Data processing with XDS and associated programs

Kay Diederichs
kay.diederichs@uni-konstanz.de

CCP4@DLS 11/2025

Outline

1) This presentation: Introduction to *XDS* (*XSCALE*, *XDSCC12*)

2) Demo: Processing data with *XDSGUI* (and using *XDSSTAT*, *XDSCC12*, *SPOT2PDB*, *COOT*, *POINTLESS*) of example data

3) Until Wednesday: process and analyze YOUR experimental X-ray data

The *XDS* program suite

- Original author: Wolfgang Kabsch (Max-Planck-Institute Heidelberg)
- Since ~1986
- I joined 2007

The *XDS*+ programs

- *XDS*: the main program - indexing, integrating, scaling, statistics
- *XSCALE*: scale several *XDS* intensity data sets together; zero-dose extrapolation; statistics
- *XDSCONV*: convert to MTZ / SHELX /... format (AIMLESS and CTRUNCATE are not needed!)

Programs independent from the *XDS* distribution:

- *XDS-Viewer*: inspect diagnostic images written by XDS, or (single) data frames. (There exist other viewers, like *adxv* or *dials.image_viewer*).
- *XDSSTAT*: additional statistics
- *XDSGUI*: graphical user interface for XDS, SHELX C/D/E, ARCIMBOLDO (open source)
- *XDSCC12*: (XDS) which frames are bad? Sometimes reasons are revealed.
  (XSCALE) which data sets are bad?
- *XSCALE_ISOCLUSTER* multi-data-set: visualize relations and cluster

Sources of information

- *XDS* main website: https://xds.mr.mpg.de ; complete, accurate, up-to-date documentation; download: Linux, Intel- and Silicon-Mac; for Windows use WSL.
- XDSwiki: https://wiki.uni-konstanz.de/xds
  Installation; data sets; documentation; download; links to e.g. Matthew J. Whitley's excellent tutorial given at CSHL 2018
- CCP4 bulletin board
- SBGrid talk (May 2020) at https://www.youtube.com/watch?v=3WU9NrILECo
- Making a difference in multi-data-set crystallography: simple and deterministic data-scaling/selection methods. Assmann, G.M., Wang, M., Diederichs, K. (2020) Acta Cryst D76, 636 (serial crystallography, *XDSCC12*)
- *XDSGUI* paper (2023) https://doi.org/10.1107/S1600576723007057
- Cheat sheet in XDSwiki https://wiki.uni-konstanz.de/xds/index.php/Cheat_sheet shows a general and detailed procedure

Automatic processing with *XDS*

- beamline software (provides XDS.INP)
- scripts: *xia2* (CCP4), *autoPROC* (Globalphasing), *generate_XDS.INP* (XDSwiki), *fast_dp* (Diamond, APS, NSLSII, ...), *xdsme* (Soleil), *autoxds* (SSRL), *autoprocess* (CMCF), … For good data, these give good results. *autoPROC* is the most advanced.
- CCP4: *pointless, xdsconv* (type CCP4_I+F, or CCP4, or CCP4_I, or CCP4_F)
- SHELX: *shelxc* reads XDS_ASCII.HKL

Principle of *XDS* processing

- There is one JOB= line in XDS.INP which specifies a list of tasks:

JOB= XYCORR INIT COLSPOT IDXREF DEFPIX INTEGRATE CORRECT

- data reduction is divided into tasks in a **modular** way
- information storage/exchange/flow between tasks by data files which may be inspected/analyzed
- each task needs the result from the previous tasks
- fine-tuning of a task does *not* require previous tasks to be repeated
- each task writes its output file <TASK>.LP

The tasks are ...

- XYCORR : write positional correction files (X-CORRECTIONS.cbf, Y-CORRECTIONS.cbf)
- INIT : find background pixels (defaults usually OK)
- COLSPOT: find reflection positions
- IDXREF : "index" reflections; user may supply/choose spacegroup
- XPLAN [not required] : strategy for data collection
- DEFPIX : mask shadows on detector (use *XDSGUI*!)
- INTEGRATE : evaluates intensities on all frames, writes INTEGRATE.HKL and SHOW_HKL.cbf
- CORRECT : scales, rejects outliers, statistics, writes scaled, unmerged XDS_ASCII.HKL (and other files)

Example XDS.INP

```
JOB= XYCORR INIT COLSPOT IDXREF DEFPIX INTEGRATE CORRECT
ORGX=1546 ORGY=1552          !Detector origin (pixels); e.g. NX/2 NY/2
DETECTOR_DISTANCE=180        !(mm)
OSCILLATION_RANGE=0.10       !degrees (>0)
X-RAY_WAVELENGTH=0.980243    !Angstroem
NAME_TEMPLATE_OF_DATA_FRAMES=frms/wga2-27_1_????.img
DATA_RANGE=1 3600             !Numbers of first and last data image collected
BACKGROUND_RANGE=1 10        !Numbers of first and last data image for background
SPACE_GROUP_NUMBER=  19      !0 for unknown crystals; cell constants are ignored.
UNIT_CELL_CONSTANTS= 44.4   86.4   104.5  90 90  90  ! not required if spgr=0
REFINE(IDXREF)=BEAM AXIS ORIENTATION CELL DISTANCE
REFINE(INTEGRATE)=DISTANCE BEAM ORIENTATION CELL ! AXIS
ROTATION_AXIS= 1.0 0.0 0.0
INCIDENT_BEAM_DIRECTION=0.0 0.0 1.0
FRACTION_OF_POLARIZATION=0.99                    ! SLS X06SA
POLARIZATION_PLANE_NORMAL= 0.0 1.0 0.0
DETECTOR=CCDCHESS         MINIMUM_VALID_PIXEL_VALUE=1     OVERLOAD=65000
DIRECTION_OF_DETECTOR_X-AXIS= 1.0 0.0 0.0
DIRECTION_OF_DETECTOR_Y-AXIS= 0.0 1.0 0.0
MINIMUM_NUMBER_OF_PIXELS_IN_A_SPOT=3 ! 3 for very sharp reflections, e.g. 6 for ordinary/bad data sets
VALUE_RANGE_FOR_TRUSTED_DETECTOR_PIXELS= 7000 30000 !Used by DEFPIX
                                  !for excluding shaded parts of the detector.
INCLUDE_RESOLUTION_RANGE=50.0 1.3 !Angstroem; used by DEFPIX,INTEGRATE,CORRECT
```

Bold keyword/parameter pairs are required; yellow ones change between experiments; this may need adjustment since it depends on the crystal. Documentation: xds.mr.mpg.de/html_doc/xds_parameters.html

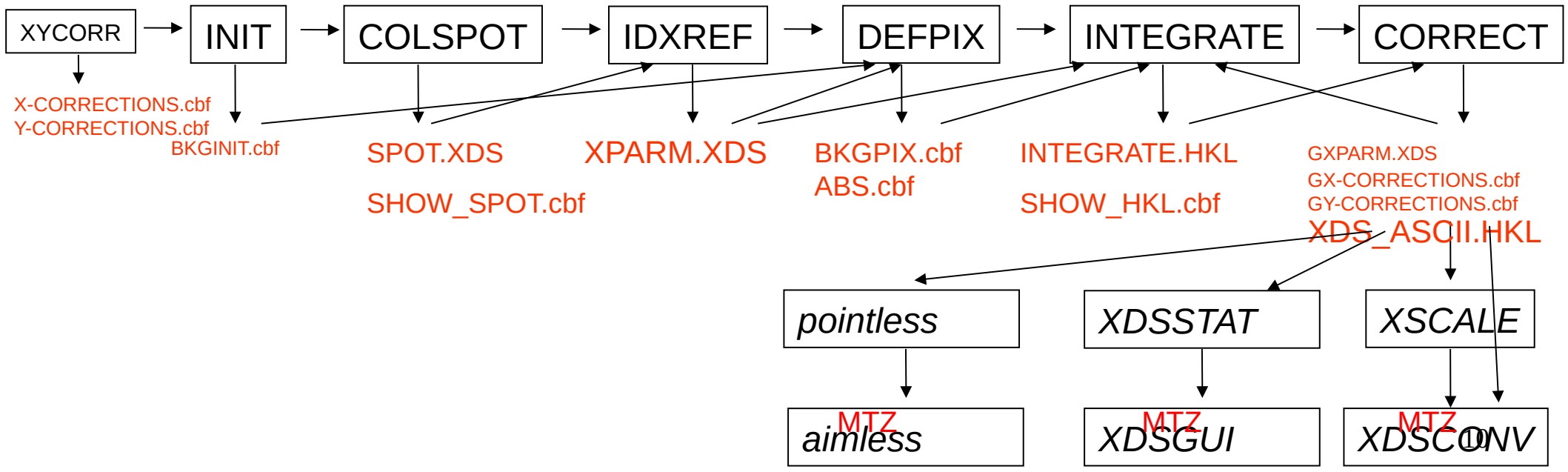
Information flow

NAME_TEMPLATE_OF_DATA_FRAMES
DETECTOR

OSCILLATION_RANGE
SEPMIN
STRONG_PIXEL

ORGX
ORGY
DETECTOR_DISTANCE
X_RAY_WAVELENGTH
SPACE_GROUP_NUMBER

DATA_RANGE



Example XSCALE.INP

```
!============ EXAMPLE 3: specific reindexing of input data sets
!
!      Use of specific reindexing of input data sets for resolving
!      indexing ambiguities in the scaled output data set. This
!      happens if the crystal's space group symmetry is lower than
!      its lattice symmetry.
!
RESOLUTION_SHELLS= 100 10 6 4 3 2 1.9
SPACE_GROUP_NUMBER=78
UNIT_CELL_CONSTANTS=57.39 57.39 106.9   90 90 90
OUTPUT_FILE=scaf8_all_merged.hkl
MERGE=TRUE FRIEDEL'S_LAW=FALSE
STRICT_ABSORPTION_CORRECTION=TRUE
INPUT_FILE= ../xds-1_2/XDS_ASCII.HKL
REIDX_ISET= -1  0  0  0  0  1  0  0  0  0 -1  0
INPUT_FILE= ../xds-2_1/XDS_ASCII.HKL
INPUT_FILE= ../xds-3_1/XDS_ASCII.HKL
INPUT_FILE= ../xds-1_4/XDS_ASCII.HKL
INPUT_FILE=*../xds-5_1/XDS_ASCII.HKL
```

Bold keyword/parameter pairs are required. Complete documentation at xds.mr.mpg.de/html_doc/xscale_parameters.html

Output file XSCALE.LP shows level of systematic error **ISa** before/after scaling.

Example XDSCONV.INP

```
! UNIT_CELL_CONSTANTS= 10 20 30 90 90 90
! SPACE_GROUP_NUMBER= 96
! GENERATE_FRACTION_OF_TEST_REFLECTIONS=0.05

INPUT_FILE=XDS_ASCII.HKL
OUTPUT_FILE=temp.hkl CCP4_I+F ! to obtain MTZ; alternatives: SHELX, CCP4, ...
FRIEDEL'S_LAW=FALSE ! store anomalous signal in output file even if weak
```

Bold keyword/parameter pairs are required. Complete documentation at xds.mr.mpg.de/html_doc/xdsconv_parameters.html

For important data, do not rely on automatic data processing

Synchrotrons typically run multiple pipelines, and the user has to choose ...
What can go wrong in automatic data processing?

- does not handle radiation damage (we need to discard frames towards the end of the data set, but where should we cut?)
- does not handle shadowed areas of detector
- does not handle indexing problems (multiple lattices, ice, …) flexibly
- does not enable to visualize diffraction in reciprocal space
- does not optimize processing
- there are no perfect criteria for the quality of a data set - “Table 1” does not tell the whole story

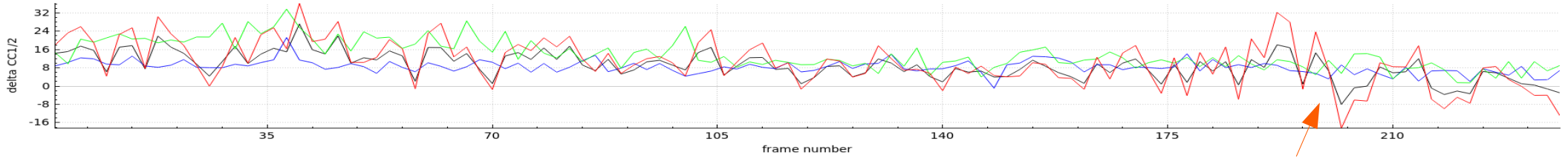
Difficult data sets typically benefit from human insight.

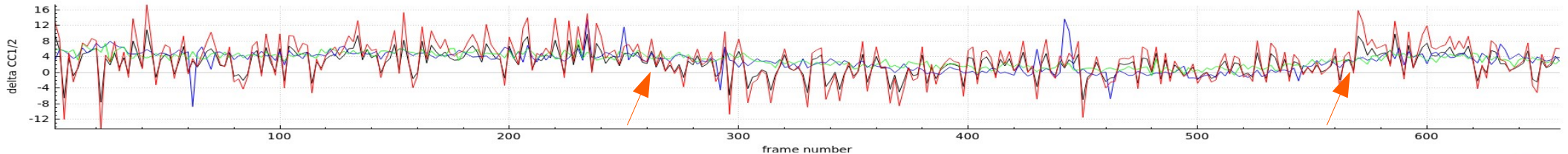
Manual processing with *XDSGUI*

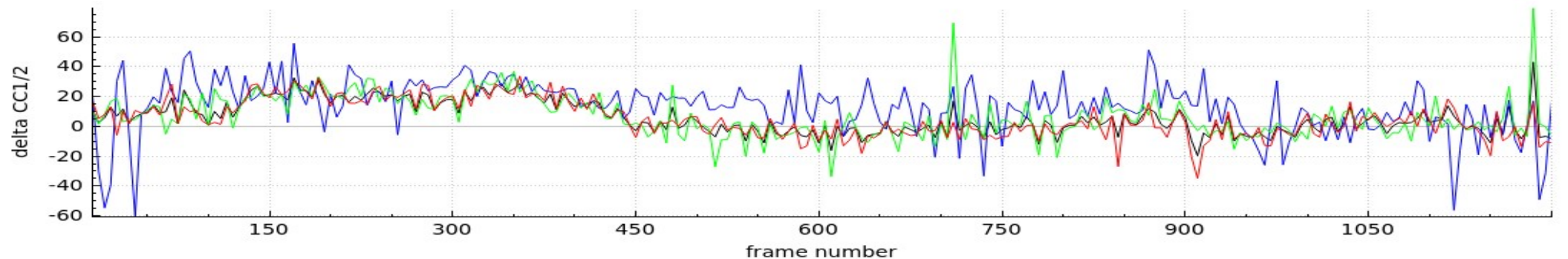
- problems in phasing and refinement: often due to bad / wrong data processing – re-process raw data!
- visually inspect frames; mask shadows
- optimize processing parameters, frame range, resolution cutoff ..
- understand experiment: presentation of tables as plots: **tools** tab/Further analyses/show spots in reciprocal space
- user-extensible / modifiable commands

Brehm, Triviño, Krahn, Usón and Diederichs (2023) XDSGUI: a graphical user interface for XDS, SHELX and ARCIMBOLDO. *J. Appl. Cryst.* **56**,1585-1594

3 examples of single data sets (plots from *XDSGUI*)







XDSCC12: calculates $\Delta CC_{1/2,i} = CC_{1/2,with\_i} - CC_{1/2,without\_i}$

Three resolution ranges (blue=low; green=medium; red=high) - i refers to batches of width 1°

- find bad frame ranges
- radiation damage

Optimize data processing

- XDSwiki:Optimisation#Re-INTEGRATEing_with_the_correct_spacegroup.2C_refined_geometry_and_fine-slicing_of_profiles

  XDSwiki:Optimisation#using_the_refined_values_for_beam_divergence_and_mosaicity_for_re-integration

- **tools** tab:
  * “Saving and comparing good results” and “Optimizing data quality”. After changing parameters, run “JOB=DEFPIX INTEGRATE CORRECT”, compare and save if better/restore old if worse.
  * “Further analysis”: Inspect indexed/unindexed spots in reciprocal space

- consider use of *STARANISO* if anisotropy is strong (i.e. visible)

… may make the difference between structure solved or not, interpretable or non-interpretable map, good or bad refinement, ...

The signal and the noise: random and systematic errors

Random error

True randomness occurs due to the quantum nature of matter.

- counting photons
- electronic noise (detector, electronics)

Random error is proportional to square root of measured value

Level of random error is given by R-values, $CC_{1/2}$ and I/sigma ("Data precision and accuracy" lecture)

Systematic error

- crystal: conditions, composition, conformation, damage due to experiment, ...
- apparatus: shadows, absorption, vibrations, photon/electron flux ...
- processing software: inaccurate or incomplete modelling of experiment

Systematic error is proportional to measured value (often 1..10% but sometimes much more e.g. in case of shadows and overloads)

Level of systematic error is given by ISa = asymptotic signal/noise (CORRECT.LP, AIMLESS, dials.scale)

XDSCC12 writes a sorted XSCALE.INP.rename_me

```
SPACE_GROUP_NUMBER=    1
UNIT_CELL_CONSTANTS=     53.00     59.93     61.34  99.975 105.633 119.747
OUTPUT_FILE= temp.ahkl
PRINT_CORRELATIONS= FALSE
SAVE_CORRECTION_IMAGES= FALSE
FRIEDEL'S_LAW= FALSE
! median of delta-cc1/2 "only" values=    11.492
! noise (MAD) of these values=     0.932
! median of delta-cc1/2 "all" values=    12.189
! noise (MAD) of these values=     1.941
! median of delta-cc1/2-ano "only" values=    -2.201
! noise (MAD) of these values=     2.699
! median of delta-cc1/2-ano "all" values=    -2.010
! noise (MAD) of these values=     2.614
! input files sorted by deltacc12_only (highest first):
! deltacc12 only / all:   12.4231   14.1301 deltacc12-ano only /all:   -4.9002  -4.6243
INPUT_FILE=../xds_14_2/XDS_ASCII.HKL
! deltacc12 only / all:   10.5600   10.2488 deltacc12-ano only /all:    0.4976   0.6043
INPUT_FILE=../xds_14_3/XDS_ASCII.HKL
...
```

output for two datasets in P1 that scale quite well

XDSCC12 output for data from *XSCALE*

```
 overall CC1/2:    99.784 nref=   29744 (but the overall CC1/2 is meaningless!)
 <CC1/2>:    68.971 (frequency-weighted average of CC1/2 in resolution shells)
 CC1/2 in resolution shells:
   99.8   99.6   99.6   99.3   97.8   92.9   81.3   52.7   33.4   12.5
 CC* in resolution shells:
   99.9   99.9   99.9   99.8   99.4   98.1   94.7   83.1   70.7   47.1
 frequency, i.e. number of unique reflections in resolution shells:
    970   1741   2257   2669   3005   3336   3604   3904   4104   4154

 headings for lines starting with a,b,c:
a:  <CC1/2> of each dataset:
a:   reflections of this dataset only      reflections of all datasets
a: set   nref    with without   delta     nref    with without   delta
b: delta-CC1/2 in resolution shells
c: # reflections for delta-CC1/2
a      1  27910  66.807  60.517  10.560    28672  66.486  60.349  10.249
b 30.906 29.045 30.431 27.944 34.663 27.727 23.134 14.772 13.087 5.266
c    912   1654   2173   2527   2803   3169   3394   3679   3849   3750
a      2  25492  68.433  61.215  12.423    26174  70.905  63.097  14.130
b 27.445 18.627 5.799 14.503 25.465 32.858 31.704 23.753 13.638 -.254
c    858   1486   2064   2349   2517   2928   3153   3435   3317   3385
```

output for two datasets in P1 that scale quite well

Garbage in – garbage out!

Lots of possibilities for mistakes in / problems due to the experiment:

- use a modern detector: at synchrotrons, Pilatus or Eiger
- wrong exposure → radiation damage: fix is to expose weakly, and collect 360°
- wrong oscillation range (“traditional” 1°): fix is to collect with Δφ=0.1°
- Badly centered crystal moves out of the beam during rotation: fix is …
- Ice: the fix is watching and listening to experts (Elspeth Garman) and practicing. According to *autoPROC*, ice rings affect 5-10% of all reflections if present.
- Multiple crystals in the beam – overlaps deteriorate data quality a lot.

It is not the goal of the experiment to collect as many data sets as are technically possible, but rather to collect the best data technically possible, for a given project.

Summary

Data processing is the crucial link between experiment and structure

* garbage in – garbage out!
* for important data or in case of downstream difficulties, do not rely on automatic data processing
* manual checks are easily performed with *XDSGUI*
* try to optimize data processing – this converts noise to signal, and may enable structure solution, and/or improve refinement

Thank you!