Apreo data loading

Table of Contents

Toolbox 1
MTEX prefs 1
load data 1
Add additional properties from .bin files 1
Setup phase details 2
Plot - Example map from the properties: PQ 2
Plot - IPF 3
Plot - Polefigure (ODF) 4

script to load data from the TFS Apreo microscope - from xTalView. load & basic plotting

Updated:12-Aug-2025 RB/TBB

Toolbox

```
% location of toolboxes - MTEX
mtex_location='C:\Users\ruthb\OneDrive\Documents\MTEX\mtex-5.11.2'; % works
 with 5.8.2, 5.10.2 and 5.11.2

% start mtex if needed
try EBSD;
catch
    run(fullfile(mtex_location,"startup.m"));
end
```

MTEX prefs

```
setMTEXpref('xAxisDirection','east');      %TFS
setMTEXpref('zAxisDirection','outofplane'); %TFS
```

load data

```
% data location
TFS_DataLoc='C:\Users\ruthb\DocumentsOnLaptop\GitHub\MTEX_Workshop\Data
\TFS_ExampleData';

% load data
[ebsdtemp,ebsd_header] = bIDX_to_EBSD(TFS_DataLoc);

ebsd=ebsdtemp; % rename ebsd
ebsd.scanUnit='um'; % update scan unit
```

Add additional properties from .bin files

e.g. PQ (pattern quality)

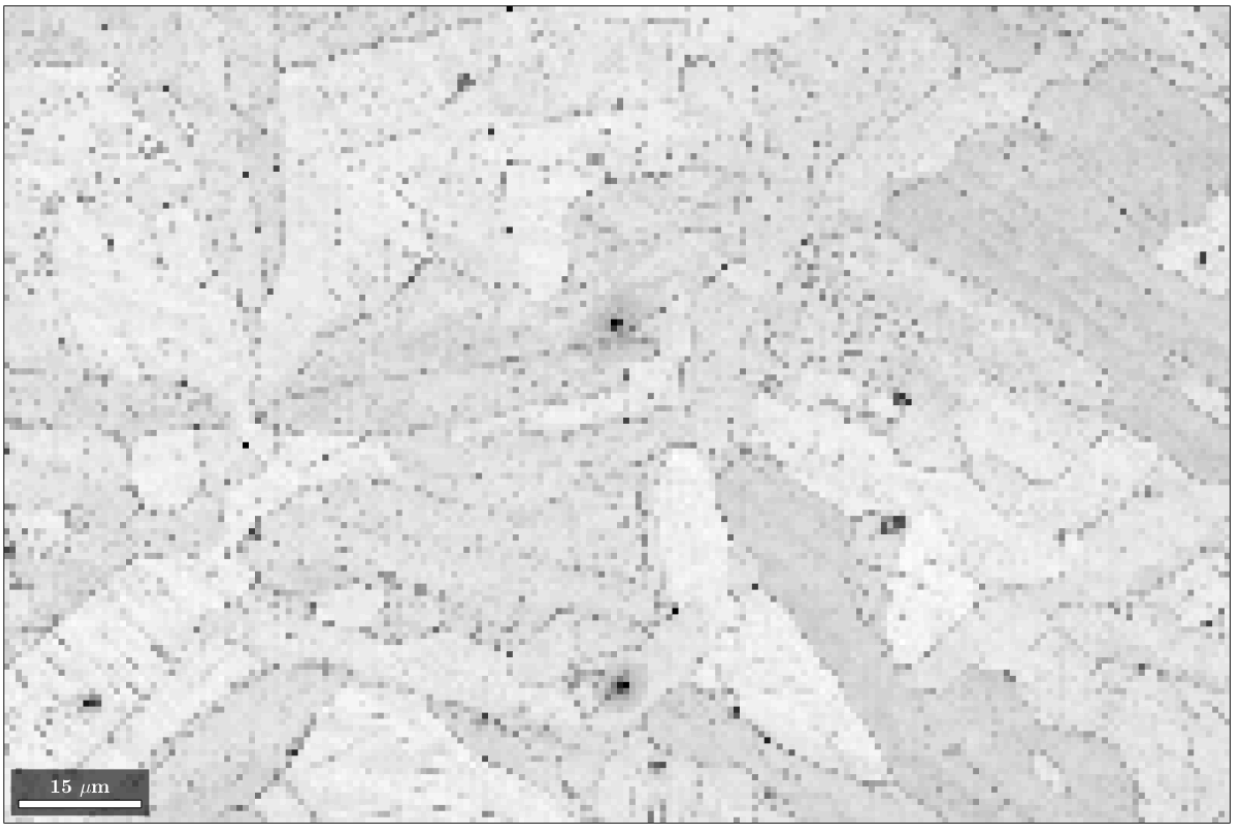
```
% get map size
MapSizeX=ebsd_header.Width;
MapSizeY=ebsd_header.Height;

% bin file to load
bin_location='C:\Users\ruthb\DocumentsOnLaptop\GitHub\MTEX_Workshop\Data
\TFS_ExampleData\results\PQ.bin';

%load data
[Map_Out]=bLoadTFS_Bin(bin_location,MapSizeY,MapSizeX);

%store in properties
ebsd.prop.PQ=Map_Out;
```

Setup phase details

```
% phase 1 details
mineral_name=ebsd.mineralList{2};   % phase name
cs=ebsd(mineral_name).CS;           % crystal symmetry
ipfKey=ipfTSLKey(cs);               % color key - alternative: ipfHSVKey(cs_1)
```

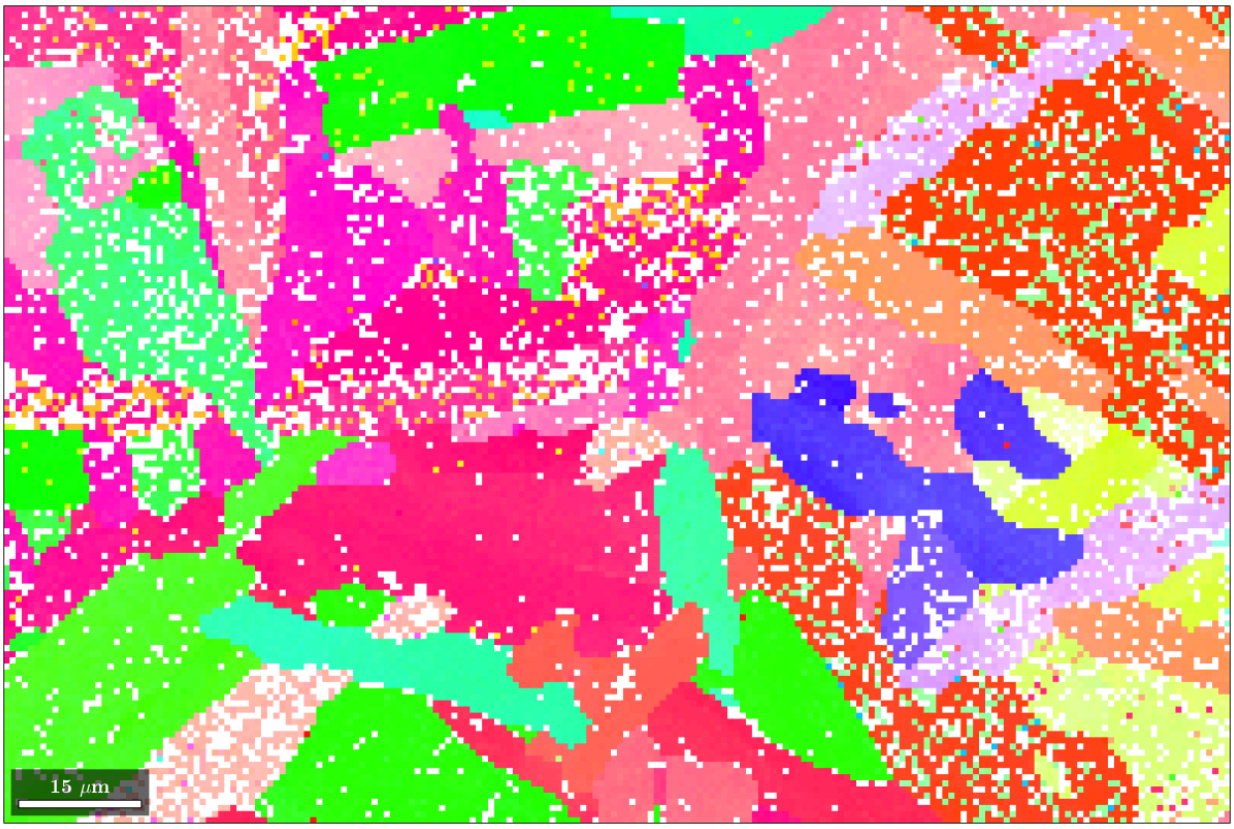
Plot - Example map from the properties: PQ

```
% plot one IPF map
f1=figure;
f1.Color=[1 1 1];
plot(ebsd,ebsd.prop.PQ,'micronbar','on');
colormap('gray'); %grayscale
%clim([0.5 0.95]) % control the colorscale limits (optional)
```



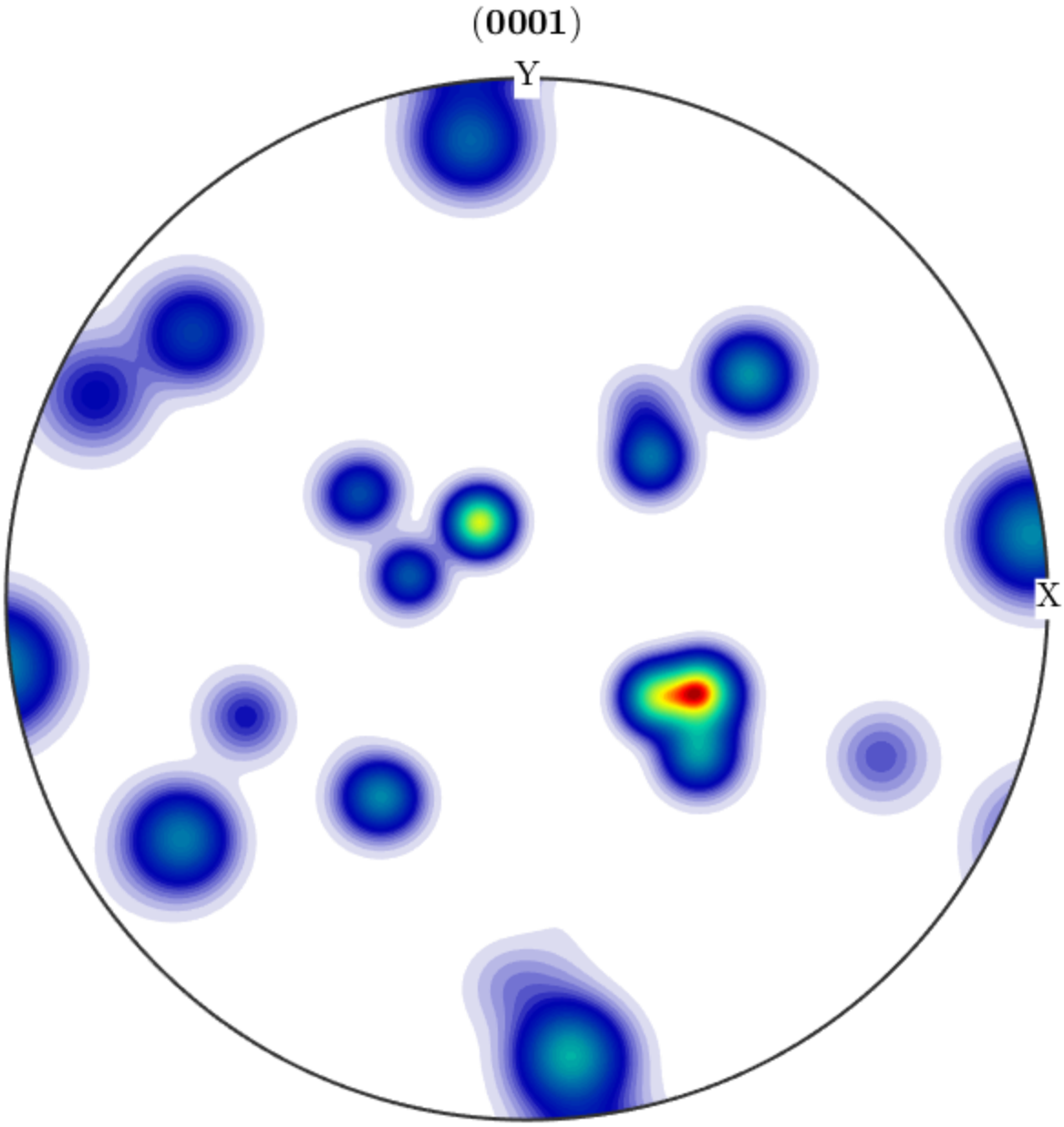
Plot - IPF

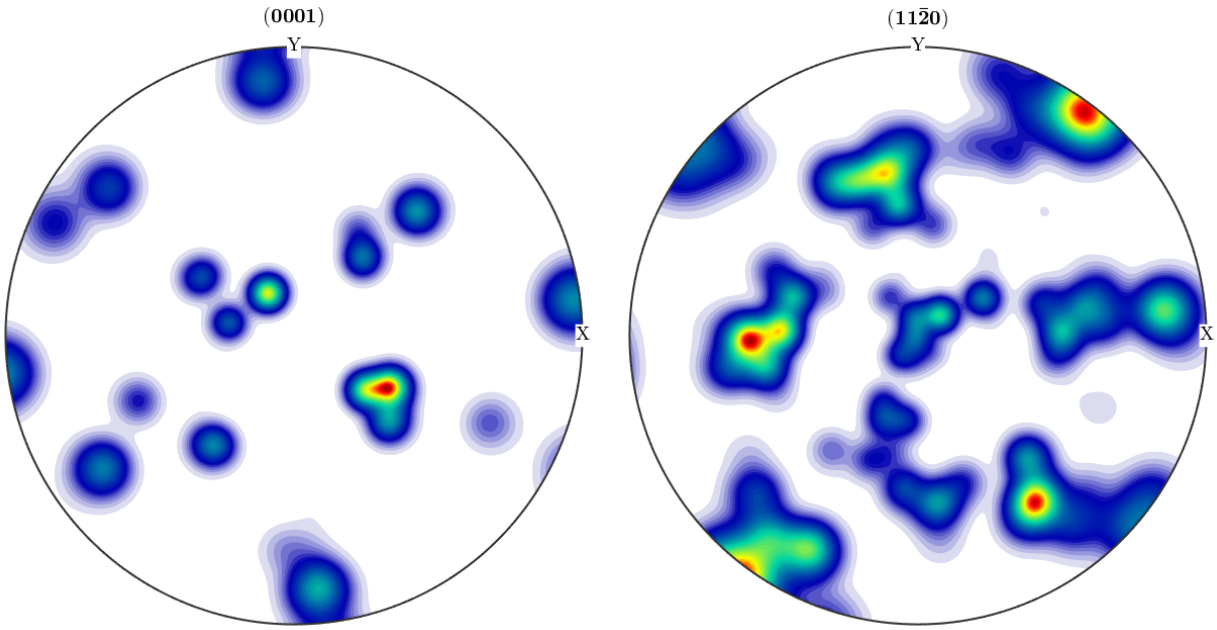
```
% choose IPF direction to plot (z is default if not defined)
ipfKey.inversePoleFigureDirection=zvector;

% plot one IPF map
f1=figure;
f1.Color=[1 1 1];
plot(ebsd,ipfKey.orientation2color(ebsd.orientations),'micronbar','on');
```



Plot - Polefigure (ODF)

```
% compute the ODF
% odf = calcDensity(ebsd(mineral_name).orientations);
% You could also change/fix the half width - compare this different ODF
odf = calcDensity(ebsd('indexed').orientations,'halfwidth',5*degree);

% specify direction or plane - specify direction or plane
h1=Miller(0,0,1,cs,'hkl'); % e.g. {001}/{0001}
h2=Miller(1,1,-2,0,cs,'hkil'); % e.g. {11-20}
% h3=Miller(0,0,1,cs_1,'uvw'); % e.g. <001>

% plot
figure;
% plot the pole figure representation of the ODF (eangle = equal angle)
plotPDF(odf,h1,'projection','eangle');

% plot multiple
figure;
plotPDF(odf,[h1,h2],'projection','eangle');
```


(0001)
Y
X



Published with MATLAB® R2022b