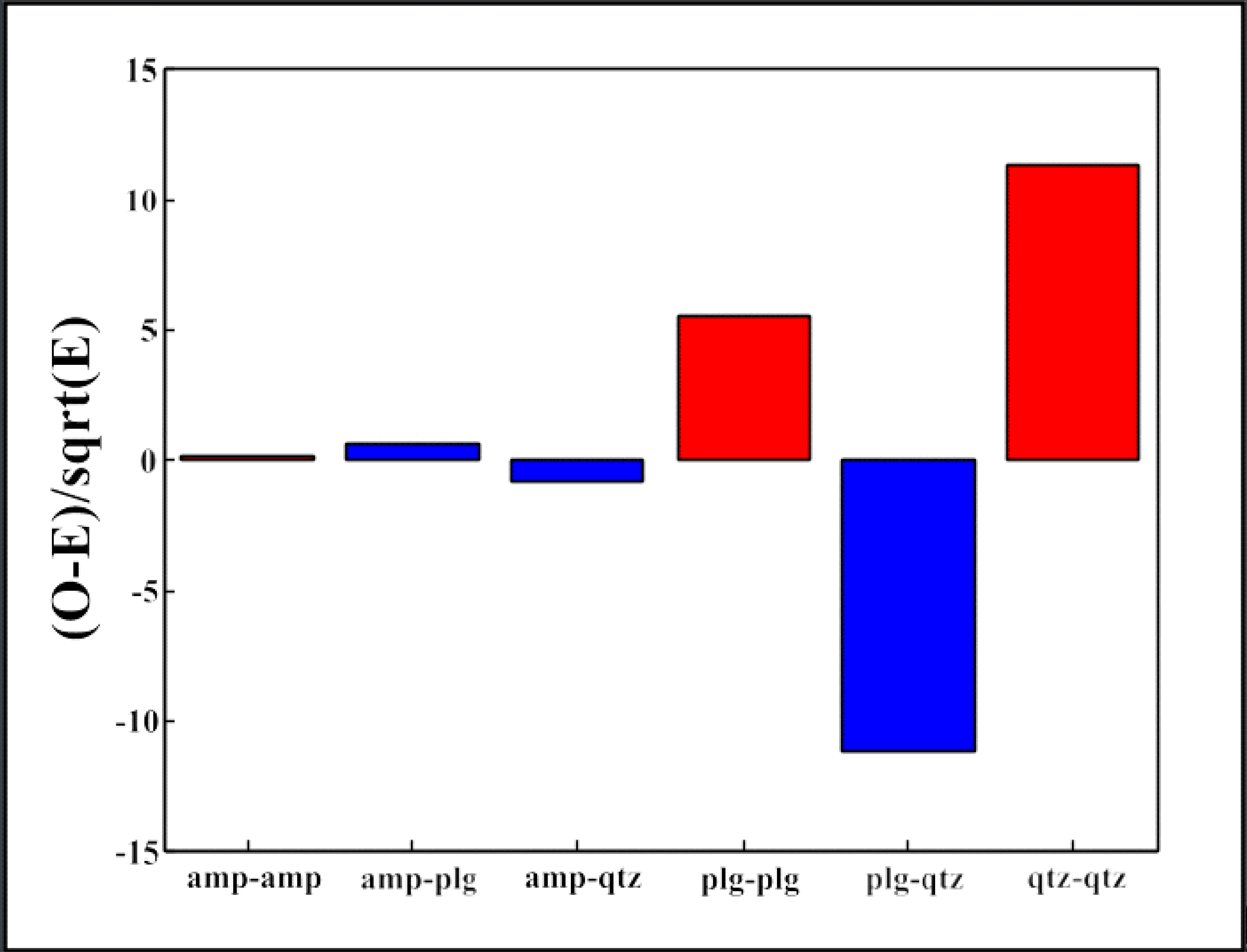
R - nearest-neighbour statistics index

0 - clustered, 1 - random, 2.149 regular



R index: 0.81614 Clustered with standard variate: 2.4114

Contact frequency method



Phase connectivity

Method adopted from graph theory (Zhang et. al. 1992)

Bulk connectivity of phase is defined as sum of *k*-order connectivity ratios, calculated for each degree of phase partition (degree of partition is equal to number of grain in it).

$$C_k = \sum \frac{b_k}{(b_0 + b_C)}$$

$$C = \sum_{k=2}^{N} C_k$$

where C_k is *k*-order connectivity

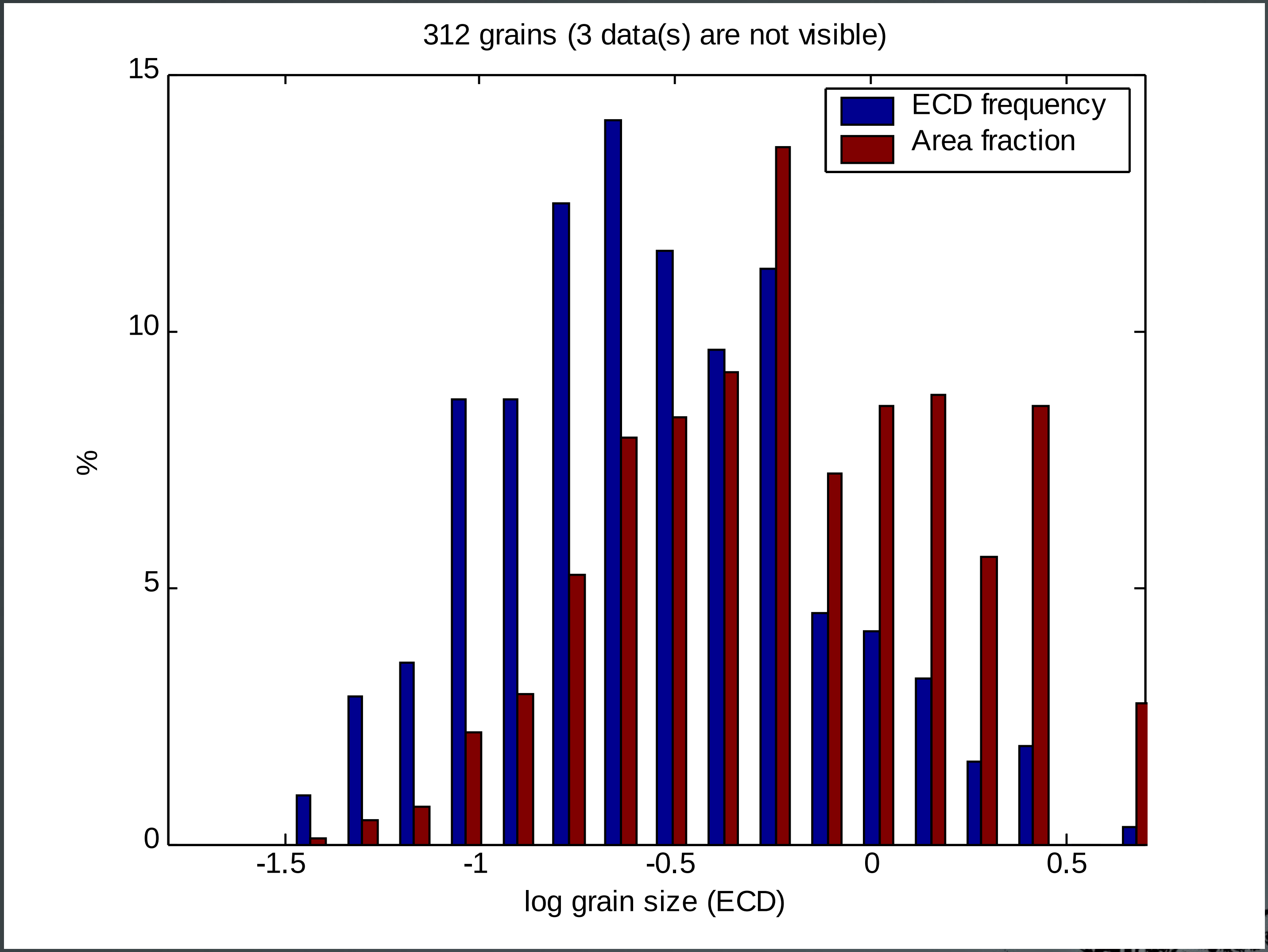
b_k is number of grains in partitions with degree *k*

b_0 is number of isolated grains a b_C is number of connected grains

'Phase'	'N'	'iso'	'Connectivity'
'ac'	[31]	[24]	[0.2258]
'amp'	[534]	[187]	[0.6498]
'grt'	[9]	[9]	[0]
'plg'	[1043]	[39]	[0.9626]
'qtz'	[1014]	[13]	[0.9872]

C for each phase between 0 (all grains are separated) and 1 (all grains are in contact).

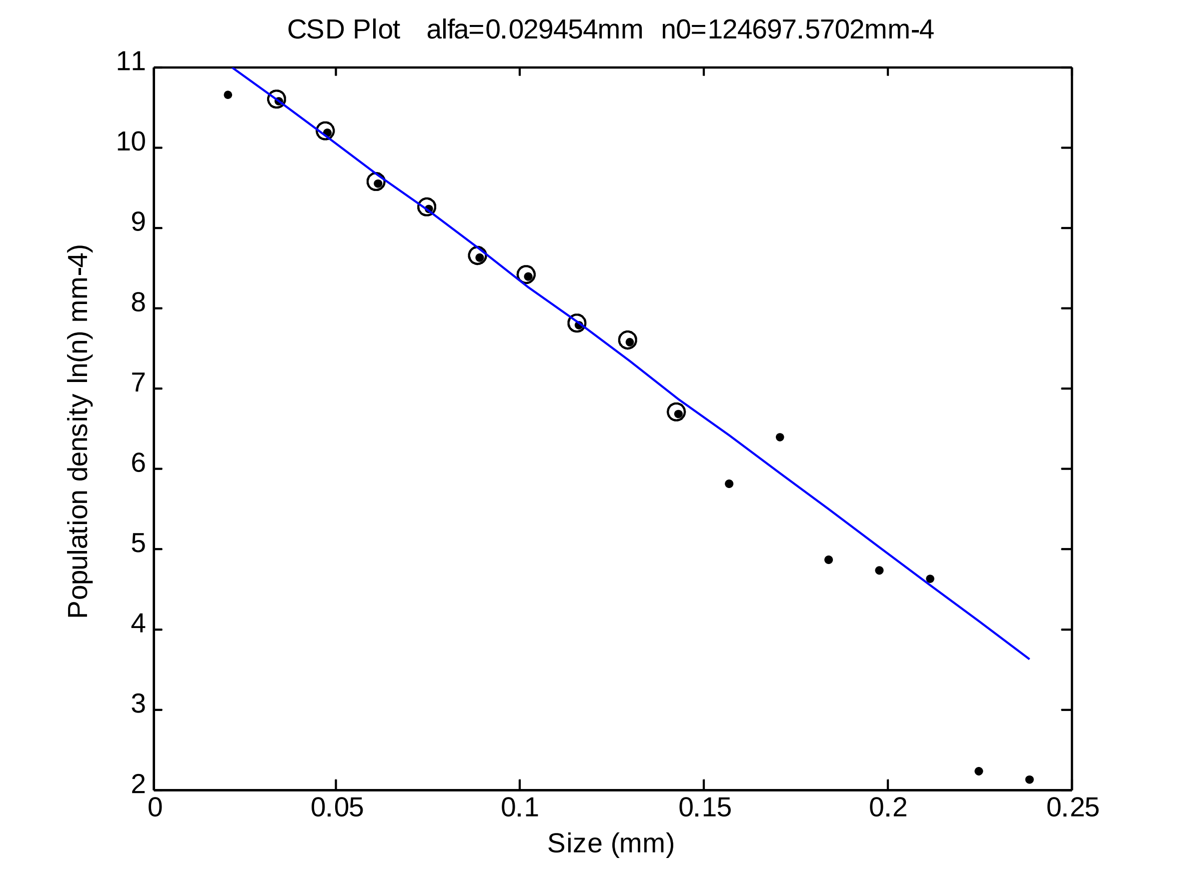
Crystal size



To plot 3D CSD curves method of Peterson, 1996 is implemented.

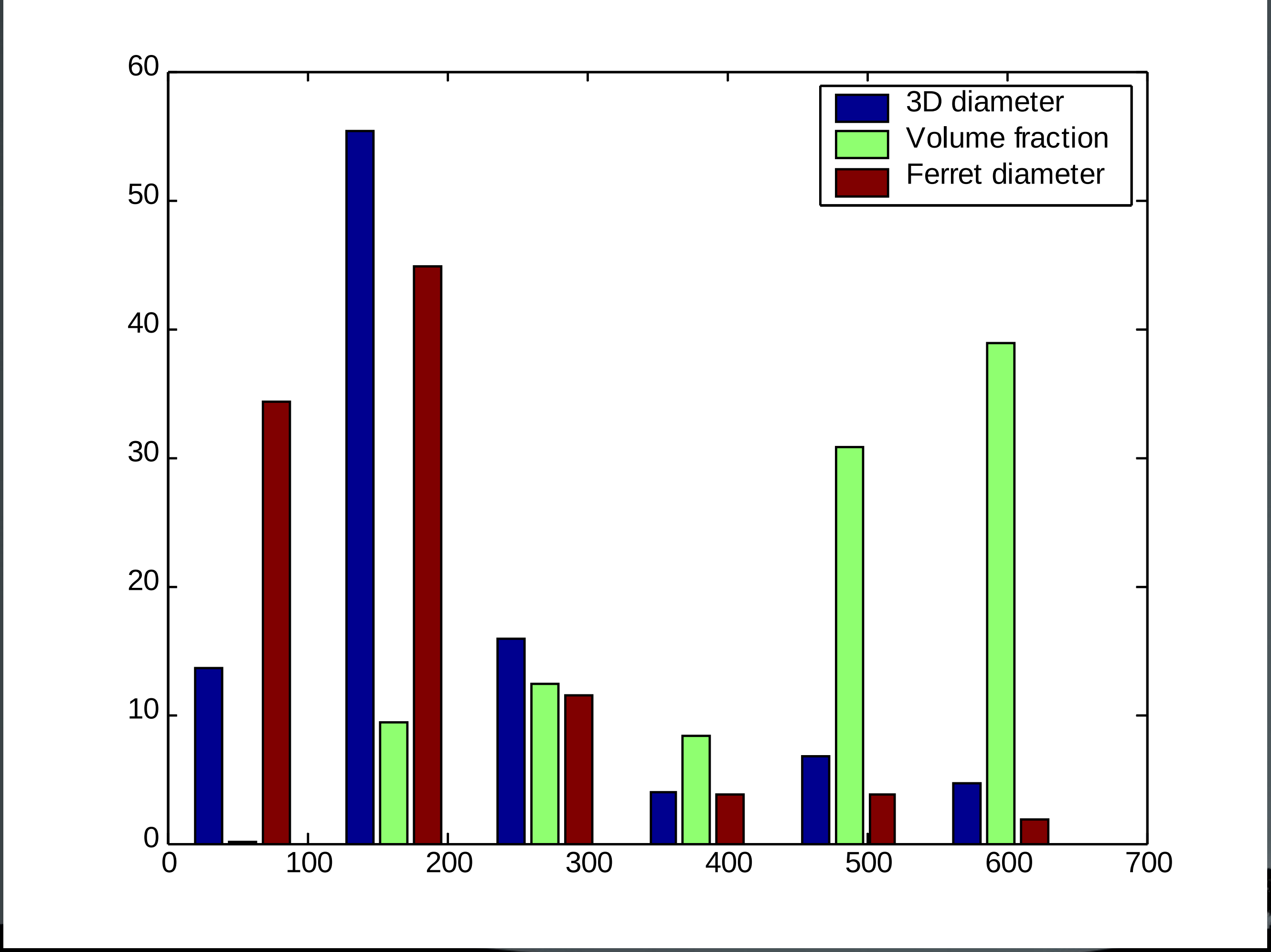
3 factors (experimentally derived) are used to correct raw data, dependent on crystal symmetry and degree of orientation of texture.

Shape	S	gamma
Sphere	1.269	0.833
Cube (as X11 plate)	0.847	0.515
Dodecahedron	1.425	1.004
Octahedron	1.501	1.139
Tetrahedron	1.022	0.812
Plate 2,1,1	0.927	0.795
Plate 10,1,1	1.063	1.128
Trachytic plate 2,1,1	1.019	0.696
Trachytic plate 10,1,1	1.100	0.780
Cube (as 111 prism)	1.151	0.662
Prism 2,2,1	1.098	0.966
Prism 3,3,1	1.107	1.083
Prism 10,10,1	1.062	1.479
Lineated prism X11	1.106	0.690
Trachytic slab 4,2,1	1.172	0.717
Trachytic slab 6,2,1	1.145	0.724
Trachytic slab 6,3,1	1.136	0.833
Trachytic slab 9,3,1	1.136	0.802



Conversion from 2D to 3D

Based on sphere sections distribution (aka StripStar)

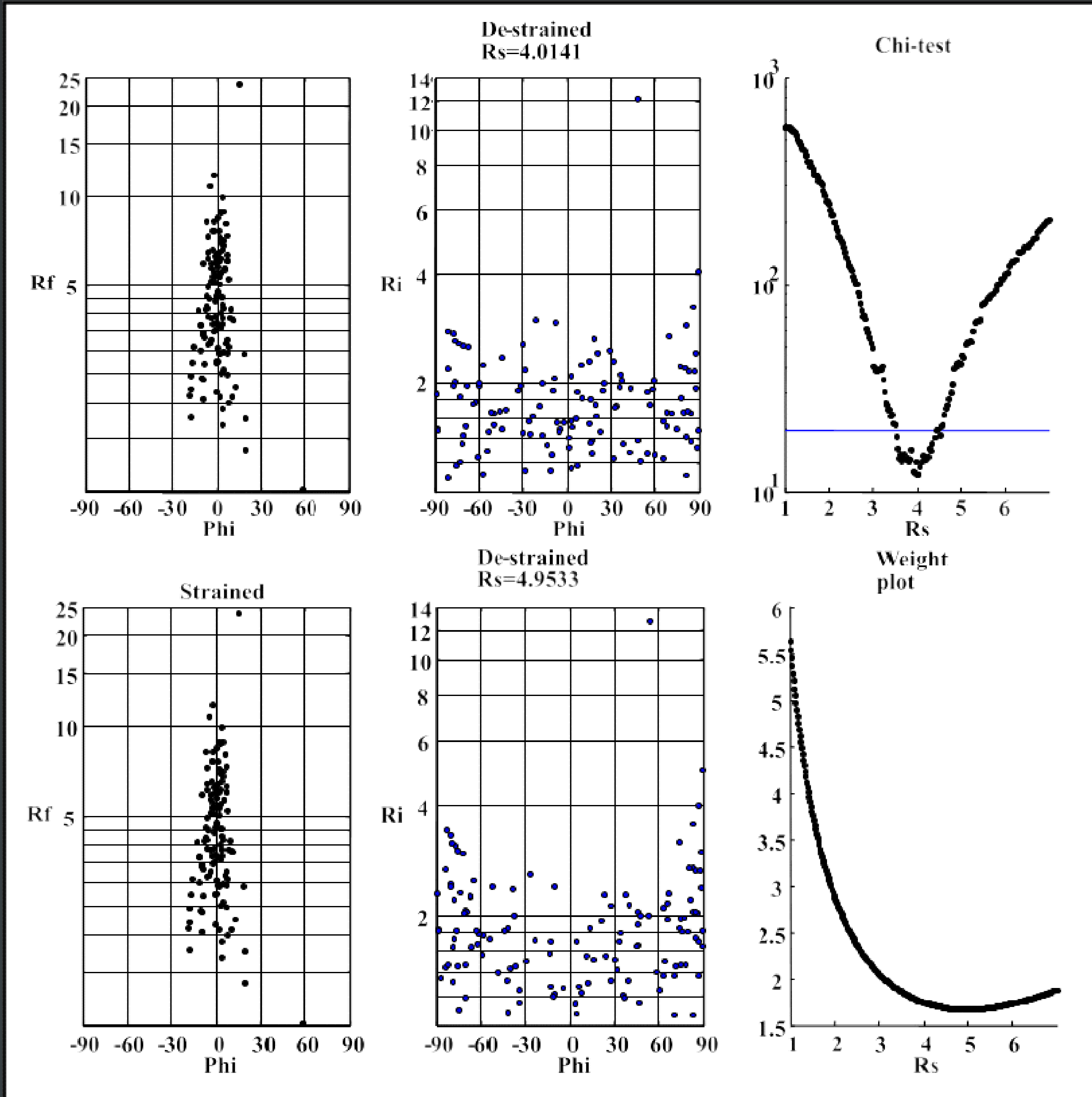


Strain determination methods

Several techniques to estimate strain are available.

Implemented methods:

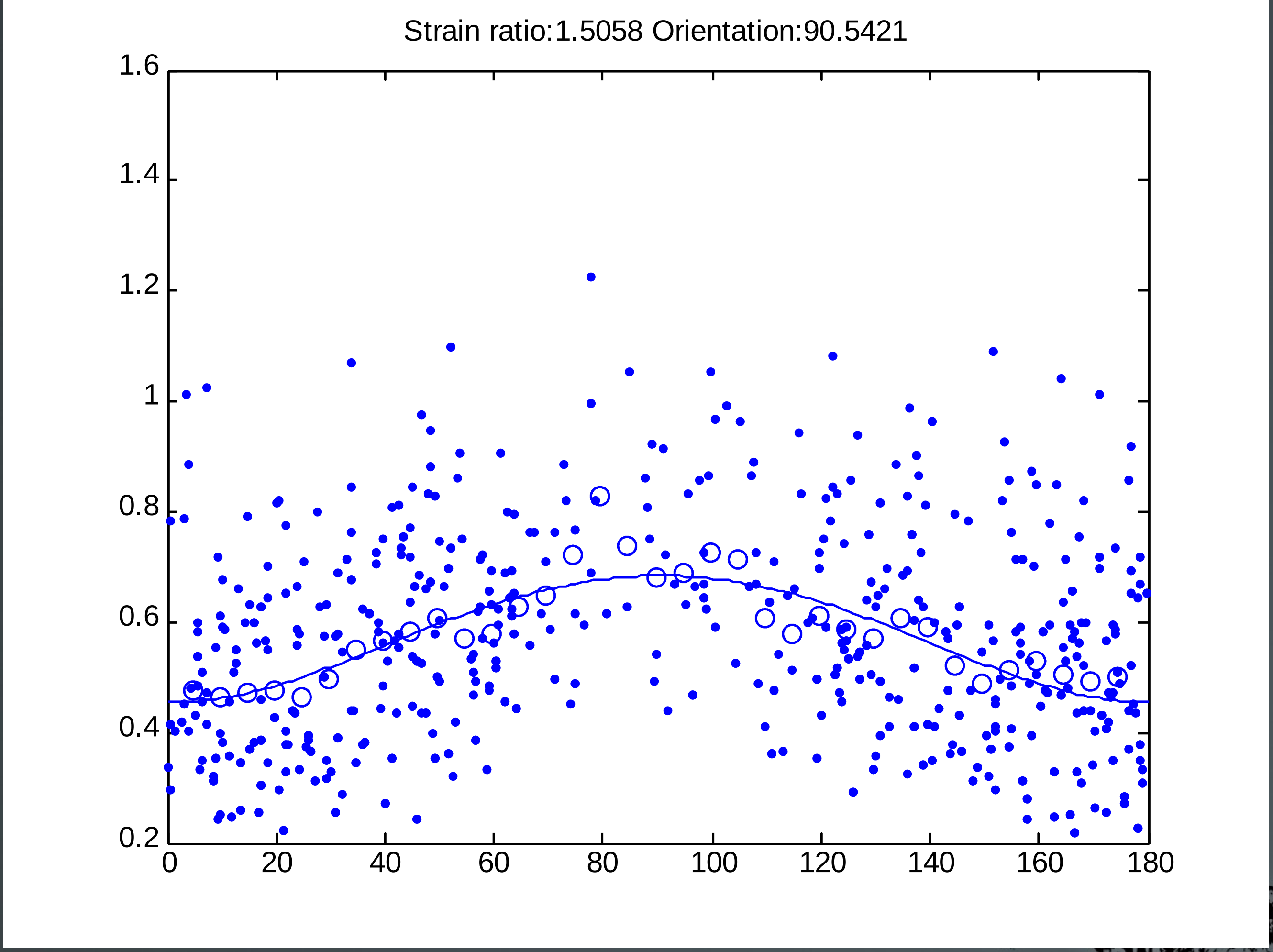
- Rf/ϕ method
- Centre-to-centre method
- Mean matrix of inertia Harvey & Fergusson (1981)
- Delaunay triangulation nearest-neighbor method
- Area weighted Rf/ϕ method
- etc.



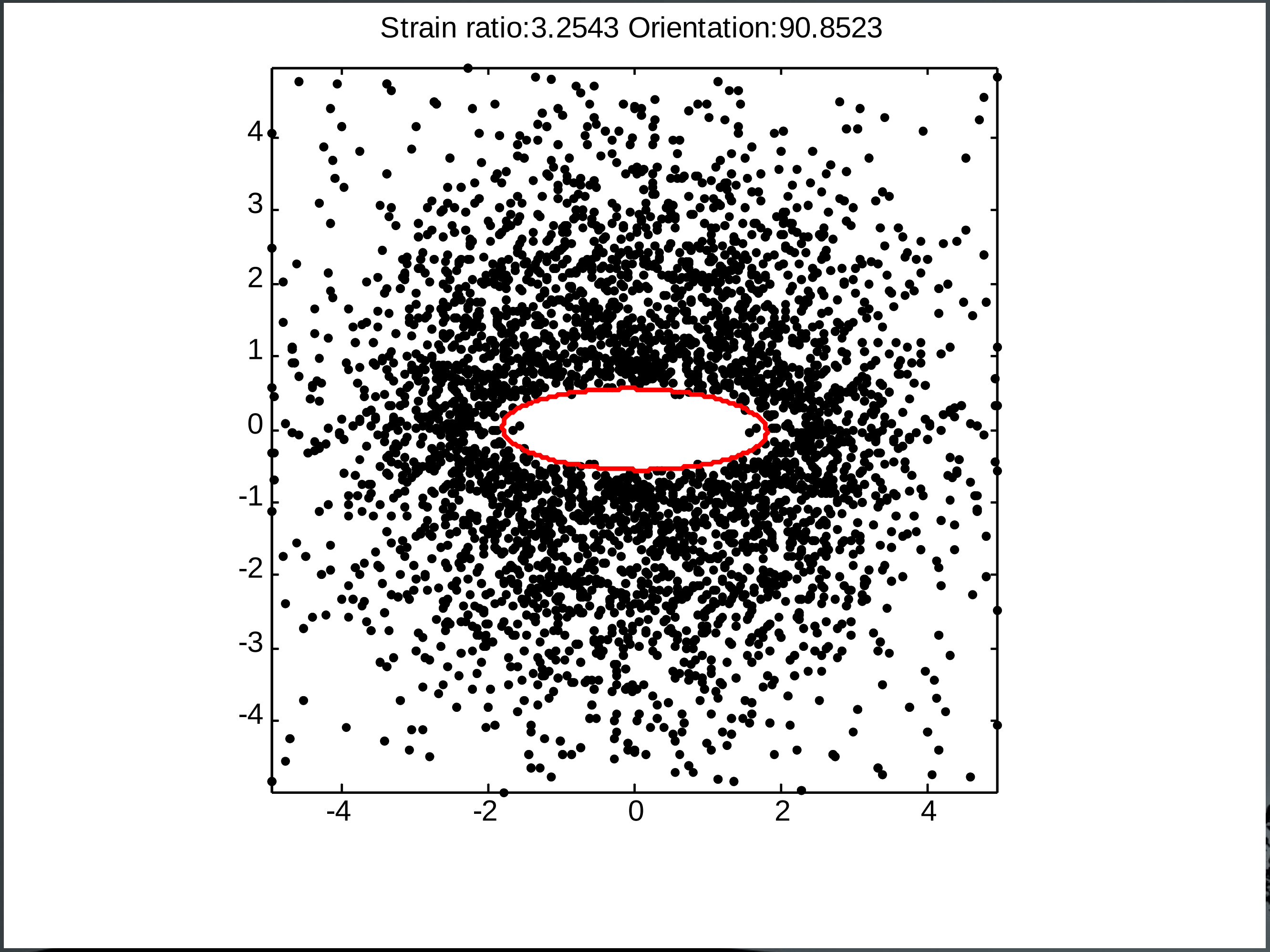
Rf/ϕ method

Area weighted Rf/ϕ method

Nearest neighbour method



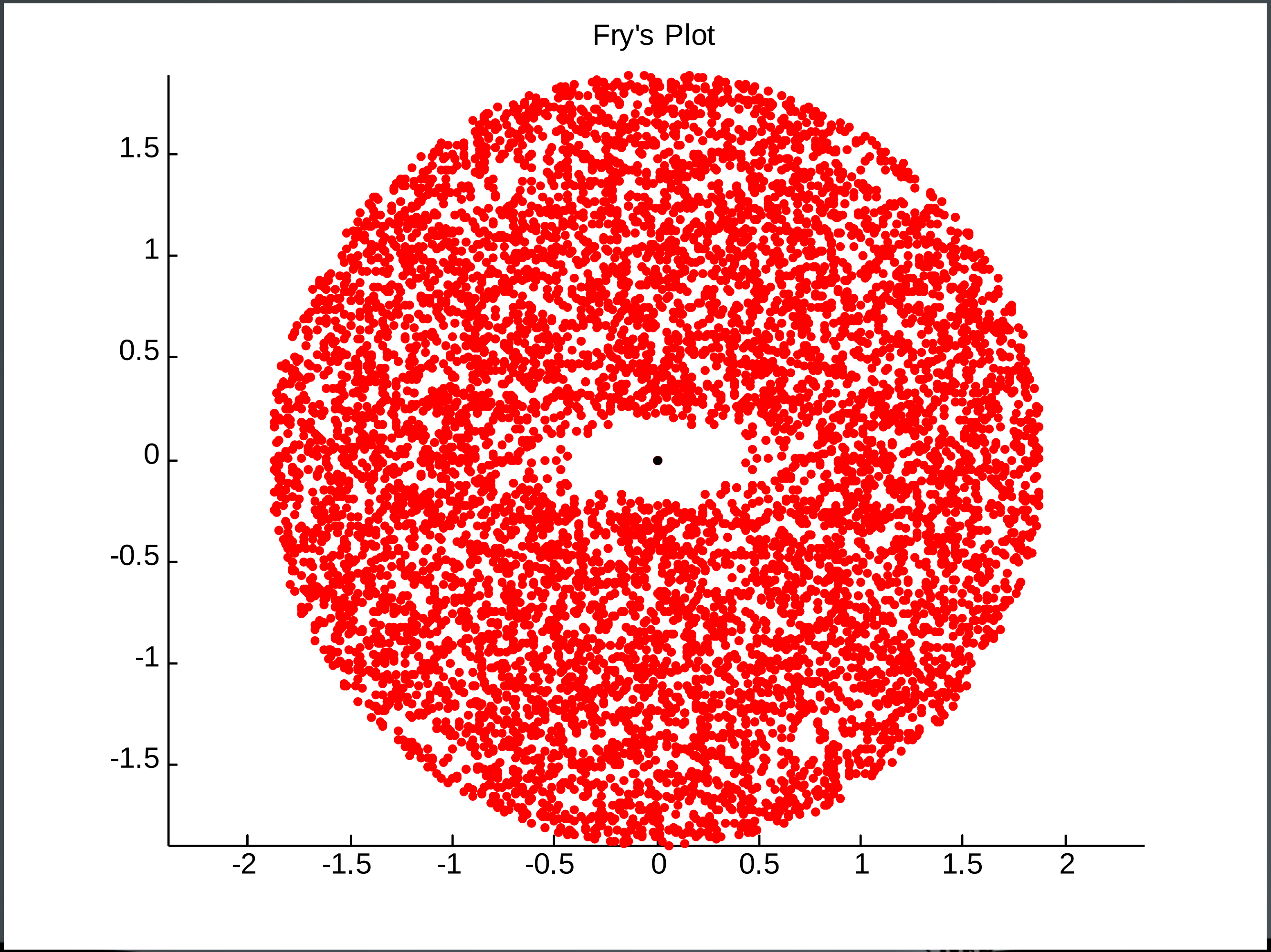
Delaunay triangulation nearest neighbour method



(Mulchrone, 2000)

Centerpoint distribution method

Modified Fry (1979)



Command-line interface

- all PolyLX function are accessible using simple functions
- microstructural data (grain and boundary objects) and results of analyses are stored in MATLAB variables
- users have no limitation to create new scripts (macros) for work automation

```
[output]=function(input);

>>g=shpread;

>>[la,sa,lao,sao]=aorten(g,{opt,plt});
```

```
>>help aorten

AORTEN - Return results of Ellipse fitting using orientation tensor
  Syntax: [la,sa,lao,sao]=aorten(g,opt,plt);
  g can be grain and boundary object(s)
  opt 0..use all vertexes 1..use convexhull of vertexes
  Plot option plt 1..plot crosses 2..plot ellipses 3..ellipses and crosses
```

Grain or boundary selection methods

When we need to analyze only specific objects

Direct indexing (ID based):

```
>>a=get(g(12),’area’);
>>a=get(g(1:100),’area’);
```

Phase selection

```
g(‘qtz’);        id=gpsel(g,’qtz’,’plg’);
>>a=get(g(‘qtz’),’area’);
>>a=get(g( gpsel(g,’qtz’,’plg’) ),’area’);
```

Grain or boundary selection methods II

Query based selection

```
>>r=get(g,'AxialRatio');
>>g(find( r>2 ));
>>a=get(g,'Area');
>>g(find( r>1.5 & a>100000 ));
```

Manual mouse-click selection

```
>>a=get(g( getsel ),'Area');
```