Quick start, calibration, detector alignment, peak finder, scripts

Content

Analysis of cxif5315

Quick Start

These commands setup the psana environment, get the code, analyze 5 events with 2 separate hit finders in the equator/arc regions, display the found hits, and save the resulting hit information to a "small-data" text file (similar to cheetah):

```
ssh -X pslogin.slac.stanford.edu
ssh -X psana
source /reg/g/psdm/etc/ana_env.sh
cp -r /reg/g/psdm/tutorials/cxi/cxif5315 .
cd cxif5315
python psana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.py
```

This command reads the resulting small-data text file and can select specific events for reanalysis. **The name of the txt file changes for different runs**, so insert the name of your .txt file in the work/ subdirectory:

```
python readSmallData.py exp=cxif5315:run=169 work/PutNameOfYourTxtFileHere.txt
```

This same script can also read cheetah-produced text files, and only analyze specific events

Data

- exp=cxif5315:run=147, 198 - dark
- exp=cxif5315:run=162 - water sample
- exp=cxif5315:run=169 - data sample
- Detector: CxiDs2.0:Cspad.0

Calibration files

Use my local calibration store

```
[psana]
calib-dir = /reg/neh/home1/dubrovin/LCLS/rel-mengning/calib
```

Dark files were obtained and deployed using Calibration Management Tool

exp=cxif5315:run=147 dark average, rms, and average difference between runs 147-198:

Difference of average dark images between runs 147-198 is small, 0.3 ADU, difference rms=0.6 ADU, this proves that CxiDs2.0:Cspad.0 is stable during 50 runs.

Image array averaging

Image array averaging is useful procedure for many purpose;

- for dark,
- for background,
- for statistics of good/bad pixels and associated mask, etc.

1) Download configuration file psana-cxif5315-cspad-ds2-NDArrAverage.cfg

2) To get average/rms/max over 10000 events data of run 169 use command:

```
psana -c psana-cxif5315-cspad-ds2-NDArrAverage.cfg -n 1000 exp=cxif5315:run=169
```

This command produces files with arrays of size (32x185x388)

- cspad-ndarr-ave-cxif5315-r0169.dat
- cspad-ndarr-rms-cxif5315-r0169.dat
- cspad-ndarr-max-cxif5315-r0169.dat

3) To get average/rms/max over 1000 events (skipping 5000) data for water sample of run 162 use command:

```
psana -c psana-cxif5315-cspad-ds2-NDArrAverage.cfg -s 5000 -n 1000 exp=cxif5315:run=162
```

This command produces files with arrays of size (32x185x388)

- cspad-ndarr-ave-cxif5315-r0162.dat - file for averaged background
- cspad-ndarr-rms-cxif5315-r0162.dat
- cspad-ndarr-max-cxif5315-r0162.dat

Geometry alignment

In accordance with CSPAD Alignment, use geometry file:

`/reg/g/psdm/detector/alignment/cspad/calib-cxi-ds2-2015-01-20/calib/CsPad::CalibV1/CxiDs2.0:Cspad.0/geometry/1-end.data`

and array with image from file: cspad-ndarr-ave-cxif5315-r0169.dat

⚠ Apparently, this geometry file had wrong indexes associated with quads (cable swap) that gives misaligned 2x1s in quads. This problem should be fixed in the file

```
/reg/g/psdm/detector/alignment/cspad/calib-cxi-ds2-2015-01-20/calib/CsPad::CalibV1/CxiDs2.0:Cspad.0
/geometry/geo-cxif5315-r0169-swap-tuned.data
```

Tuning of geometry can be done with Detector alignment tool running command:

```
geo -i cspad-ndarr-ave-cxif5315-r0169.dat -g geo-cxif5315-r0169-swap-tuned.data
```

Thus obtained geo-cxif5315-r0169-swap-tuned.data gives image, radial, and spectral histograms for averaged array:

or see image array with maximal intensities by the command

```
geo -i cspad-ndarr-max-cxif5315-r0169.dat -g geo-cxif5315-r0169-swap-tuned.data
```

looks like: ,

and similar for water background from run 162:

ROI Masks

Using image array in cspad-ndarr-max-cxif5315-r0169.dat and geometry file geo-cxif5315-r0169-swap-tuned.data, one can generate masks using the Mask Editor embedded in Calibration Manager.

Run command `calibman`, select tab `ROI` and performing all steps of the mask creation procedure, one can draw ROI on image, create the mask for this image, convert this mask to ndarray, and use this ndarray for data processing. This ndarray can be used later for image reconstruction with the same geometry file.

For data processing of exp=cxif5315:run=169 two masks were produced for arc and equatorial regions:

Files with mask ndarray for arc and equator regions:

- roi_mask_nda_arc.txt
- roi_mask_nda_equator.txt

Peak finder

Who is doing what

Peak finding algorithm is based on psana module ImgAlgos::NDArrDropletFinder with preceded CSPAD ndarray producer CSPadPixCoords::CSPadNDArrProducer and its calibration ImgAlgos::NDArrCalib. Detector geometry information is provided by the module ImgAlgos::PixCoordsProducer. Configuration file psana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.cfg (download) defines parameters for these modules. Two instances of modules are used in order to process two region of interests for Arc and Equator.

For each event psana modules are executed first and save the list of found peaks in the event store. Then, python script psana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.py (download) works with lists of peaks, performs additional selection, plots image with peaks, and saves selected peaks in the file.

Latest version of packages

On 2015-04-16:

Release `ana-0.14.2` is up to date.

Until `ana-current` is `ana-0.14.1`, one has to use `sit_setup` script with release tag parameter:

```
sit_setup ana-0.14.2
```

External files

In order to run examples create directory with some arbitrary name `my_analysis` and `work` directory in it:

```
mkdir my_analysis
mkdir my_analysis/work
cd    my_analysis
sit_setup ana-0.14.2
```

Download files in `my_analysis` directory:

- configuration file for ndarray averaging psana-cxif5315-cspad-ds2-NDArrAverage.cfg (download)
- configuration file for droplet(peak) finder psana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.cfg (download)
- python script for droplet(peak) finder for droplet(peak) finderpsana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.py (download)

Configuration script expects a few files in the local directory `work/`; two files with masks and file with a shape of background needs to be downloaded in the my_analysis/work directory:

- roi_mask_nda_arc.txt
- roi_mask_nda_equator.txt
- cspad-ndarr-ave-cxif5315-r0162.dat

In the local or standard `calib/` directory the files for calibration should be available for `geometry, pedestals, pixel_status, common_mode` etc., whatever is going to be used in the modules.

How to run

Use command:

```
python psana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.py
```

Example: get table of peaks for different regions

Image with peaks for separate and both regions can be plotted:

File with table of parameters for found peaks has an unique name like `work/peaks-2015-04-14-15:57:44-cxif5315-r0169.txt` and contains records with peak parameters:

```
# Exp     Run  Event  Date       Time      time(sec)   time(nsec) fiduc   Reg  Seg  Row  Col      Amax
Atot Npix   X(um)   Y(um)
cxif5315  169      3  2015-02-22 02:20:47  1424600447  503058551  104427  equ    1  145   27     165.8
2970.8   70    -224    9450
cxif5315  169      3  2015-02-22 02:20:47  1424600447  503058551  104427  equ   17  155   47     164.0
3130.3   58     506   -9674
cxif5315  169      7  2015-02-22 02:20:47  1424600447  536405573  104439  arc    8   10   20     500.9
10085.8   74  -48264    1186
cxif5315  169      7  2015-02-22 02:20:47  1424600447  536405573  104439  equ    1  161   13     483.6
3238.8   56    1318    7695
cxif5315  169      7  2015-02-22 02:20:47  1424600447  536405573  104439  equ   16  152    8     191.7
3393.9   74   -3801  -32359
cxif5315  169      7  2015-02-22 02:20:47  1424600447  536405573  104439  equ   17   30   69     295.3
2644.8   50    2903  -23418
cxif5315  169      7  2015-02-22 02:20:47  1424600447  536405573  104439  equ   17  167   48     153.0
2723.5   59     618   -8355
cxif5315  169      7  2015-02-22 02:20:47  1424600447  536405573  104439  equ   17  171   32     283.2
3447.1   59   -1139   -7913
cxif5315  169      8  2015-02-22 02:20:47  1424600447  544737897  104442  equ   17  168   47     201.7
2797.8   54     508   -8245
cxif5315  169     12  2015-02-22 02:20:47  1424600447  578091436  104454  equ    1  152   27     159.8
2374.2   58    -222    8681
cxif5315  169     14  2015-02-22 02:20:47  1424600447  594757102  104460  equ    1  153   29     160.8
3308.4   62    -442    8571
...
```

The 1st line in this file is a (commented) header with colon-titles.

Remarks

- Peak finder parameters - all parameters in the peak finder are set without optimization. They need to be tuned in two places;
    1. in the configuration file for two modules `ImgAlgos.NDArrDropletFinder(:Arc/:Equ)`, and
    2. in the python script in calls of `peaks_filter(...)` method.
- ROI for peakfinder is defined by combination of two mechanisms;
    1. mask - ndarray with values 1/0. Pixels masked by "0", are ignored.
    2. a set of rectangular windows on sensors, defined by the parameter windows in module ImgAlgos::NDArrDropletFinder.
- **Peak finder parameters need to be optimized**!

Peaks in Arc region

Selection parameters

Selection of peaks and events is defined in few stages, as listed below.

1. Peak selection parameters in module ImgAlgos.NDArrDropletFinder:

```
[ImgAlgos.NDArrDropletFinder:Arc]
...
threshold_low  = 10
threshold_high = 150
peak_radius    = 3
...
```

2. Peak list selection parameters:

```
nda_peaks_arc = peaks_filter(nda_droplets_arc, atot_thr=2000, npix_thr=20, npeaks_min=1, npeaks_max=10)
```

3. Peak/event selection parameters at peak-list file readout

```
def peakIsSelected() :
    """Apply peak selection criteria to each peak from file
    """
    amax, atot, npix, x, y, r, phi = sp.amax, sp.atot, sp.npix, sp.x, sp.y, sp.r, sp.phi
    if amax<150  : return False
    if atot<2500 : return False
    if npix<30   : return False
    if r<434     : return False
    if r>444     : return False
    if phi<150 and phi>-150 : return False
    return True
```

Few sets of parameters listed in table were tested in order to justify selection parameters.

Parameter	Sel.1 (rpeak5)	Sel.2 (rpeak7)	Sel.3 (rpeak3)
in module NDArrDropletFinder:Arc			
threshold_low	10	10	10
threshold_high	100	100	150
peak_radius	5	7	3
in method peaks_filter(...)			
atot_thr	2000	2000	2000
npix_thr	20	30	20
npeaks_min	1	1	1
npeaks_max	10	20	10
Save selected peaks in file			

Process peaks from file and use selector method `peakIsSelected()`:

Parameter	Sel.1 (rpeak5)	Sel.2 (rpeak7)	Sel.3 (rpeak3)
Process peaks from file selector in method peakIsSelected()			
amax	150	150	150
atot	2500	2500	2500
npix	20	40	20
rmin	434	434	434
rmax	444	444	444
phi	<-150 & >150	<-150 & >150	<-150 & >150

Plots for selection parameters

Distribution for selection parameters are shown below. The 1st group of plots is accumulated for all peaks saved in the file with list of peaks:

- phi - azimuths angle [degree] of the peak, evaluated as `atan2(y,x)`
- Amax - maximal pixel intensity [ADU]
- Atot - total peak intensity [ADU] for all pixels at the distance in x and y less or equal than `peak_radius` (peak region)
- Npix - number of pixels in the peak region with intensity above `threshold_low`
- r - peak radius, evaluated as sqrt(x*x + y*y)
- Npeaks - number of peaks in the arc region after peak-finder after

Two other plots were accumulated after method `peakIsSelected()` is applied

- Npeaks selected - number of peaks in the event
- Distance - between two peaks [pixels] in the event which has two selected peaks only

Sel.1 (rpeak=5):

These distributions were obtained for "reasonable-by-eye" selection criteria. Now the question is how optimal the selection parameters are?

Sel.2 (rpeak=7):

These plots look noisy, because it was allowed to save up to 20 peaks in the event.

Sel.3 (rpeak=3):

Tight parameters in this case reduce a number of signal events in the Distance distribution.

Optimal parameters

- peak_radius = 3, 5, 7 - or peak region sixe 7x7, 11x11, 15x15, respectively does not show significant influence on figure of merit (FOM) distance distribution. Keep 5 as golden-medium.
- amax and atot show significant increase in number of peaks for <150 and <2500, respectively. These selection criteria do not decrease a signal peak in FOM. To be safe we may still pre-select peaks with amax<100 and atot<2000 and then apply tight selection later.
- npix -it is safe to require >20 or >30.
- r - chosen range 434 r 444 looks optimal
- npeaks_max is less than 6 after selection. It looks safe to constrain npeaks_max = 10. Used 20 adds too much noise.

So far original Selection 1 looks optimal.

Peaks in Whiskers region

Parameters

Set in psana.cfg file peak selection parameters pretty much the same as in Arc region

```
[ImgAlgos.NDArrDropletFinder:Whi]
source         = DetInfo(CxiDs2.0:Cspad.0)
key            = nda_clb
key_droplets   = nda_droplets_whi
key_smeared    = nda_sme_whi
threshold_low  = 10
threshold_high = 150
sigma          = 0
smear_radius   = 3
peak_radius    = 3
low_value      = 0.
mask           = work/roi_mask_nda_whi.txt
masked_value   = 0
windows =  1 0 185 0 388 \
          17 0 185 0 388
```

save peaks in file without additional selector.

Events

Event 03: - normal, single hit event

Event 07: - multi-hit event

Event 37: - multi-hit event

Left plot - peak phi<0, right plot for phi>0.

⚠ In the matrix frame x-axis points down...

Distributions for peak parameters without selection

Number of peaks

Peak azimuth angle

Peak radius

Distributions for >1-peak event in each region

Average over peaks azimuth angle

Difference between maximal and minimal values of peak azimuth angle

Standard deviation of azimuth angle spread

Combined distributions for two regions

Difference between averaged azimuth angles in two regions

Orientation angle

Remarks

1. Peak radius equal to 3 or 5 pixel with low and high thresholds 10 and 150ADU, respectively, work fine.
2. More than 90% events have more than 2 peaks in each side of "whiskers" region. Constrains on number of peaks >1 in each region should be used to select good single-hit events.
3. Constrain on `phi_std<0.5` or `phi_ptp<1` can be used to select good single-hit events.
4. Average azimuth angle can be used as a measure of event orientation.

Peaks in Equatorial region

Peak-finder parameters

```
[ImgAlgos.NDArrDropletFinder:Equ]
source         = DetInfo(CxiDs2.0:Cspad.0)
key            = nda_clb
key_droplets   = nda_droplets_equ
key_smeared    = nda_sme_equ
threshold_low  = 10
threshold_high = 150
sigma          = 0
smear_radius   = 3
peak_radius    = 3
low_value      = 0.
mask           = work/roi_mask_nda_equ.txt
masked_value   = 0
windows =  0 0 185 0 388 \
           1 0 185 0 388 \
           3 0 185 0 388 \
           8 0 185 0 388 \
           9 0 185 0 388 \
          11 0 185 0 388 \
          16 0 185 0 388 \
          17 0 185 0 388 \
          19 0 185 0 388 \
          24 0 185 0 388 \
          25 0 185 0 388 \
          27 0 185 0 388
fname_prefix =
print_bits   = 3
```

Distribution of parameters after peak-finder

Where additional peak selection parameters in the python script:

```
    if npix>42   : return False
    if r<100     : return False
    if r>460     : return False
```

Combined analysis for Arc and Equ regions

Peak finding in psana

In psana event processing the same peak definition parameters were used for Arc and Equ regions:

```
[ImgAlgos.NDArrDropletFinder:Equ]
...
threshold_low  = 10
threshold_high = 150
peak_radius    = 3
```

All peaks found in the event were saved in the peak table file without additional selection.

Peak selection in file processing

Arc region selection

```
    if atot<2500 : return False
    if npix>42   : return False
    if r<434     : return False
    if r>444     : return False
    if phi<150 and phi>-150 : return False
```

Equ region selection

```
    if npix>42   : return False
    if r<100     : return False
    if r>420     : return False
```

Event selection

```
def eventIsSelected() :
    sp.event_is_selected = False
    if sp.count_arc_pks_sel != 2 : return False
    if sp.count_equ_pks_sel  > 5 : return False
    if sp.count_equ_pks_sel  < 1 : return False
    sp.event_is_selected = True
    return True
```

For events with exactly two peaks in Arc region orientation is evaluated as an average angle: orient = (phi1 + phi2) / 2:

Orientation distribution shows that in many ~2/3 (left and right peaks) of events two peaks are close to each other... Presumably single peak is split for two or more peaks. Further we select clean events from central peak only:

```
abs(orient) < 20 :
```

- azimuth angle of peaks in Equ region

- orientation corrected azimuth angle of peaks in Equ region

- radial distribution of peaks in Equ region

- number of peaks in Equ region (constrain on sp.count_equ_pks_sel is not applied).

Scripts for analysis on 2015-05-18

Current version of scripts with comments is collected here. It is assumed that you work on one of psana nodes which has access to data, for example:

```
ssh -Y psana
cd <your-analysis-directory>
sit_setup
```

Download files in `<your-analysis-directory>`

- psana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.cfg (download) - configuration file for psana. This script sets parameters for a few psana modules,
  - ImgAlgos.PixCoordsProducer - for each run get CSPAD geometry info from "geometry" calibration file and saves it in the psana internal calibration store,
  - CSPadPixCoords.CSPadNDArrProducer - for each event get raw CSPAD data and saves them as [32,185,388] ndarray in the event store,
  - ImgAlgos.NDArrCalib - get raw CSPAD data ndarray from the event store, apply requested intensity corrections (pedestals, common mode, etc.) and saves calibrated ndarray in the event store,
  - ImgAlgos.NDArrDropletFinder:Arc - uses calibrated ndarray to search for peaks in the arc region and saves the list (ndarray) of peaks in the event store,
  - ImgAlgos.NDArrDropletFinder:Equ - the same as above in the equatorial region,

- ImgAlgos.NDArrDropletFinder:Whi - the sabe as above in the whiskers region,
- ImgAlgos.Tahometer - convenience module for performance monitoring.

Beside automatically loaded calibration files (for pedestals, common mode, masks, geometry, etc.) this configuration file uses four external files with region masks and background

- work/cspad-ndarr-ave-cxif5315-r0162.dat - file with ndarray for background subtraction
- work/roi_mask_nda_arc.txt - ndarray with mask for arc region
- work/roi_mask_nda_equ.txt - ndarray with mask for equator region
- work/roi_mask_nda_whi.txt - ndarray with mask for whiskers region
- work/roi_mask_nda_all.txt - ndarray with mask for combined arc, equator, and whiskers regions.

These files are expected to be in the `<your-analysis-directory>/work` directory.

- psana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.py.txt (download) - python script, which used *.cfg file to produce all ndarrays in the event store, gets these ndarrays, does preliminary processing, saves list of peaks in the output text file, draw events (if requested).
  This script can be run interactively or in batch by commands:

```
// interactive command:
psana -c psana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.cfg exp=cxif5315:run=169


// batch job submission command:
bsub -q psfehq -o log-r169.log python psana-cxif5315-r0169-cspad-ds2-NDArrDropletFinder.py
```

  At least 2 parameters need to be adjusted in this script for each mode:

```
    do_plot = True             - should be = False in the batch mode
    #events_max  = 10000000    - for batch mode loop over all events
    events_max  = 1000         - for loop over 1000 events in interactive mode
```

  This job produces text file with all found peaks with name like `work/peaks-2015-05-09-23:52:30-cxif5315-r0169-all.txt`, which can be used for further analysis

- proc-peaks-from-file.py (download) - python script for processing of the text file with list of peaks.
  Input file name and output file path prefix should be set at the end of this script:

```
    procPeaksFromFiles (('work/peaks-2015-05-09-23:52:30-cxif5315-r0169-all.txt',), '2015-05-14-figs-cxif5315/plot-cxif5315-r0169-all')
```

  After that this script can be executed by the command:

```
python proc-peaks-from-file.py
```

  which loops over peaks in file, does analysis, and plots requested histograms at the end.