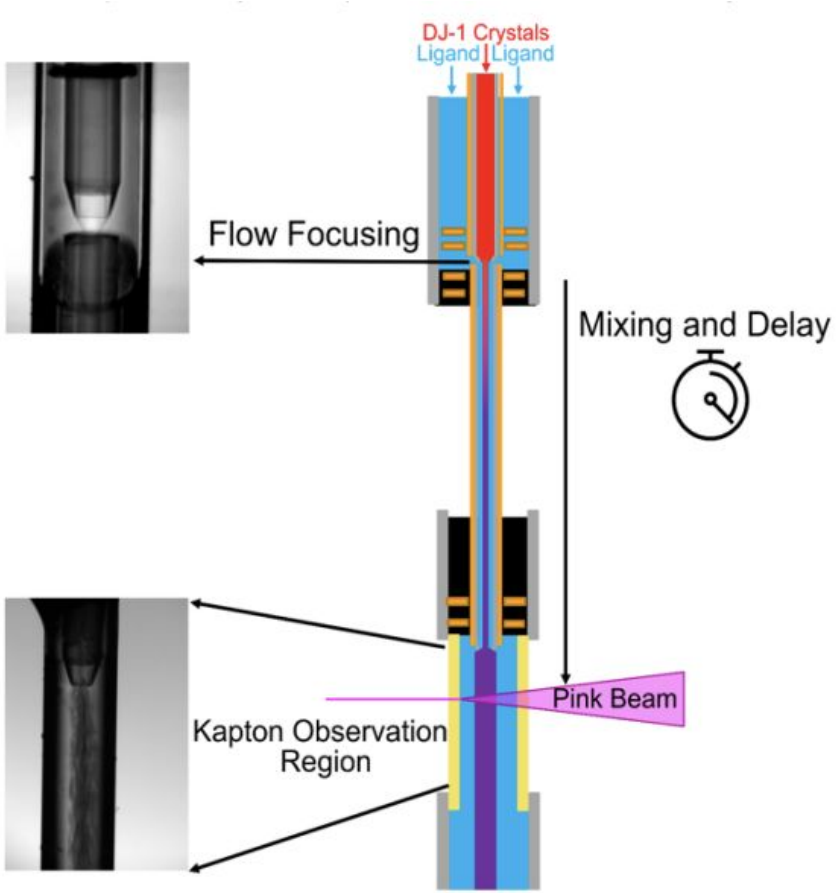
ID-29 MISC Data Processing

Kara Zielinski

Experiment



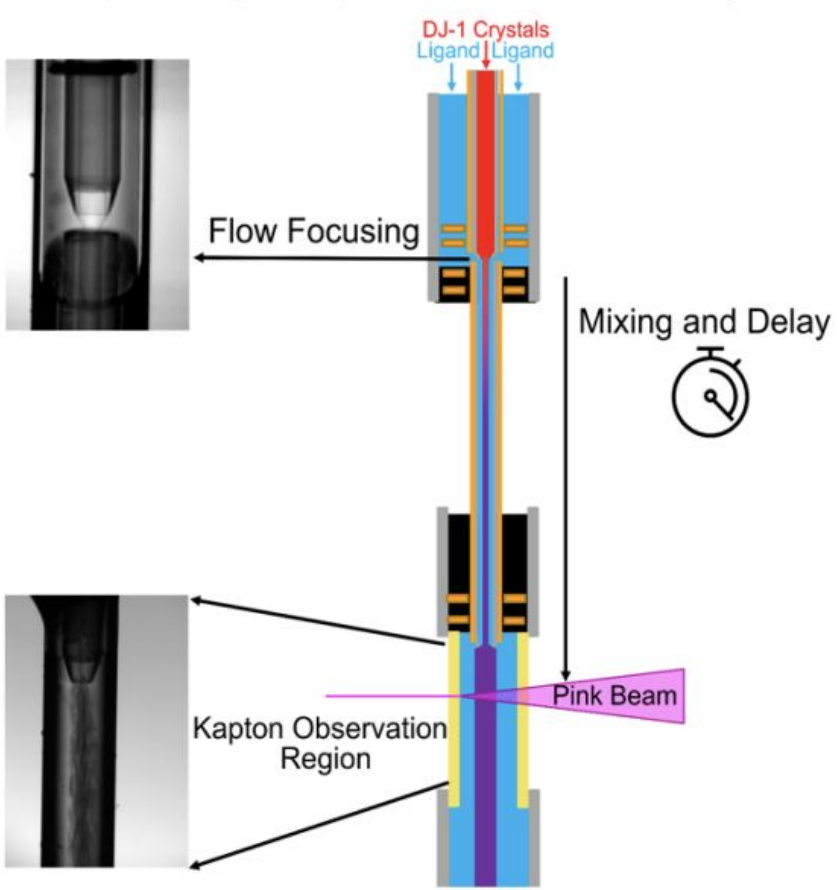
ID-29 Settings

90 us exposure time
231.25 Hz rep rate
2 x 10^15 ph/s, 1% BW
2 um x 4 um beam

Data collection:
SOS chips
Kapton flow cell

200k-500k runs
2-5 runs per dataset

Experiment



ID-29 Settings

90 us exposure time
231.25 Hz rep rate
2 x 10^15 ph/s, 1% BW
2 um x 4 um beam

Data collection:
SOS chips
Kapton flow cell

200k-500k runs
2-5 runs per dataset

Due to Kapton burning, did +/- 20 um oscillation

Context

- ESRF ID-29 mostly supports CrystFEL processing
- ID-29 has collaboration with James Beilsten-Edmunds on dials.ssx but work in progress
- We figured we would just try dials.stills_process, likely the first people to use dials.stills_process on ID-29 data

Resources

1. 2017 dials.still_process wiki: http://viper.lbl.gov:8080/cctbx.xfel/index.php/2017_dials.stills_process
2. 2018 dials.still_process github wiki: https://github.com/phyy-nx/dials_refinement_brewster2018/wiki
3. cctbx.xfel github (especially the merge page): https://github.com/cctbx/cctbx_project/tree/master/xfel/merging
4. Recent cctbx.xfel pre-print: https://www.biorxiv.org/content/10.1101/2025.05.04.652045v1
5. Aaron Brewster
6. Kevin Dalton

Processing Overview: Run by run on the command line

1. dials.stills_process
   a. With starting geometry
   b. With space group and starting unit cell
2. dials.combine_experiments and dials.refine
3. Quality check
   a. dxtbx.plot_detector_models combined.expt refined.expt
   b. cctbx.xfel.detector_residuals refined.* hierarchy=1
   c. dials.unit_cell_histogram combined.expt
4. dials.stills_process redo
   a. Update geometry
   b. Update unit cell
5. Repeat cycle 2-3 times until converge
6. dials.stills_process
   a. Optimized geometry/unit cell
   b. Add in multi lattice
7. cctbx.xfel.merge

Processing Overview: Run by run on the command line

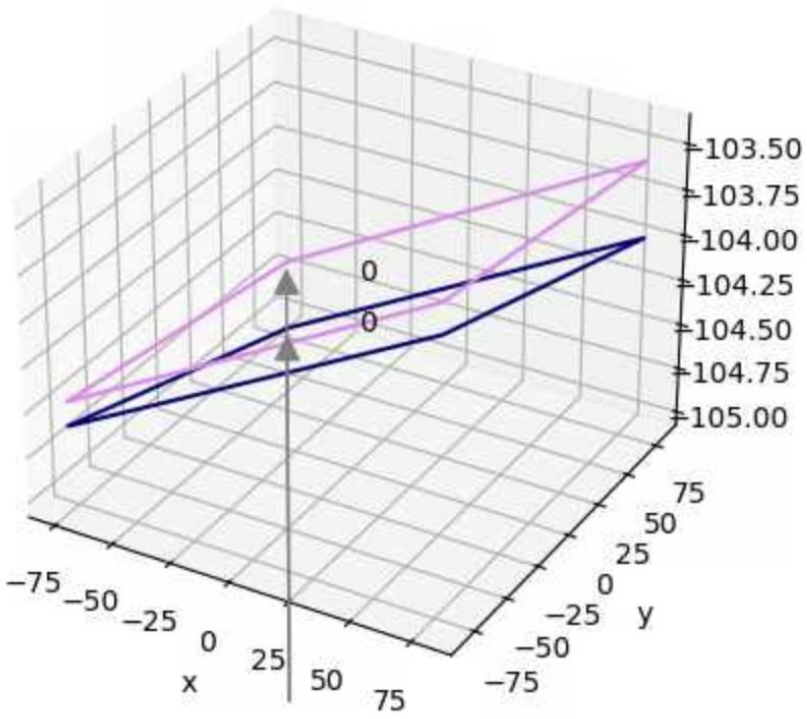
1. dials.stills_process
   a. With starting geometry
   b. With space group and starting unit cell
2. dials.combine_experiments and dials.refine
3. Quality check
   a. dxtbx.plot_detector_models combined.expt refined.expt
   b. cctbx.xfel.detector_residuals refined.* hierarchy=1
   c. dials.unit_cell_histogram combined.expt
4. dials.stills_process redo
   a. Update geometry
   b. Update unit cell
5. Repeat cycle 2-3 times until converge
6. dials.stills_process
   a. Optimized geometry/unit cell
   b. Add in multi lattice
7. cctbx.xfel.merge

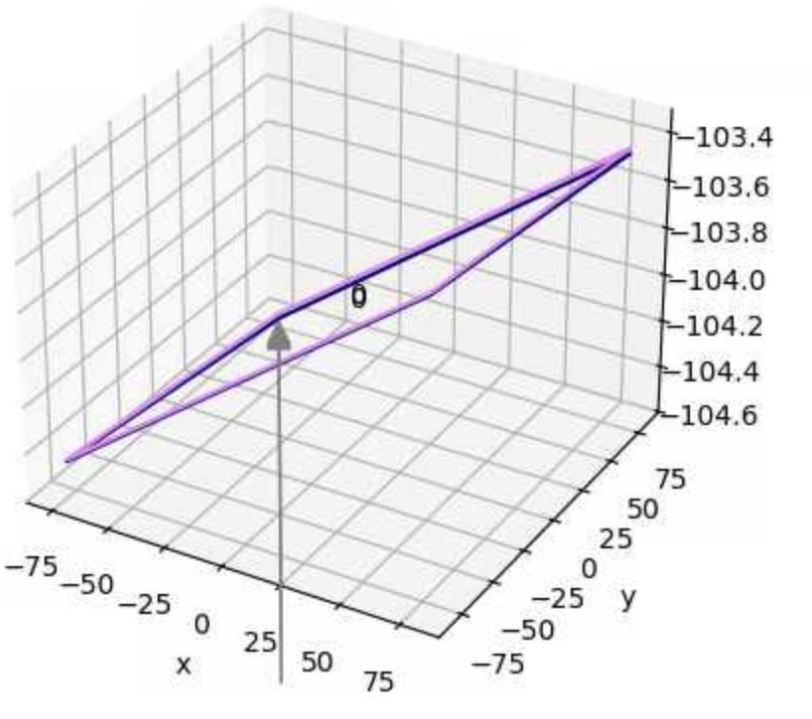
Depending on number of hits per run - a few different options for refinement

1. Aaron recommends chunks of 2000
2. I have gone up to ~10,000
3. You break everything at ~30,000 - need alt approach

Original Processing

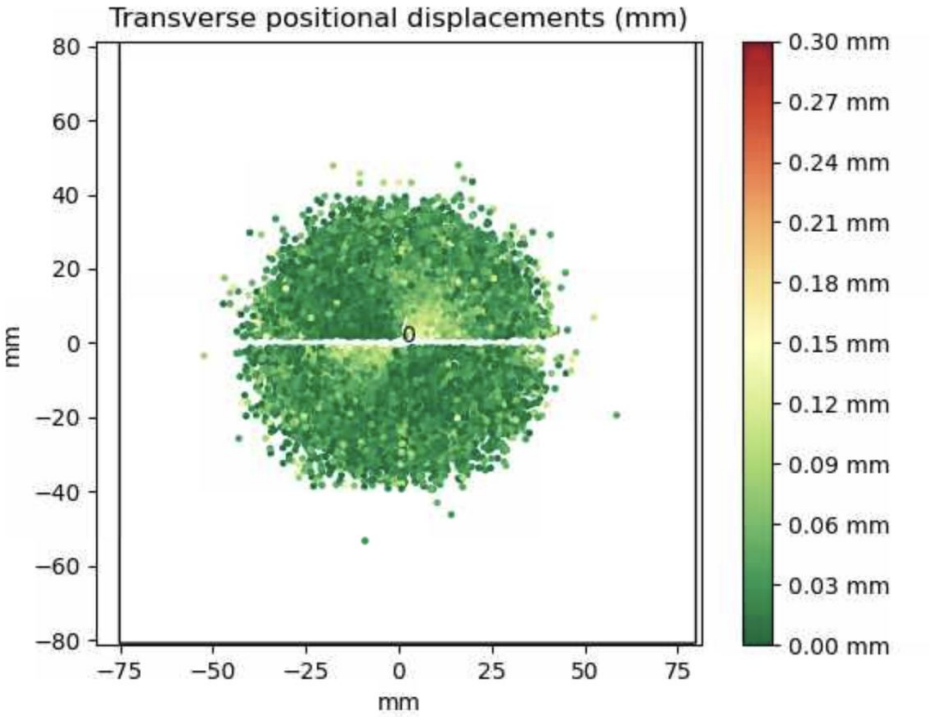
Did refinement cycle, evaluated with `dxtbx.plot_detector_models combined.expt refined.expt`





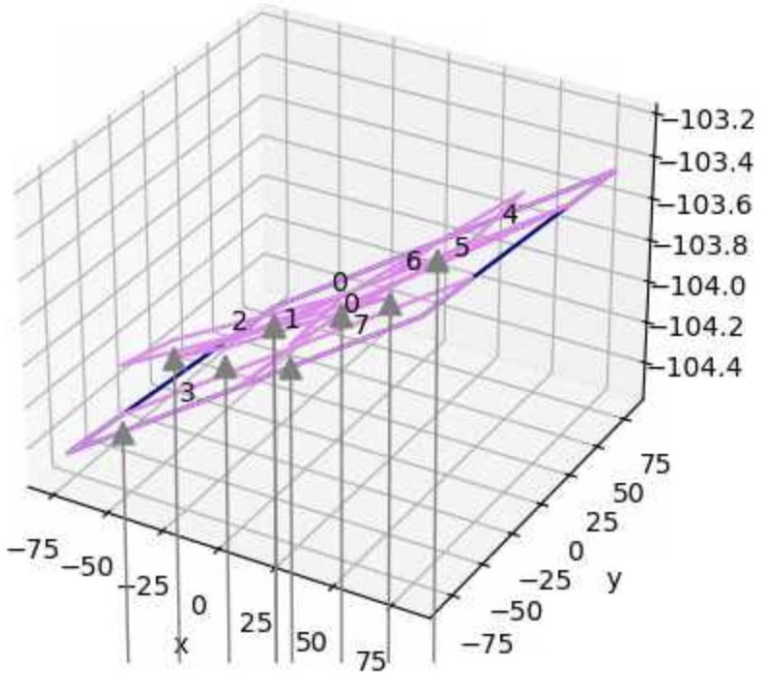
Geometry Issue

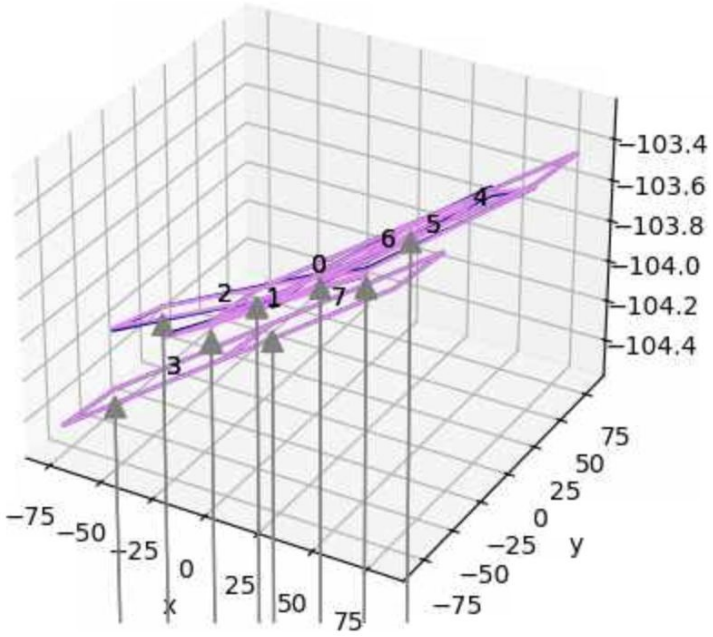
Did refinement cycle, evaluated with `cctbx.xfel.detector_residuals refined.* hierarchy=1`



Split Detector

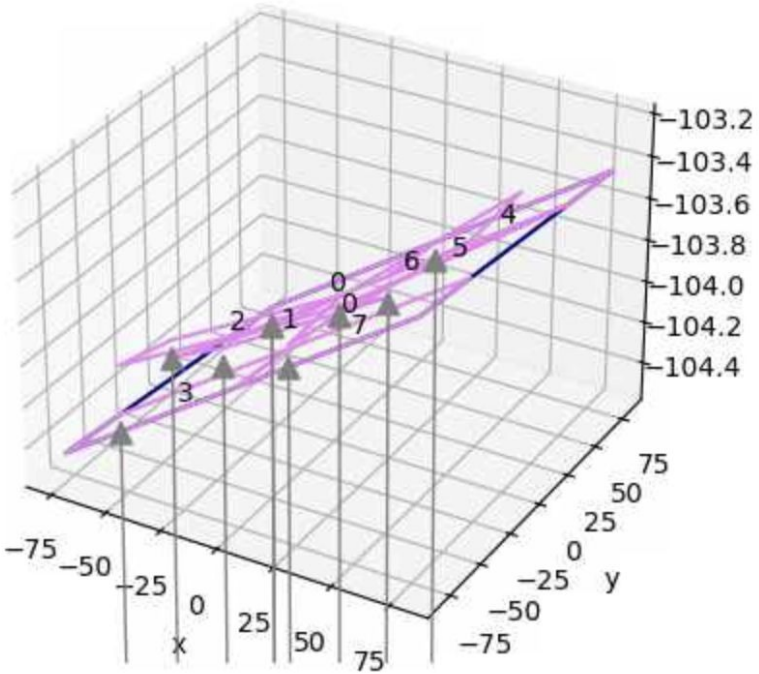
Installed on S3DF, access with `export SPLIT_ESRF_JUNGFRAU=1`

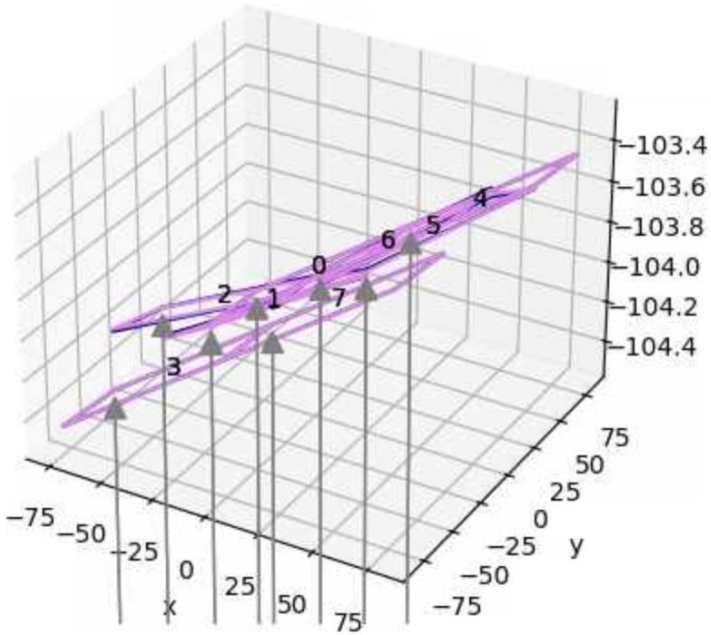




Split Detector

Installed on S3DF, access with `export SPLIT_ESRF_JUNGFRAU=1`



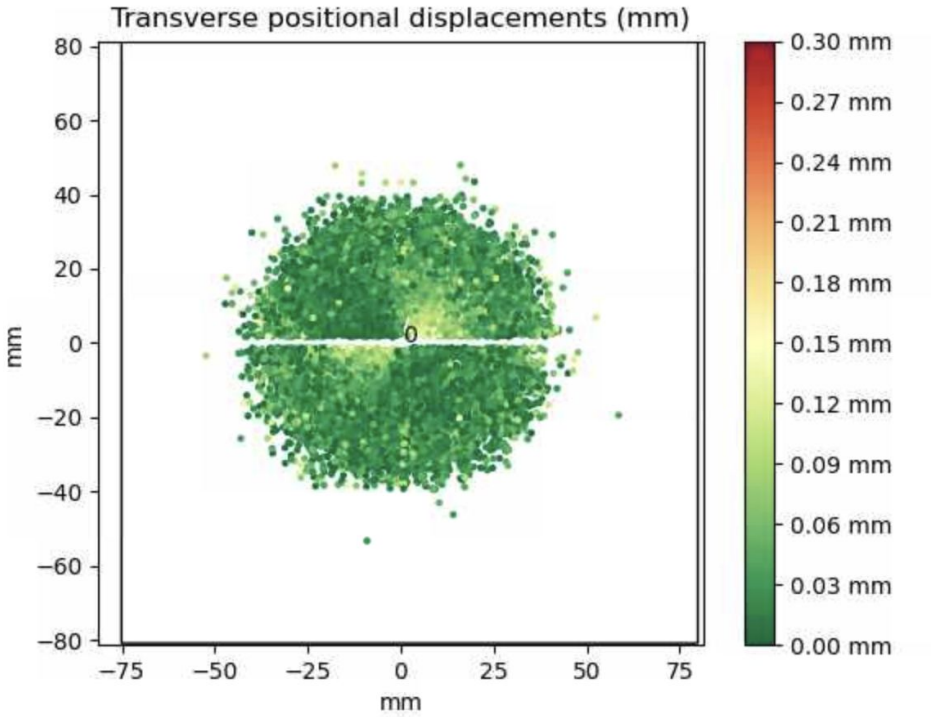


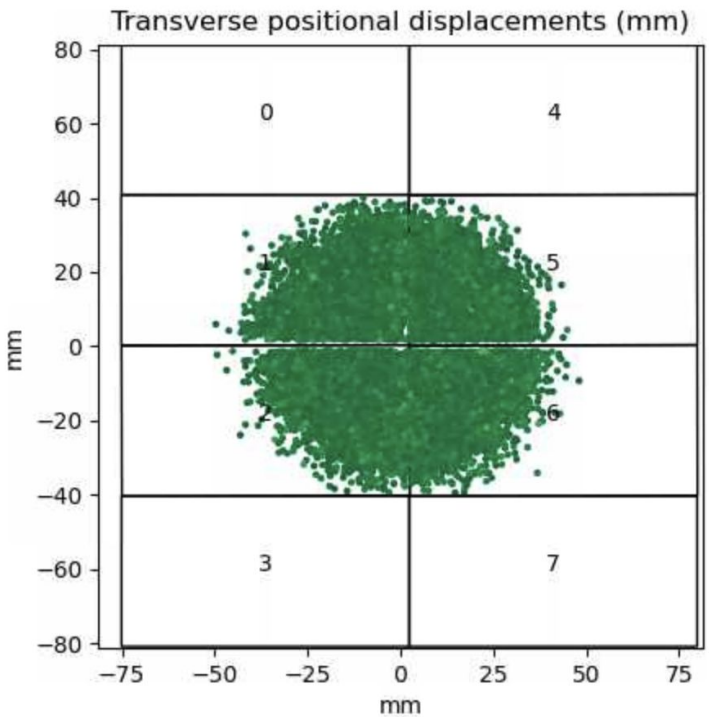
Note: not enough high res spots to refine the outer panels, could try with larger batches or filtering for images with higher res to improve this

Otherwise, can change dmin for refinement

Geometry Fixed

Did refinement cycle, evaluated with `cctbx.xfel.detector_residuals refined.* hierarchy=1`





dials.stills_process

```
input.reference_geometry=/sdf/home/k/kaz42/esrf/0s/run1/m
etrology/reprocess/refine/reprocess/refine/refined.expt
output.composite_output = True
spotfinder.filter.min_spot_size=2
spotfinder.filter.max_separation=4
spotfinder.threshold.dispersion.sigma_strong=2
spotfinder.threshold.dispersion.sigma_background=5
indexing {
  known_symmetry {
    space_group = 19
    unit_cell = 58.03 81.21 110.71 90 90 90
  }
  refinement_protocol.n_macro_cycles = 1
  refinement_protocol.d_min_start=1.5
  stills.indexer=stills
  stills.method_list=fft1d fft3d real_space_grid_search
}
indexing.stills.reflection_subsampling.enable=True
```

If indexing fails, randomly throw out some reflections and try again

```
integration {
  integrator=stills
  profile.fitting=False
  background {
    simple {
      outlier {
        algorithm = null
      }
    }
  }
}
profile {
  gaussian_rs {
    min_spots.overall = 0
}
}
refinement {
  parameterisation {
    beam.fix=all
    detector.fix=all
    detector.hierarchy_level=0
    auto_reduction {
      action=fix
      min_nref_per_parameter=1
    }
  treat_single_image_as_still=True
  }
  reflections {
    outlier.algorithm=null
    weighting_strategy.override=stills
    weighting_strategy.delpsi_constant=1000000
```

dials.combine and dials.refine: under ~10k indexed

```
idx-0024_indexed.refl     idx-0054_integrated.expt  idx-0084_integrated.refl  idx-0114_refined.expt
idx-0024_integrated.expt  idx-0054_integrated.refl  idx-0084_refined.expt     idx-0115_indexed.refl
idx-0024_integrated.refl  idx-0054_refined.expt     idx-0085_indexed.refl     idx-0115_integrated.expt
idx-0024_refined.expt     idx-0055_indexed.refl     idx-0085_integrated.expt  idx-0115_integrated.refl
idx-0025_indexed.refl     idx-0055_integrated.expt  idx-0085_integrated.refl  idx-0115_refined.expt
idx-0025_integrated.expt  idx-0055_integrated.refl  idx-0085_refined.expt     idx-0116_indexed.refl
idx-0025_integrated.refl  idx-0055_refined.expt     idx-0086_indexed.refl     idx-0116_integrated.expt
idx-0025_refined.expt     idx-0056_indexed.refl     idx-0086_integrated.expt  idx-0116_integrated.refl
idx-0026_indexed.refl     idx-0056_integrated.expt  idx-0086_integrated.refl  idx-0116_refined.expt
idx-0026_integrated.expt  idx-0056_integrated.refl  idx-0086_refined.expt     idx-0117_indexed.refl
idx-0026_integrated.refl  idx-0056_refined.expt     idx-0087_indexed.refl     idx-0117_integrated.expt
idx-0026_refined.expt     idx-0057_indexed.refl     idx-0087_integrated.expt  idx-0117_integrated.refl
idx-0027_indexed.refl     idx-0057_integrated.expt  idx-0087_integrated.refl  idx-0117_refined.expt
idx-0027_integrated.expt  idx-0057_integrated.refl  idx-0087_refined.expt     idx-0118_indexed.refl
idx-0027_integrated.refl  idx-0057_refined.expt     idx-0088_indexed.refl     idx-0118_integrated.expt
idx-0027_refined.expt     idx-0058_indexed.refl     idx-0088_integrated.expt  idx-0118_integrated.refl
idx-0028_indexed.refl     idx-0058_integrated.expt  idx-0088_integrated.refl  idx-0118_refined.expt
idx-0028_integrated.expt  idx-0058_integrated.refl  idx-0088_refined.expt     idx-0119_indexed.refl
idx-0028_integrated.refl  idx-0058_refined.expt     idx-0089_indexed.refl     idx-0119_integrated.expt
idx-0028_refined.expt     idx-0059_indexed.refl     idx-0089_integrated.expt  idx-0119_integrated.refl
```

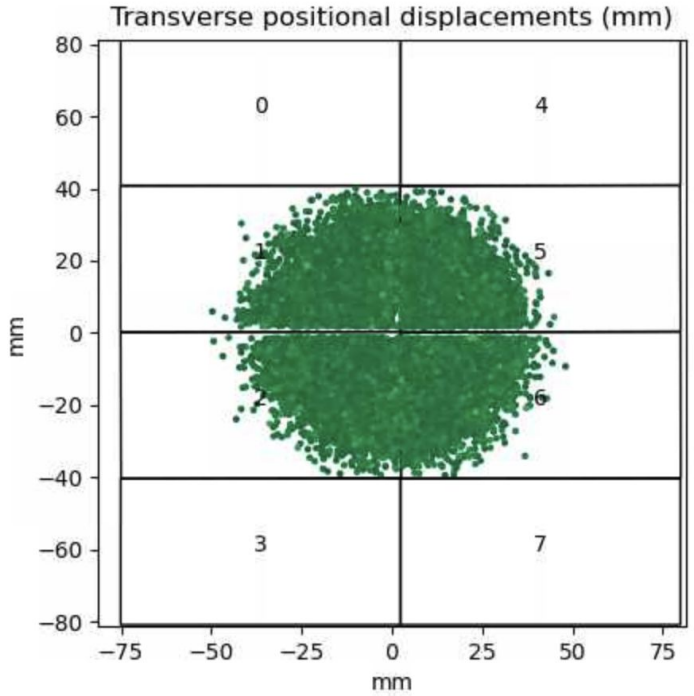
```
dials.combine_experiments ../idx-0???_indexed.refl ../idx-0???_refined.expt
reference_from_experiment.detector=0 reference_from_experiment.beam=0
```

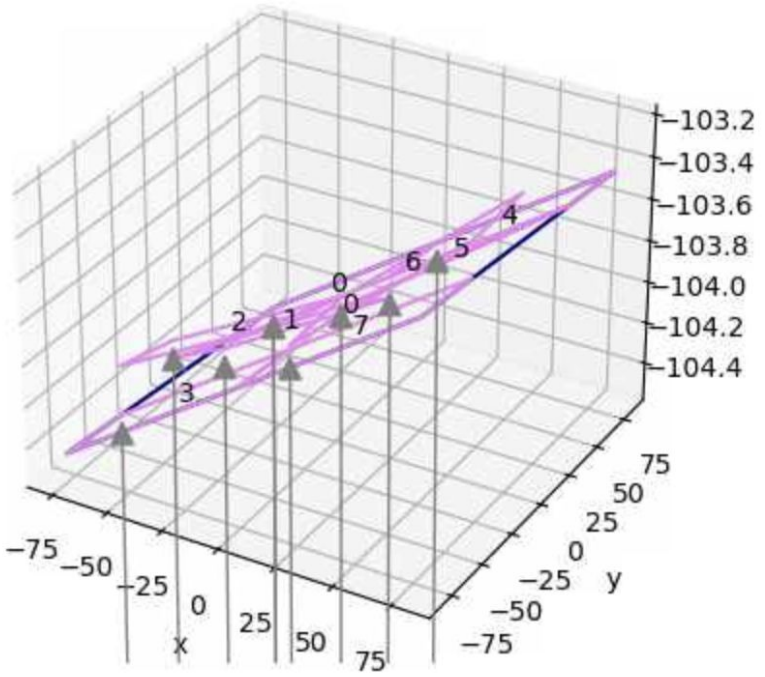
dials.combine and dials.refine: under ~10k indexed

```
refinement {
  parameterisation {
    auto_reduction {
      min_nref_per_parameter = 3
      action = fail fix *remove
    }
    beam {
      fix = *all in_spindle_plane out_spindle_plane wavelength
    }
    detector {
      fix_list = Group1Tau1
      hierarchy_level = 1
    }
  }
  refinery {
    engine = SimpleLBFGS LBFGScurvs GaussNewton LevMar *SparseLevMar
  }
  reflections {
    outlier {
      algorithm = null auto mcd tukey *sauter_poon
      separate_experiments = False
      separate_panels = True
    }
  }
}
```

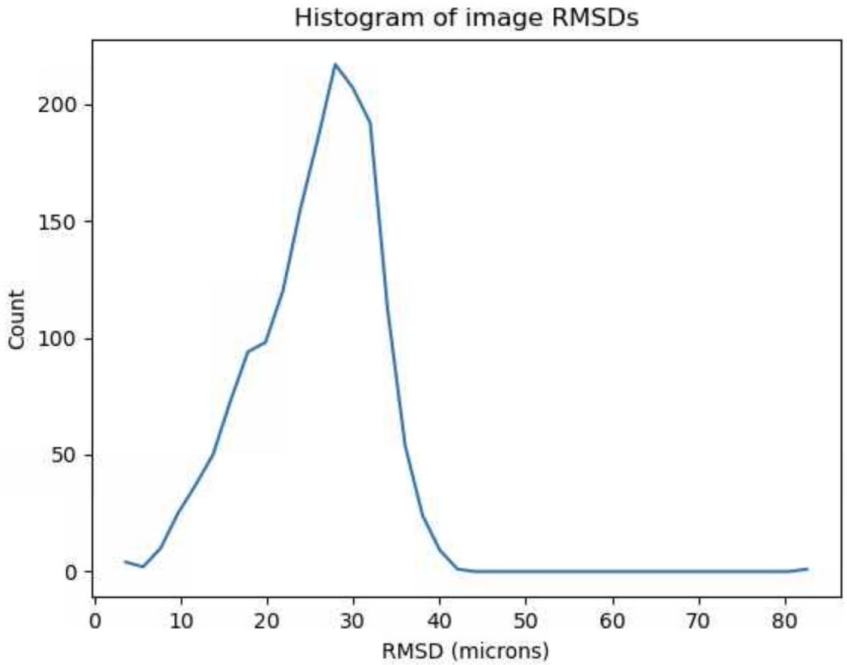
Refine quadrants

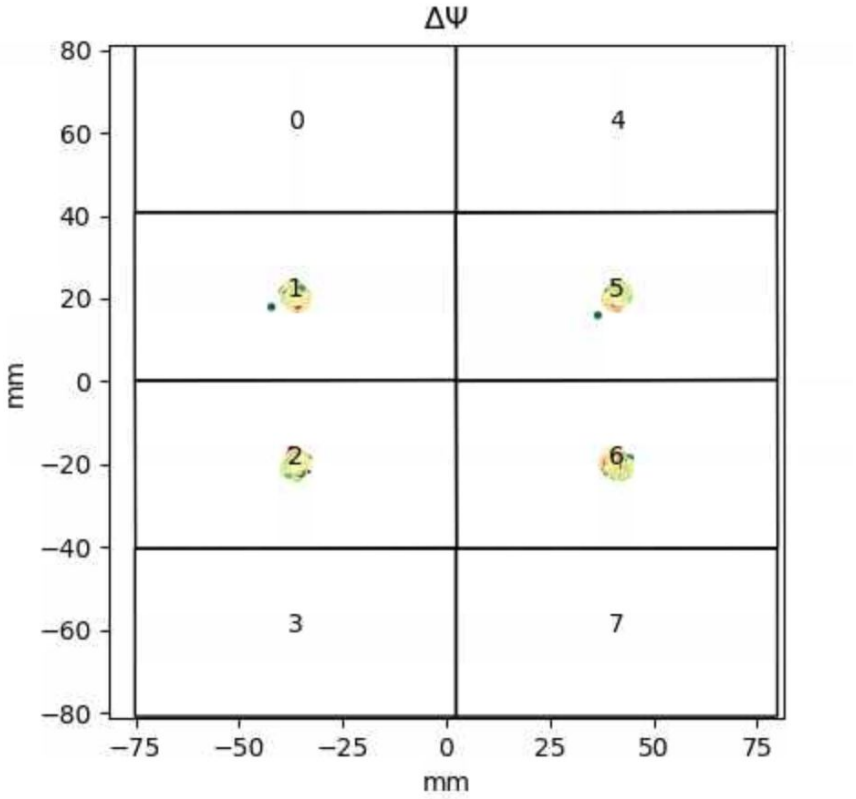
Quality check - Detector



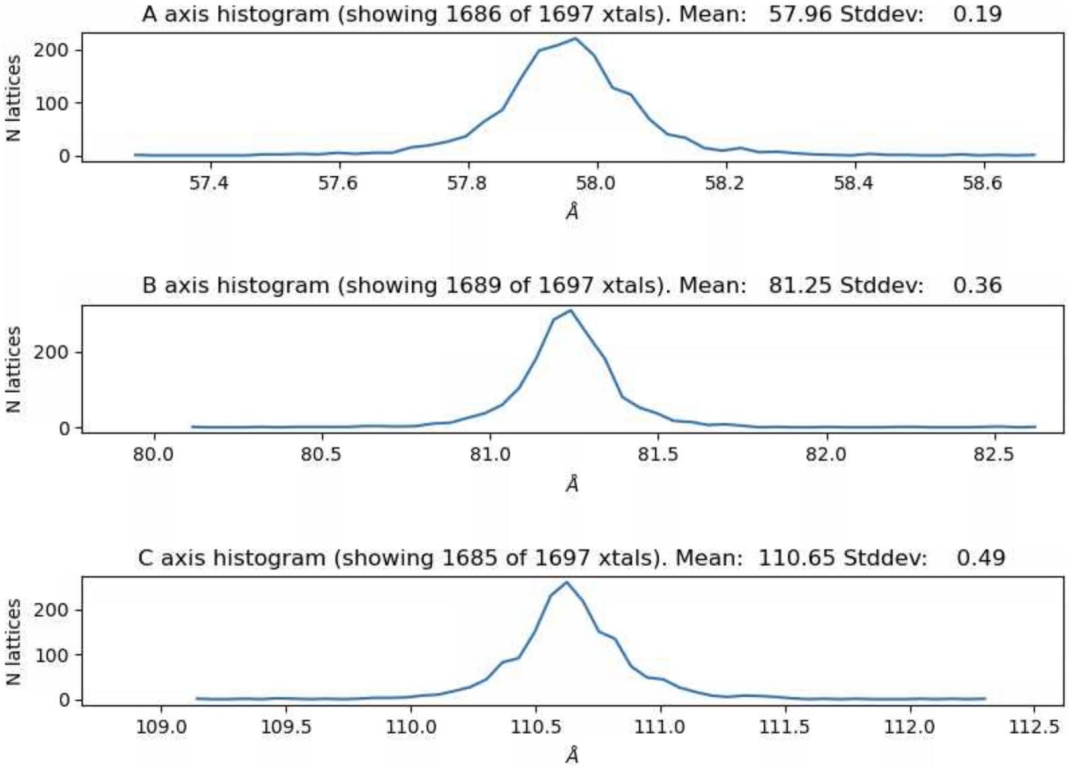


Quality check - Detector





Quality check - unit cell, check for change

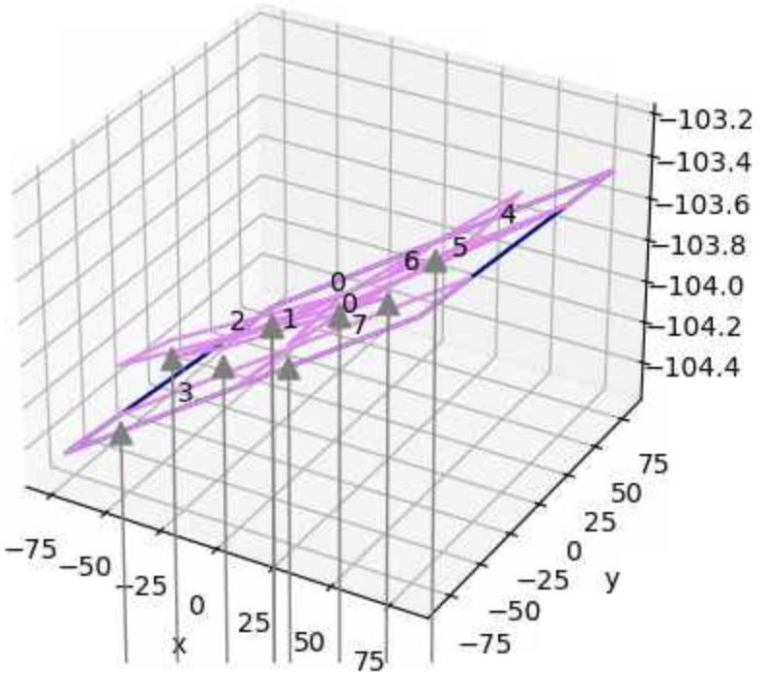


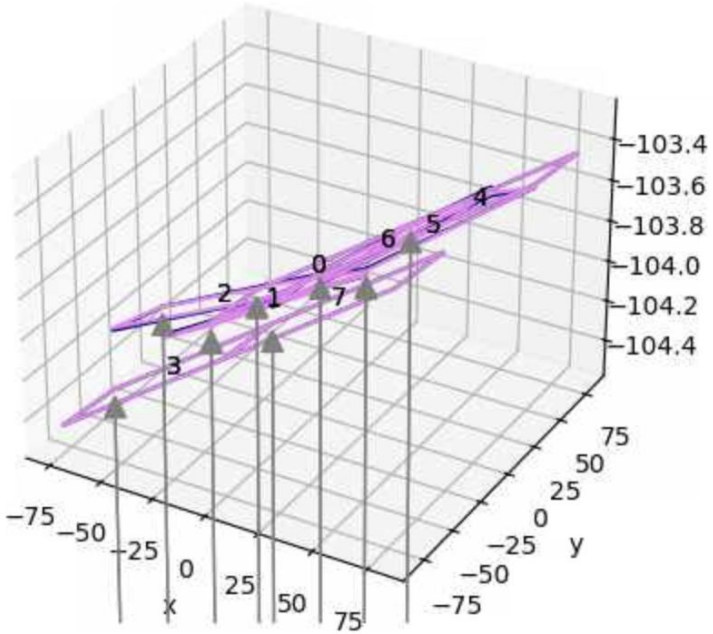
dials.stills_process re-run

```
input.reference_geometry=/sdf/home/k/kaz42/esrf/0s/run1/metrology/reprocess/refine/reprocess/refine/refined.expt
output.composite_output = True
spotfinder.filter.min_spot_size=2
spotfinder.filter.max_separation=4
spotfinder.threshold.dispersion.sigma_strong=2
spotfinder.threshold.dispersion.sigma_background=5
indexing {
  known_symmetry {
    space_group = 19
    unit_cell = 58.03 81.21 110.71 90 90 90
  }
  refinement_protocol.n_macro_cycles = 1
  refinement_protocol.d_min_start=1.5
  stills.indexer=stills
  stills.method_list=fft1d fft3d real_space_grid_search
}
indexing.stills.reflection_subsampling.enable=True
```

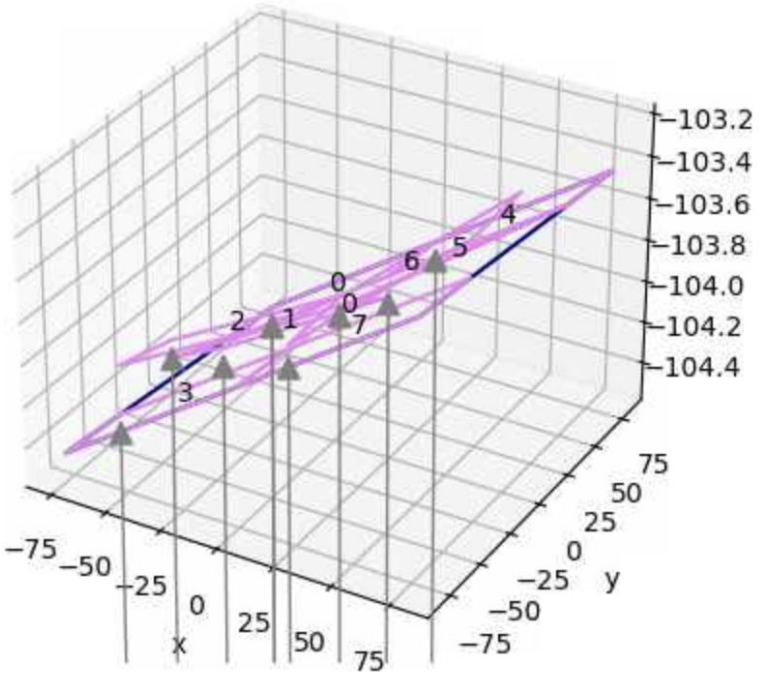
```
integration {
  integrator=stills
  profile.fitting=False
  background {
    simple {
      outlier {
        algorithm = null
      }
    }
  }
}
profile {
  gaussian_rs {
    min_spots.overall = 0
  }
}
}
refinement {
  parameterisation {
    beam.fix=all
    detector.fix=all
    detector.hierarchy_level=0
    auto_reduction {
      action=fix
      min_nref_per_parameter=1
    }
  treat_single_image_as_still=True
  }
  reflections {
    outlier.algorithm=null
    weighting_strategy.override=stills
    weighting_strategy.delpsi_constant=1000000
```

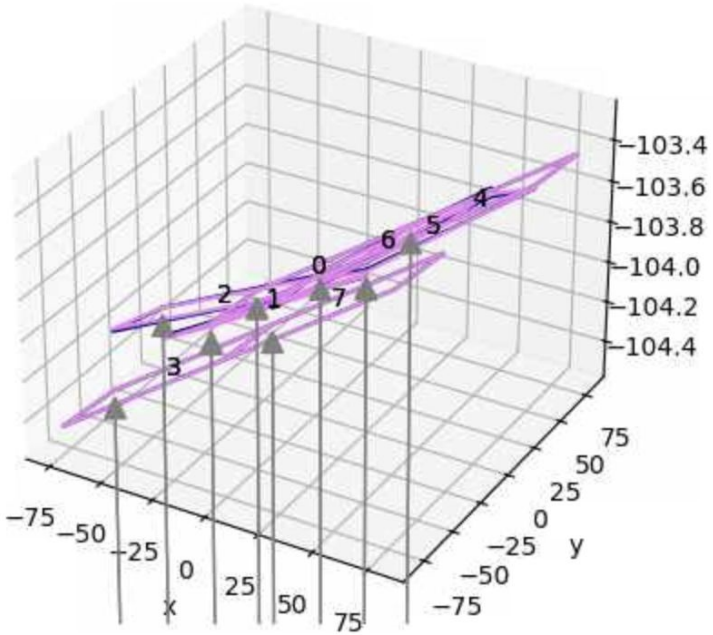
Quality check - cycle through to convergence





Quality check - cycle through to convergence





For a novel geometry (aka new position on flow cell) - would need 4-5 cycles
For repeat run in same location - would need 1-2 cycles

Final dials.still_process

```
dials.stills_process
/sdf/data/lcls/ds/prj/prjesrf0125/hdf5/Hsp31_foil/unfiltered/run_01_ssx_foil_co
llection/aggregated/master_unfiltered_01-Hsp31_foil_dense.h5 process.phil
mp.nproc=120 multiple_lattice_search.max_lattices=2
```

dials.combine and dials.refine: over ~10k
Why is this a problem

dials.refine tries to update detector and crystal

So as the number of indexed patterns increases, have way too many parameters to handle

dials.combine and dials.refine: over ~10k
The long way - challenging outside the GUI

```
dials.combine_experiments ../idx-0???_indexed.refl ../idx-0???_refined.expt
reference_from_experiment.detector=0 reference_from_experiment.beam=0 max_batch_size=2000

dials.refine combined_004.{expt,refl} refine.phil output.experiments=refined_004.expt
output.reflections=refined_004.refl

cctbx.xfel.recompute_mosaicity refined_019* output.experiments=refinedmos_019.expt
output.reflections=refinedmos_019.refl

mpirun n -1 cctbx.xfel.mpi_integrate input.path=refinedmos_0* integrate.phil input.experiments_suffix=.expt
input.reflections_suffix=.refl integration.integrator=stills
```

dials.combine and dials.refine: over ~10k
The long way - challenging outside the GUI

```
dials.combine_experiments ../idx-0???_indexed.refl ../idx-0???_refined.expt
reference_from_experiment.detector=0 reference_from_experiment.beam=0 max_batch_size=2000

dials.refine combined_004.{expt,refl} refine.phil output.experiments=refined_004.expt
output.reflections=refined_004.refl

cctbx.xfel.recompute_mosaicity refined_019* output.experiments=refinedmos_019.expt
output.reflections=refinedmos_019.refl

mpirun n -1 cctbx.xfel.mpi_integrate input.path=refinedmos_0* integrate.phil input.experiments_suffix=.expt
input.reflections_suffix=.refl integration.integrator=stills
```

Problem: tedious and cannot do further cycles of geometry refinement because stills_process only runs on the master so cannot specify unique geometries within the run

dials.combine and dials.refine: over ~10k
Kevin/Kara hack

Problem: large number of indexed patterns gives too many parameters

dials.combine and dials.refine: over ~10k
Kevin/Kara hack

Problem: large number of indexed patterns gives too many parameters

Fix: have dials.refine only update detector and let dials.stills_process refine crystal

Pattern becomes:

1. Stills.process
2. Combine and dials.refine, refine detector (fix crystals)
3. Stils.process, refine crystals (fix detector)
4. Combine and dials.refine, refine detector
5. Repeat until convergence

dials.combine and dials.refine: over ~10k Kevin/Kara hack

```
refinement {
  parameterisation {
    auto_reduction {
      min_nref_per_parameter = 3
      action = fail fix *remove
    }
    beam {
      fix = *all in_spindle_plane out_spindle_plane wavelength
    }
    detector {
      fix_list = Group1Tau1
      hierarchy_level = 1
    }
    crystal.fix = all
  }
  refinery {
    engine = SimpleLBFGS LBFGScurvs GaussNewton LevMar *SparseLevMar
  }
  reflections {
    outlier {
      algorithm = null auto mcd tukey *sauter_poon
      separate_experiments = False
      separate_panels = True
    }
  }
}
```

Merge: mpirun -n 120 --oversubscribe cctbx.xfel.merge merge.phil

```
input.path=/sdf/home/k/kaz42/esrf/0s_redo/run1/metrology/rep
rocess/refine/reprocess/refine/reprocess/refine/reprocess/re
fine/converged
input.path=/sdf/home/k/kaz42/esrf/0s_redo/run2/metrology/rep
rocess/refine/reprocess/refine/reprocess/refine/reprocess/re
fine/refine/refine/converged
input.path=/sdf/home/k/kaz42/esrf/0s_redo/run3/metrology/con
verged
input.path=/sdf/home/k/kaz42/esrf/0s_redo/run4/metrology/rep
rocess/refine/converged
input.path=/sdf/home/k/kaz42/esrf/0s_redo/run5_split_all/met
rology/reprocess/refine/converged
input.experiments_suffix=_integrated.expt
input.reflections_suffix=_integrated.refl
#dispatch.step_list=input balance model_scaling modify
filter scale postrefine statistics_unitcell statistics_beam
model_statistics statistics_resolution group
statistics_deltaccint errors_merge statistics_intensity
merge statistics_intensity_cxi
filter.outlier.min_corr=0.1
filter.algorithm=unit_cell
filter.unit_cell.value.relative_length_tolerance=0.03
```

```
select.algorithm=significance_filter
select.significance_filter.sigma=0.1
#select.significance_filter.min_ct=200
#select.significance_filter.max_ct=300
scaling.model=/sdf/home/k/kaz42/esrf/reference_data
/Hsp31_crystfel.pdb
scaling.resolution_scalar=0.96
merging.d_min=1.5
merging.error.model=mm24
merging.merge_anomalous=True
postrefinement.enable=True
statistics.n_bins=20
output.do_timing=True
output.output_dir=/sdf/home/k/kaz42/esrf/0s_redo/me
rge
output.prefix=Hsp31_0s
```

select.significance_filter.min_ct/max_ct: these values are for larger unit cells. For small unit cells, leave out these values and use the defaults. This is an unresolved issue in the program

Statistics: Before/After
split detector/sub-sampling/Kevin hack

Parameter	Before	After
d_min (CC_half)	2.20	2.18
Total patterns after filtering during merge	57614	62906
Completeness	99.99 (100)	99.99 (100)
Merged <I/sig(I)>	4.895 (0.821)	8.321 (1.319)
Multiplicity	86.43 (28.43)	138.39 (47.02)
CChalf (highest)	0.9959 (0.3258)	0.9974 (0.3358)