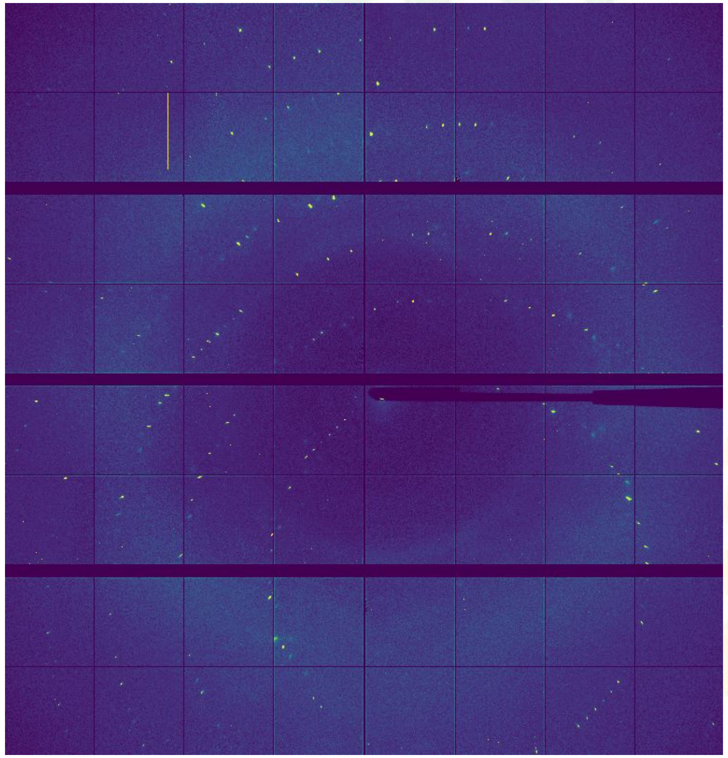
Signal separation for single crystal and serial crystallography

Jérôme Kieffer
ESRF - Grenoble (France)

STREAMLINE has received funding from the European Union’s Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Outline

- Image analysis for single crystal frames
- Lossy data compression
- Peak-finding
- Conclusions



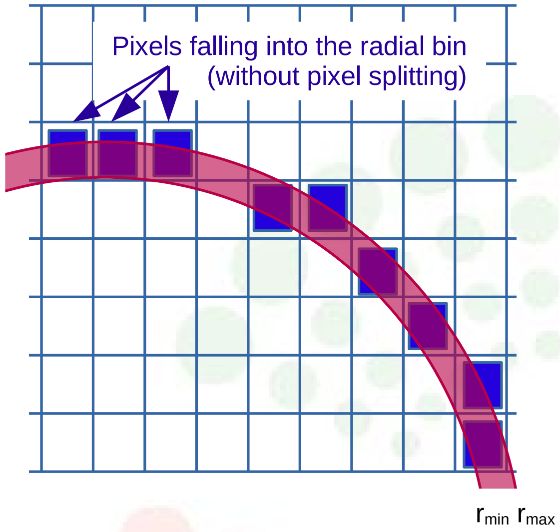
First diffraction image obtained with Jungfrau detector

STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Average pixels along Debye-Scherrer rings

- Pixel intensity needs to be corrected:

$$I_{cor} = \frac{I_{raw} - I_{dark}}{F \cdot \Omega \cdot P \cdot A \cdot I_0} = \frac{signal}{normalization}$$



- Intensity average per ring:
  - Pixel splitting: $c_{i,r}$ is the fraction of pixel *i* in the ring *r*
  - Normalization issue due to polarization, …

→ this is a weighted average: implemented in pyFAI

$$\overline{I}_r = \frac{\sum_{i \in bin_r} c_{i,r} \cdot signal_i}{\sum_{i \in bin_r} c_{i,r} \cdot normalization_i} = \frac{V_{bin_r}}{\Omega_{bin_r}}$$

- Use of accumulators:
  - Simplifies notation
  - Suitable for parallel reduction

$$V = \sum \omega \cdot v = \sum c \cdot signal$$

$$\Omega = \sum \omega = \sum c \cdot normalization$$

$$\Omega\Omega = \sum \omega^2 = \sum c^2 \cdot normalization^2$$

STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Uncertainties in azimuthal integration (1)

- Uncertainties on the average value
  - Called *sem* and reported by pyFAI
  - Not of interest for background evaluation

$$\sigma(\bar{I}_r) = \frac{\sqrt{\sum_{i \in bin_r} c_i^2 \cdot variance_i}}{\sum_{i \in bin_r} c_i \cdot normalization_i} = \frac{\sqrt{VV_r}}{\Omega_r}$$

- Uncertainties on pixel value
  - Called *std* and larger than *sem* *by a factor* √*N*

$$\sigma(I_r) = \sqrt{\frac{\sum_{i \in bin_r} c_i^2 \cdot variance_i}{\sum_{i \in bin_r} c_i^2 \cdot normalization_i^2}} = \sqrt{\frac{VV_r}{\Omega\Omega_r}}$$

- Poisson error model:
  - For all pixels belonging to a common distribution:

    variance = <signal>
  - Usually simplified in:

$$\begin{cases} variance_i = signal_i \\ VV = \sum c^2 \cdot signal \end{cases}$$

$$V = \sum \omega \cdot v = \sum c \cdot signal$$

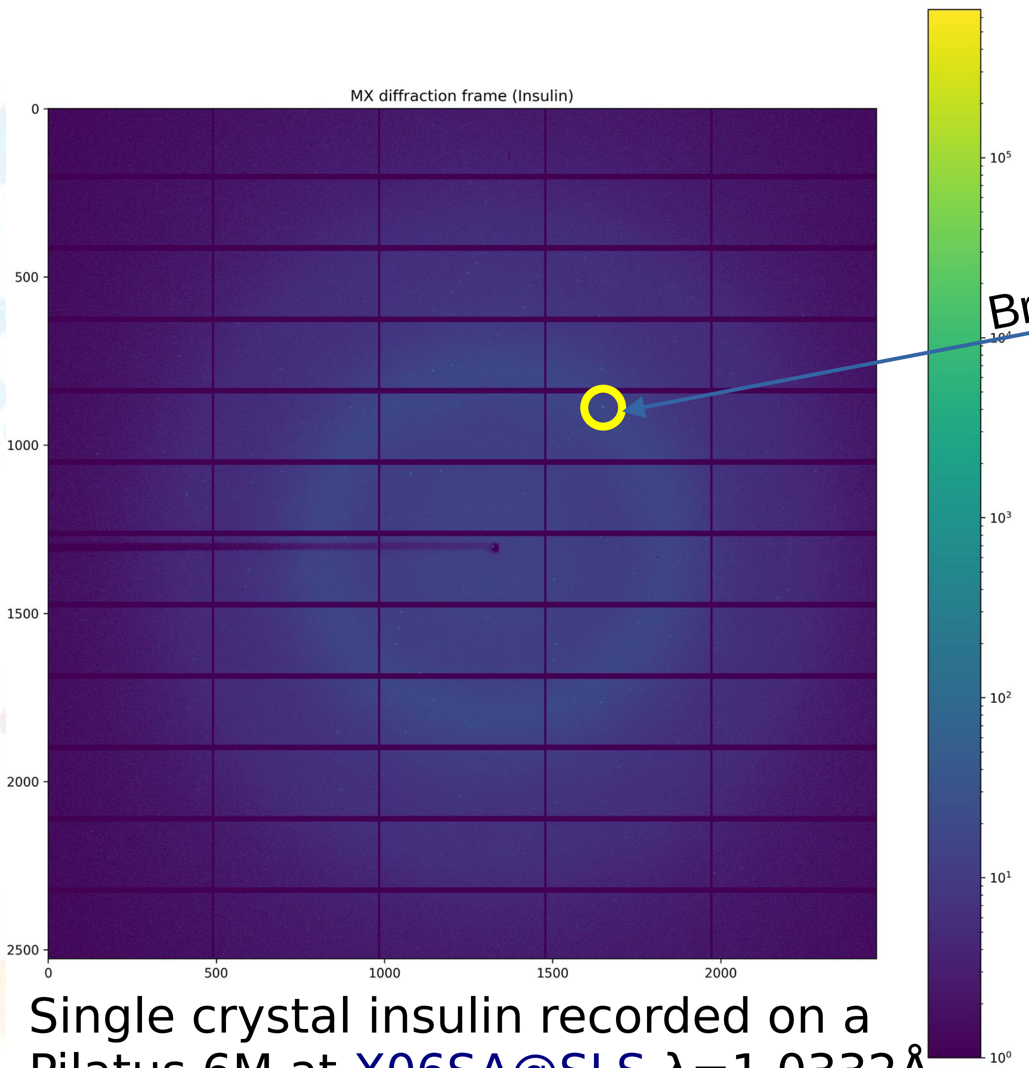
$$\Omega = \sum \omega = \sum c \cdot normalization$$

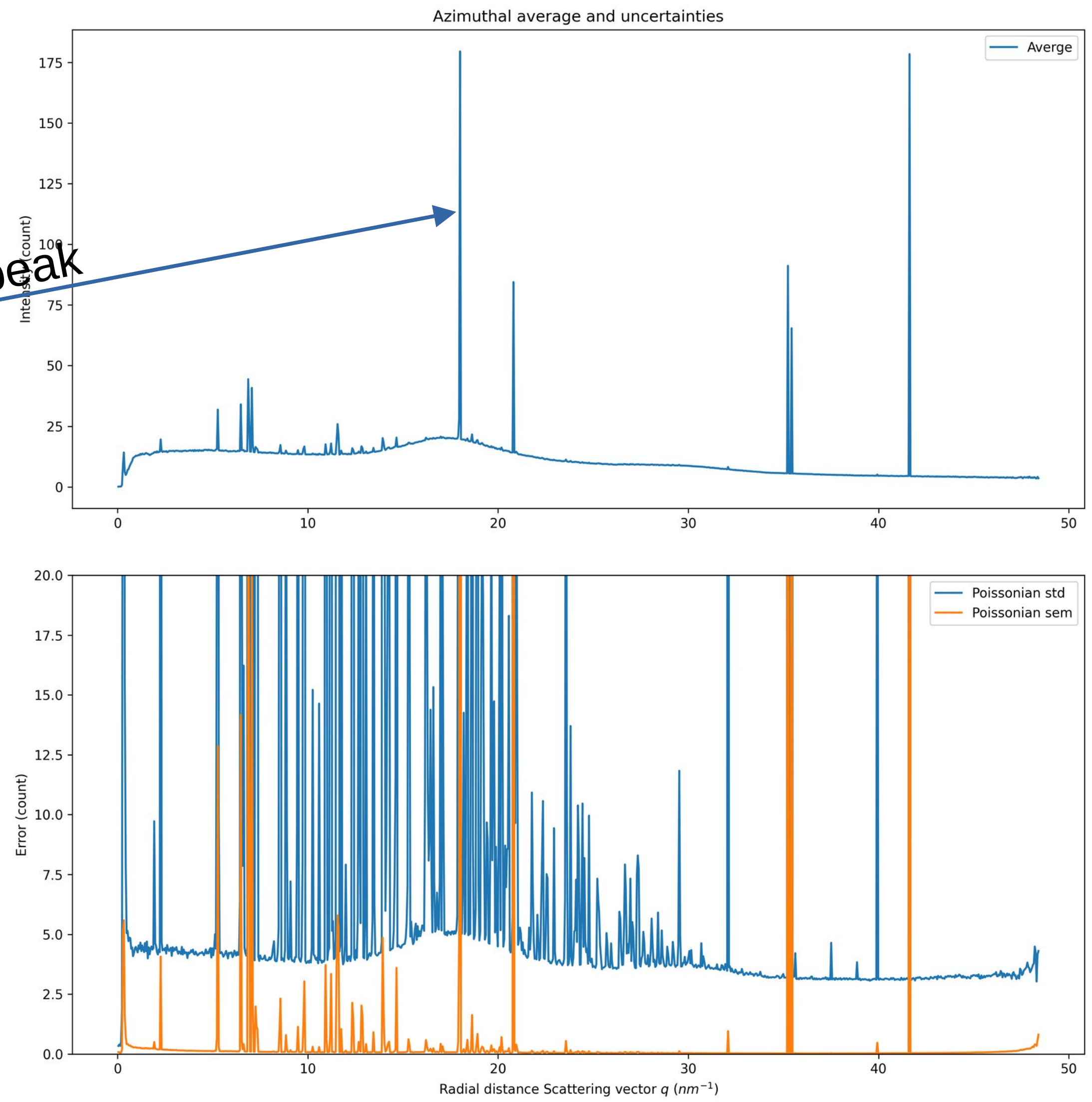
$$\Omega\Omega = \sum \omega^2 = \sum c^2 \cdot normalization^2$$

Erich Schubert and Michael Gertz. 2018.
Numerically Stable Parallel Computation of (Co-)Variance.
SSDBM ’18: 30th Intl. Conf. on Sci, & Statistical DB Mangt.

STREAMLINE has received funding from the European Union’s Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Example on an insulin diffraction frame:

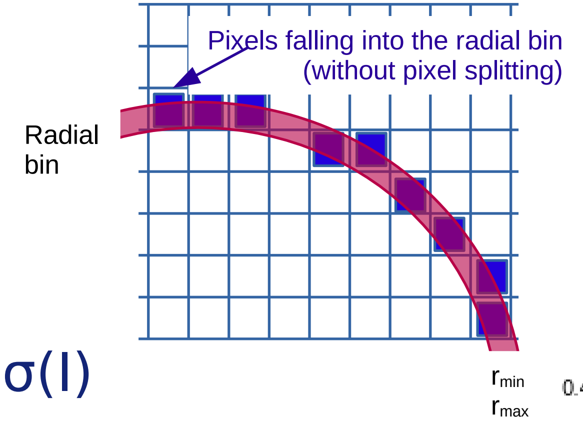




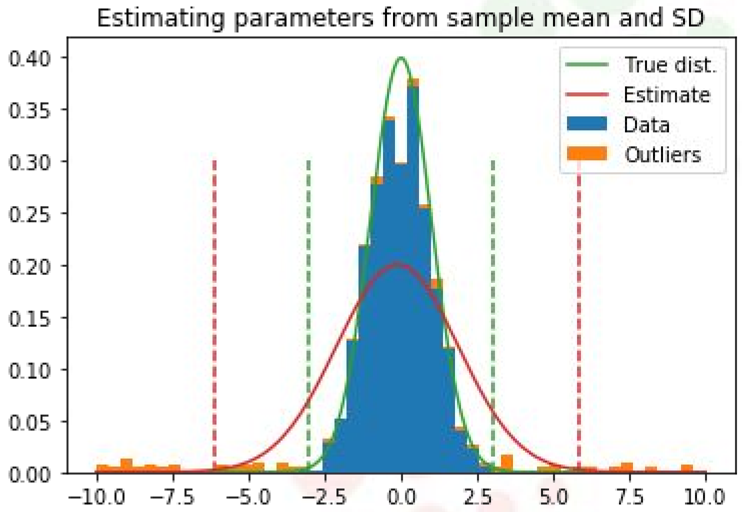
Single crystal insulin recorded on a Pilatus 6M at X06SA@SLS λ=1,0332Å

STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Sigma-clipping



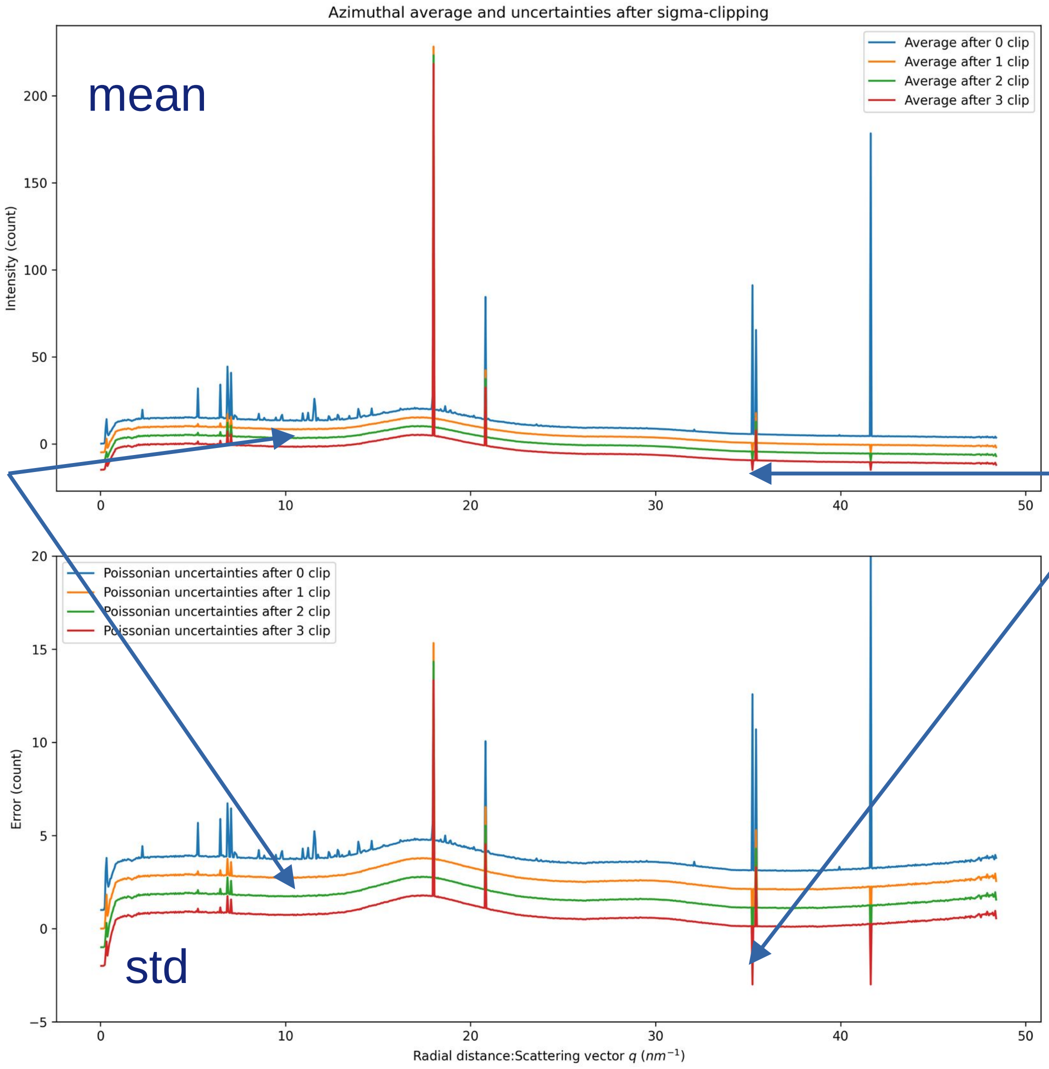
- Iterative algorithm:
  - Integrate to calculate $\bar{I}$ and $\sigma(I)$
  - Mask out any pixel with: $|I - \bar{I}| > n \cdot \sigma(I)$
- Removes both tails from the distribution:
- Good approximation of the background
- Number of iterations:
  - 3 to 5 are common
- Cut-off parameter (SNR)
  - Default value provided by Chauvenet:
  - Discard at worse 1 pixel per ring per cycle on a normal distribution
  - Depends on the size, thus on the number of bins: SNR_{clip}=2.7~3.5



$$SNR_{chauvenet} = \sqrt{2 \log\left(\frac{N}{\sqrt{2\pi}}\right)}$$

STREAMLINE has received funding from the European Union’s Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Sigma-clipping with Poisson error-model



Remove most peaks with few cycles

Empty bins results with mean=0 & std=0

Jeopardizes subsequent analysis

Uncertainties in azimuthal integration (2)

- Limits of the Poisson error model:
  - Requires all pixels in a ring to be from the **same** distribution
  - Thus incompatible with Bragg-peaks!
  - Consider for example a distribution of 2 pixels of value 1 and 99:
    - Mean: 50, std: 10, both pixels are at 5σ –> empty ensemble

- Azimuthal error model:

$$\begin{cases} variance_i = \omega_i^2 \cdot \left(v_i - \overline{v_r}\right)^2 \\ VV = \sum \omega^2 \cdot \left(\frac{signal}{normalization} - \frac{V}{\Omega}\right)^2 \end{cases}$$

  - Single-pass implemented with:

$$V_{A \cup b} = \sum \omega \cdot v = V_A + \omega_b \cdot v_b$$

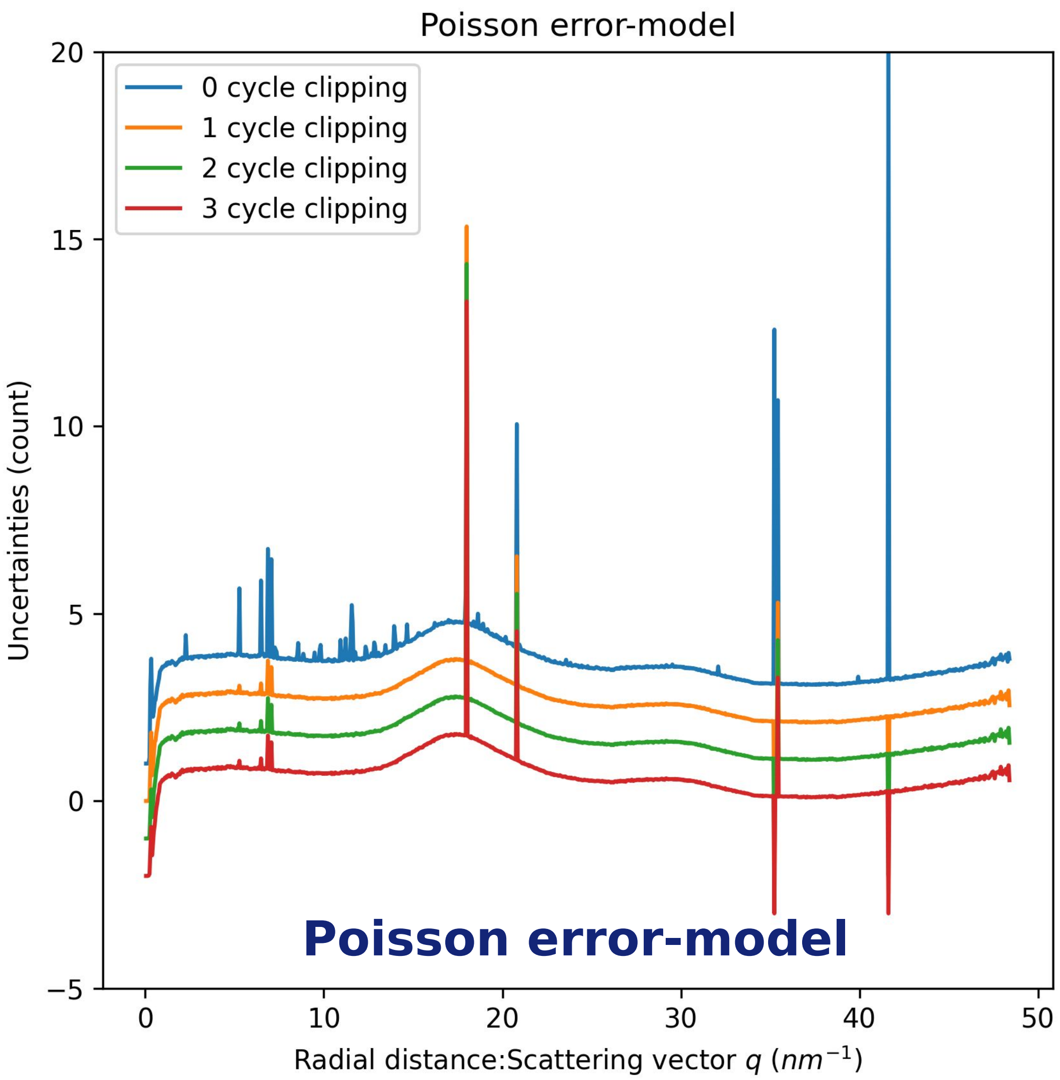
$$\Omega_{A \cup b} = \sum \omega = \Omega_A + \omega_B$$

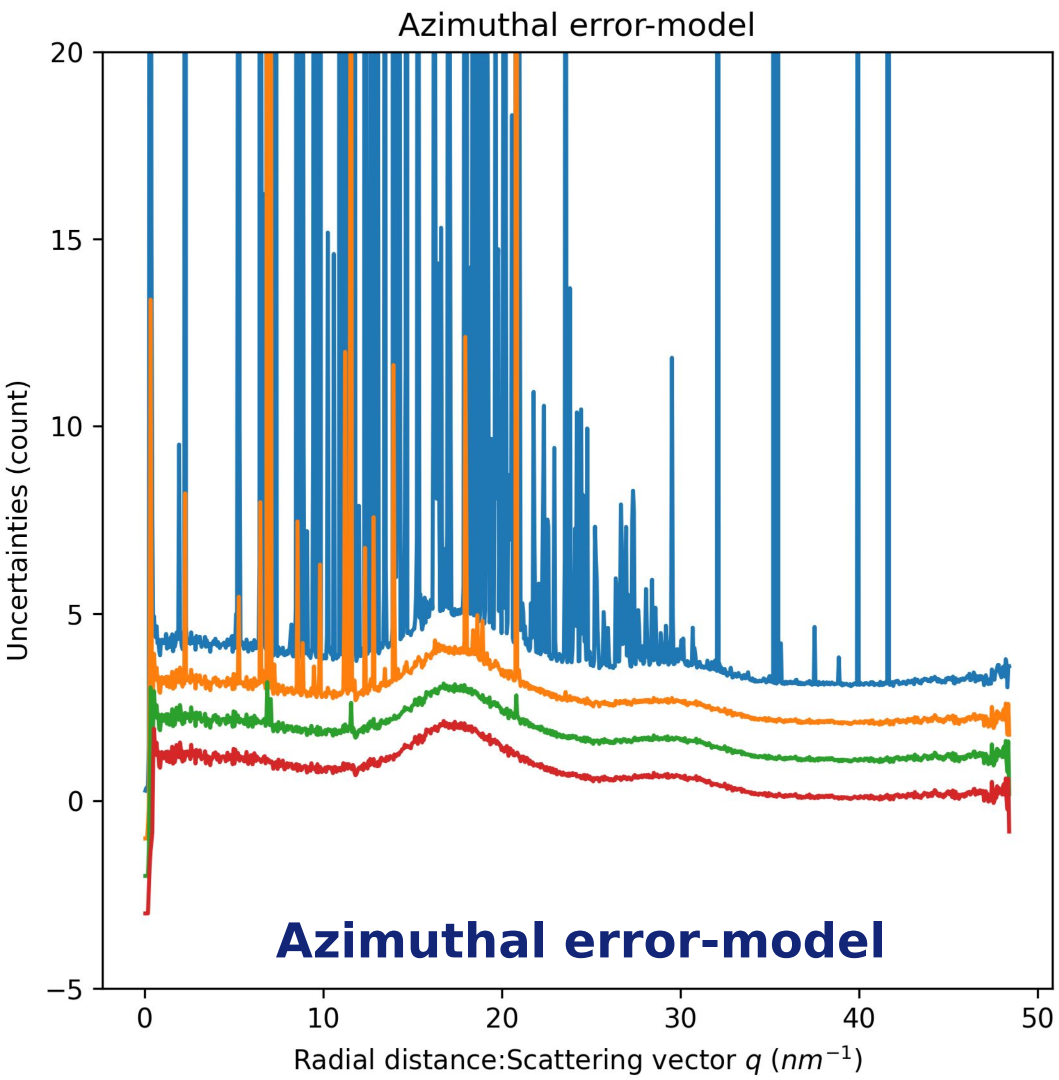
$$\Omega\Omega_{A \cup b} = \sum \omega^2 = \Omega\Omega_A + \omega_B^2$$

$$VV_{A \cup b} = VV_A + \omega_b^2 \left(v_b - \frac{V_A}{\Omega_A}\right)\left(v_b - \frac{V_{A \cup b}}{\Omega_{A \cup b}}\right)$$

STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Comparison of error-models for σ-clipping

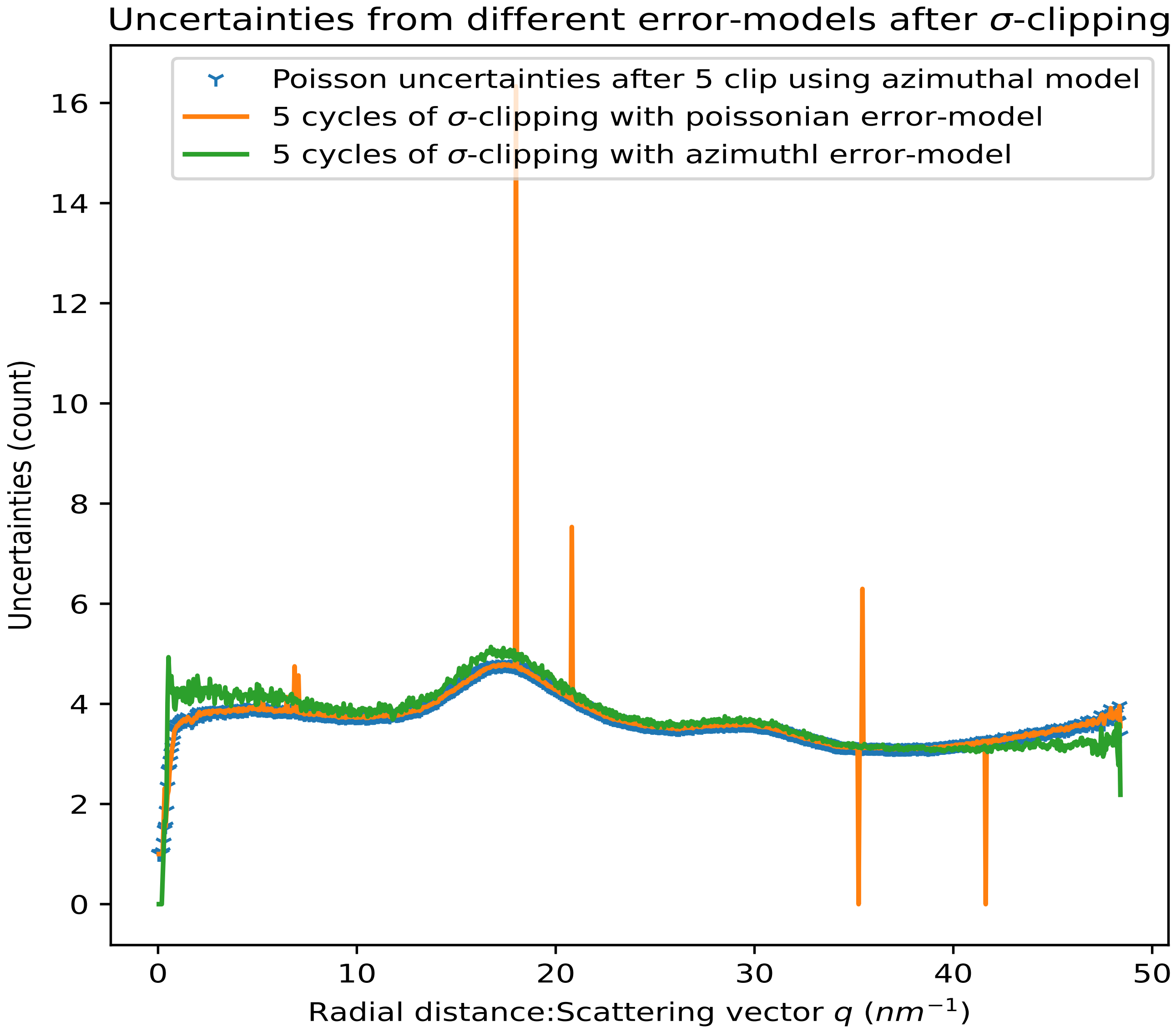




STREAMLINE has received funding from the European Union’s Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Hybrid error-model:

- Use azimuthal model for σ-clipping
  - Robust to Bragg-peaks
- Use Poisson model for subsequent analysis
  - Less noisy
  - Limits of Poisson when count → 0



STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation prog. under grant agreement No. 870313.

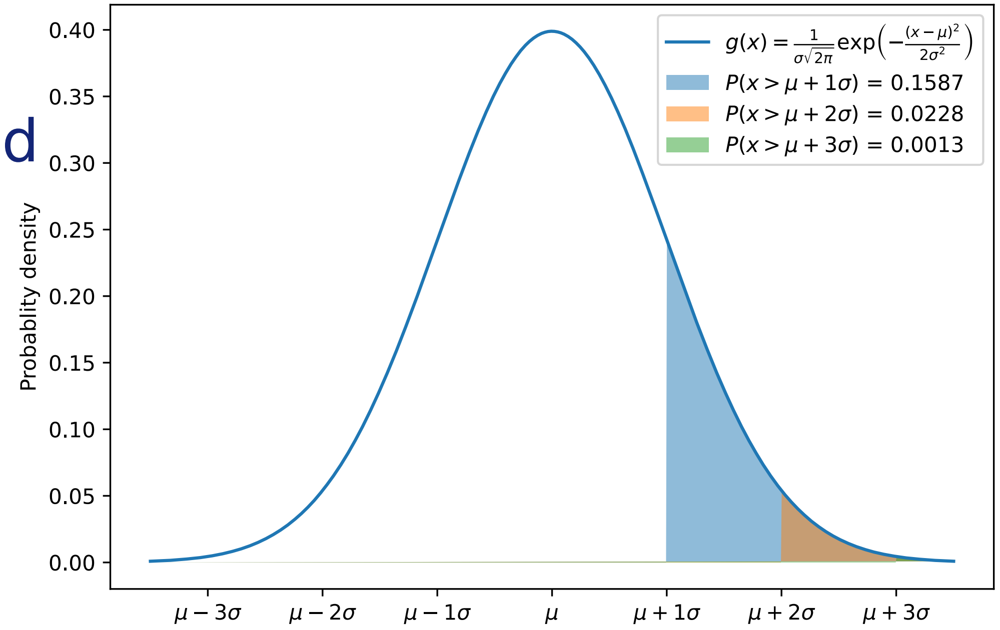
Save only intensity of pixel of interest

- Image analysis for single crystal frames
- Lossy data compression
- Peak-finding
- Conclusions

STREAMLINE has received funding from the European Union’s Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Sparsification: lossy compression

- Sparsification:
  - Store positive outlier with SNR > threshold
  - Record also its position
  - Record background avg (μ) & std (σ)
  - Compression-rate can be estimated assuming a normal distribution
  - Implemented using OpenCL in pyFAI



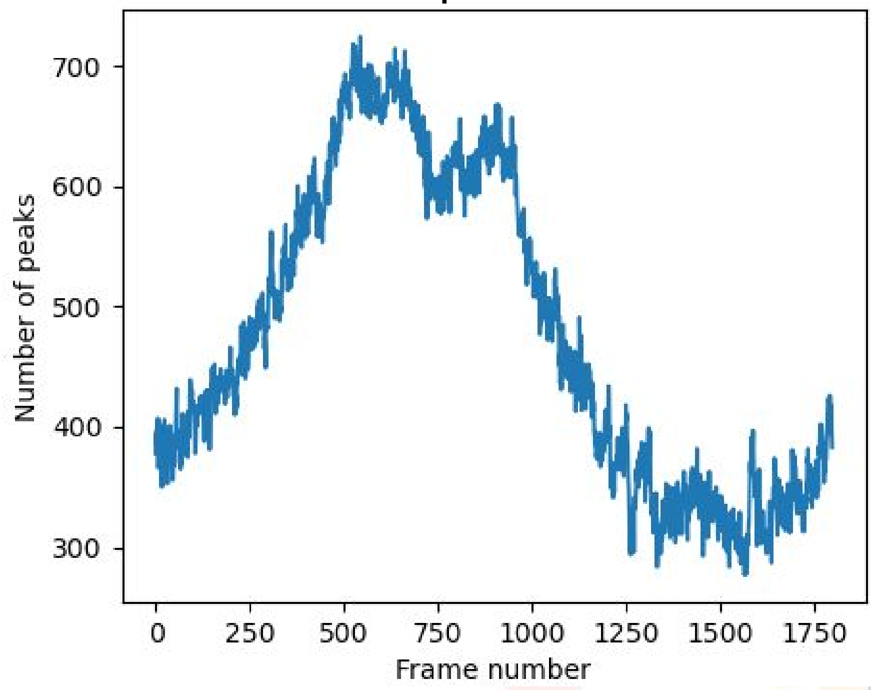
- Densification:
  - Available as part of FabIO
  - Restores frames with (or without) background noise
  - Implemented in C (GIL-free) + multi-threading

STREAMLINE has received funding from the European Union’s Horizon 2020 research and innovation prog. under grant agreement No. 870313.

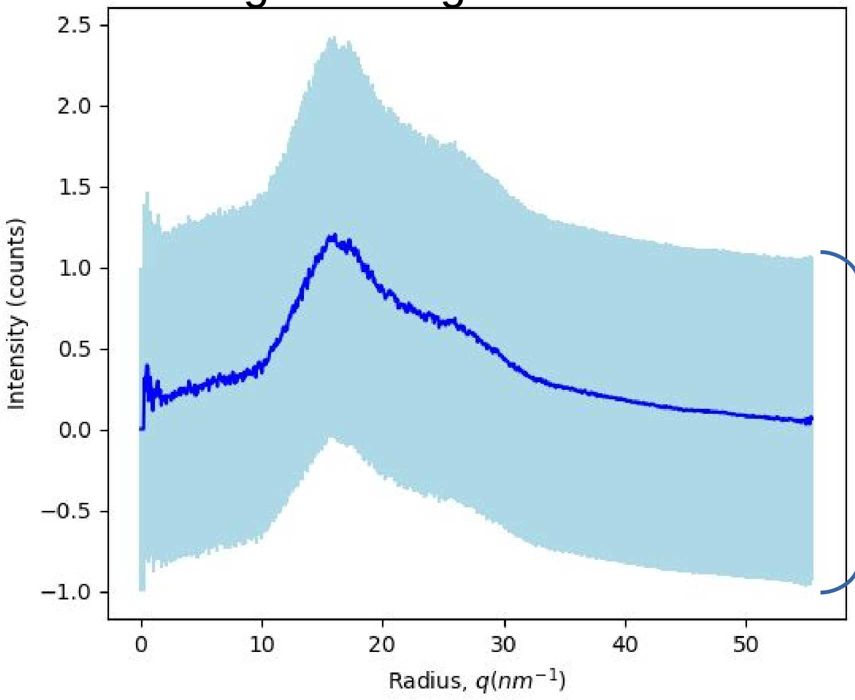
Validation of sparsified dataset:

- Raw dataset: Insulin acquired at SLS with an Eiger4M
- Comparison of quality indicator from XDS
- Sparse data compressed with:
  - Poissonian error-model
  - SNR_{clip}: automatic
  - SNR_{pick}: 1σ
  - SNR_{peak}: 5σ
  - Cycles: 5
  - Bins: 800

Peaks per frame



Background signal/uncertainties



Signal null at large q
→ std tend to 1
→ pixels ≥ 2 get recorded

Performances & quality:

- Compression of a factor: **5x** when cut-of at 1σ
- Compression speed: **250 fps** (GPU)
- Decompression speed: **200 fps** (CPU)
- Limits of the Poisson model at low count rate :μ=0→ σ=1

Indicator	Raw data			Spasified (1σ, poisson) + densified (noise)		
Size	2357 MB			439 MB		
Shell	2.91 Å	2.06 Å	total	2.91 Å	2.06 Å	total
Completeness	99.8 %	93.7 %	92.9%	99.8 %	94.1 %	93.2 %
R_{obs}	9.8 %	56.9 %	12.4%	8.9 %	67.8 %	11.0 %
R_{exp}	8.7 %	73.7 %	14.7%	8.0 %	85.6 %	12.0 %
R_{meas}	10.3 %	60.8 %	13.1%	9.3 %	72.6 %	11.6 %
$CC_{1/2}$	99.7	94.6	99.7	99.7	94.4	99.8
I/σ	25.86	5.38	10.54	26.85	3.70	10.14

STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Peak finding algorithm on a diffraction frame

- Image analysis for single crystal frames
- Lossy data compression
- Peak-finding
- Conclusions

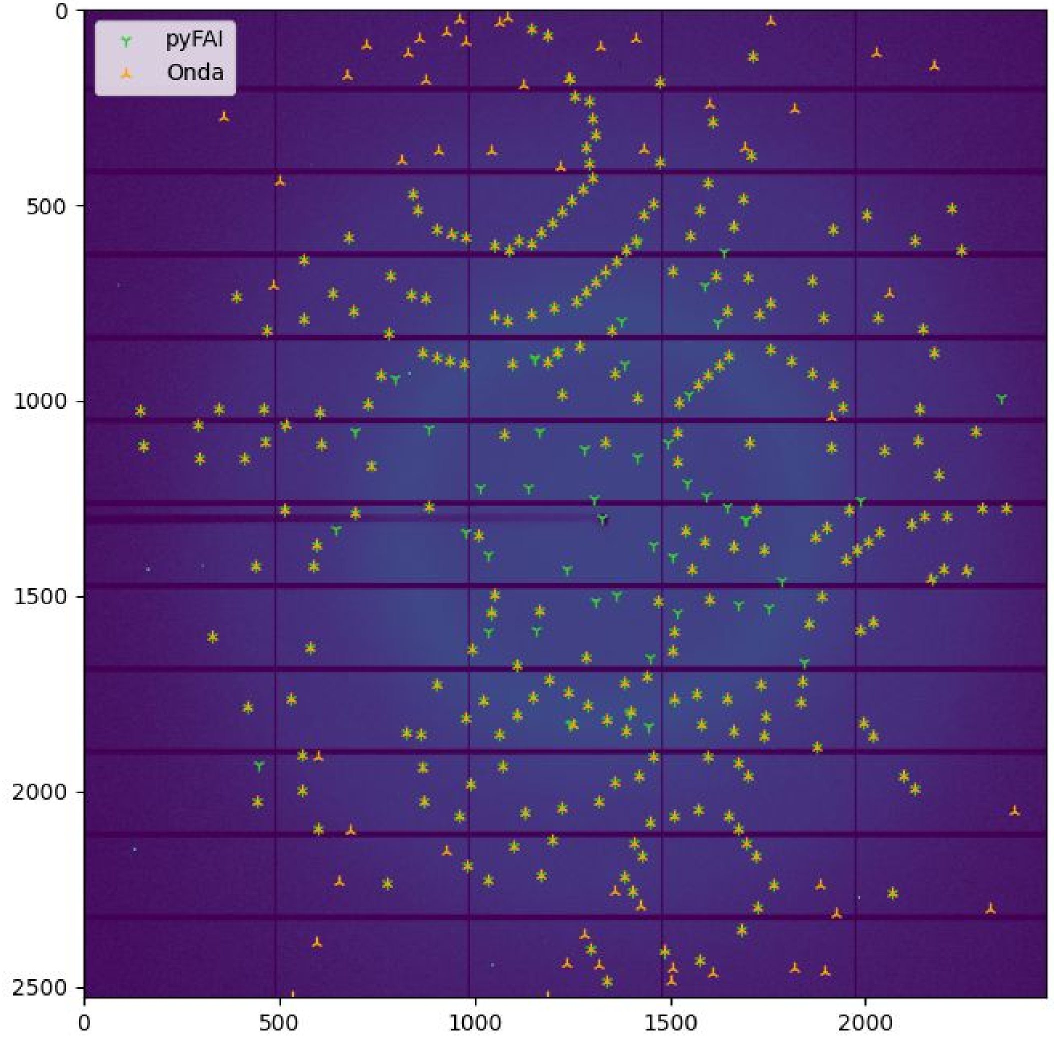
STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation prog. under grant agreement No. 870313.

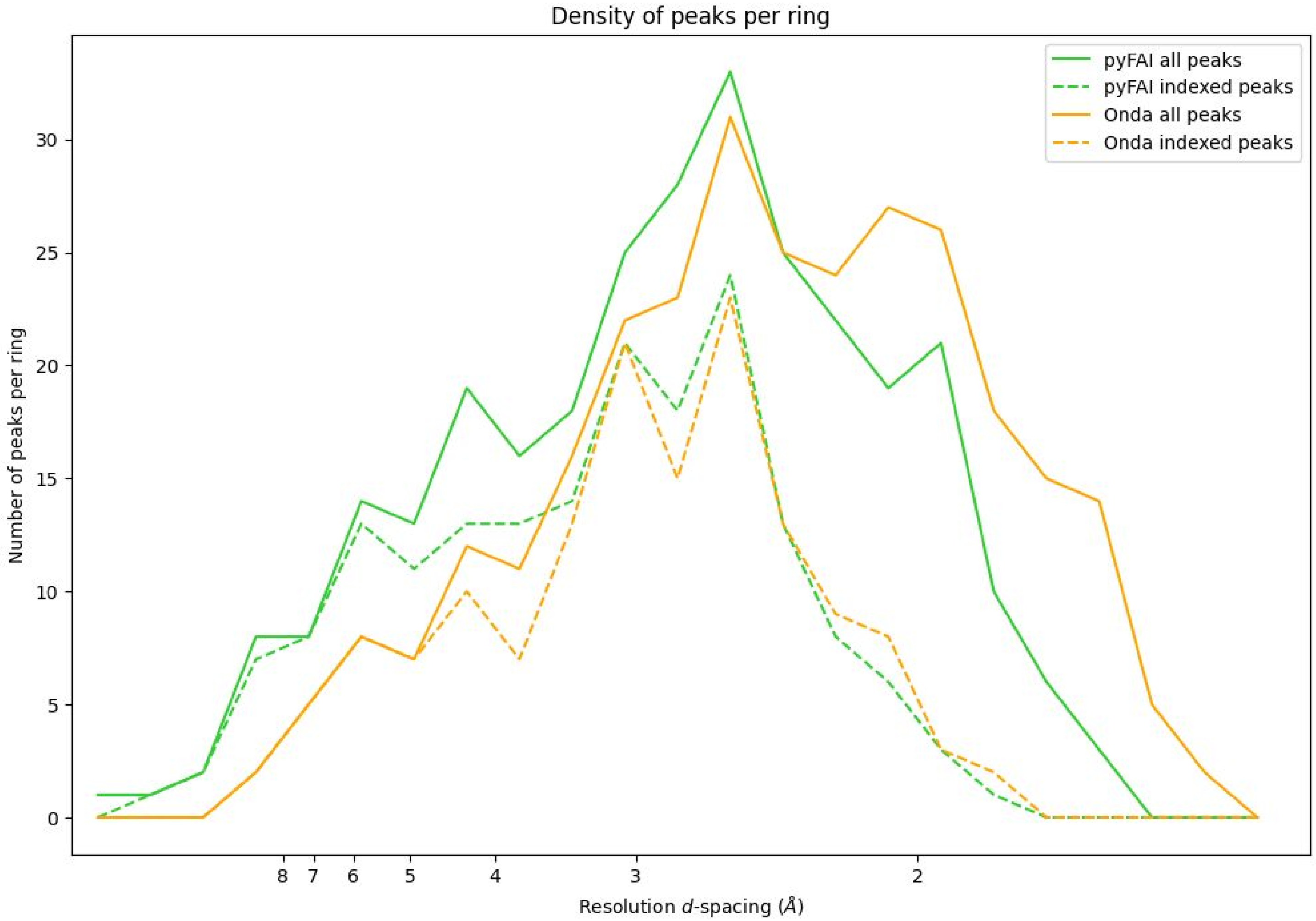
Layout of the peak-picking algorithm:

- Subtract background intensity (from σ-clipping)
  - Clip to 0 negative values. Those are all discarded.
- Pixel is a peak if:
  - Maximum within the local neighborhood (3x3 or 5x5)
  - Subtracted signal is greater than a picking threshold (SNR_{pick})
  - At least 2 or 3 other pixels in the neighborhood meet the SNR_{pick} criteria
- Then:
  - Sum subtracted intensities on the neighborhood (+ uncertainties propagation)
  - Calculate the center of mass of the peak
- Implemented on GPU using OpenCL
  - Same execution time as sparsification

STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Comparison with PeakFinder8





OnDA : Mariani, V et al. J. Appl. Cryst. 49, 1073-1080 (2016).

Cheetah: Barty, A. et al., J. Appl. Crystallogr. 47, 1118–1131 (2014).

STREAMLINE has received funding from the European Union’s Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Indexation with CrystFEL / XGANDALF

Indexer :	XGANDALF		XGANDALF-Fast	
Peak-picker	Indexation rate	Run-time	Indexation rate	Run-time
Zaef	10 %	2178 s	10 %	430 s
PeakFinder8	49.5 %	10397 s	48.5 %	1757 s
PeakFinder9	44.2 %	8328 s	43.5 %	1436 s
Robust PeakFinder	22.4 %	6314 s	21.2 %	1628 s
PyFAI peakfinder	50.2 %	9325 s	50.0 %	1595 s

1000 micro-crystal from HEWL Lysozyme collected on an Eiger 4M at ESRF-ID30a3

STREAMLINE has received funding from the European Union’s Horizon 2020 research and innovation prog. under grant agreement No. 870313.

Conclusion

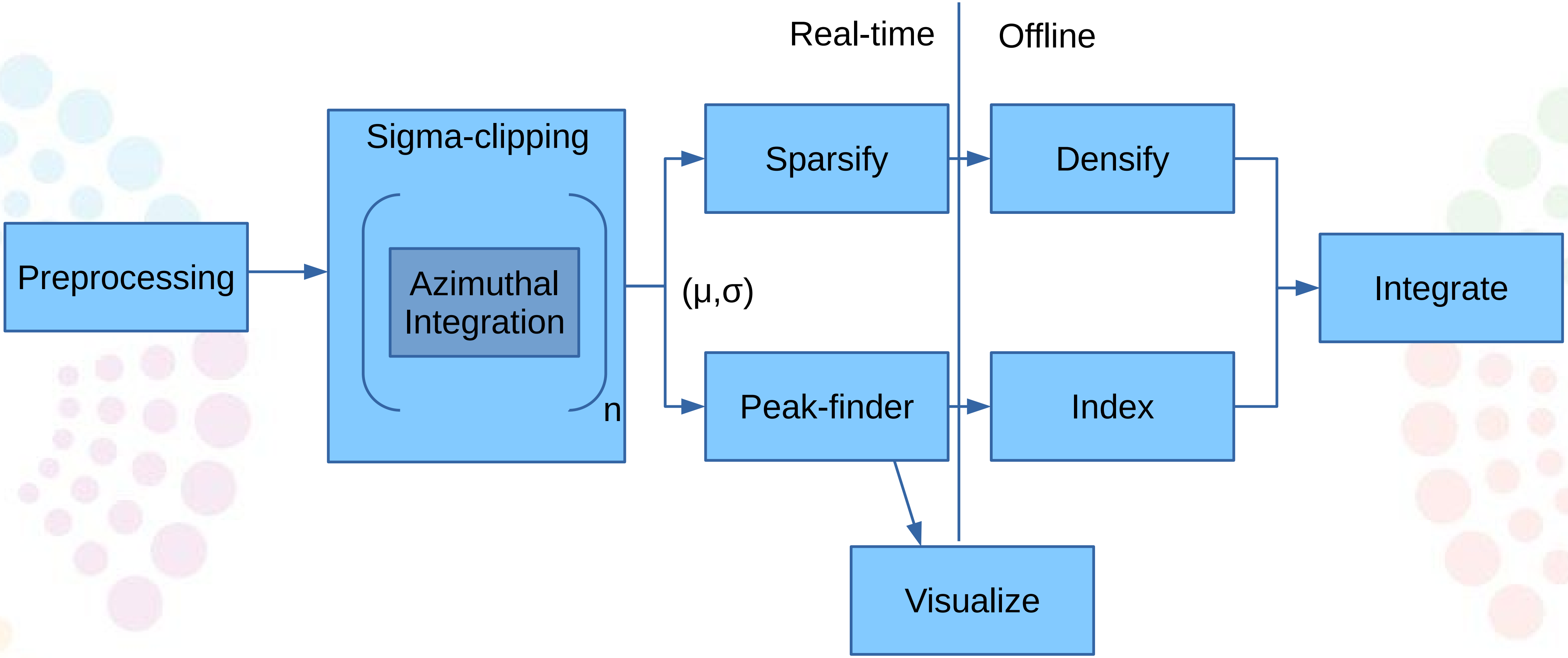
- Separation of Bragg-peaks from amorphous background using σ-clipping
  - Several error-models: Poisson, azimuthal and hybrid
  - Performance critical section for all algorithms (~3-4 ms for 4 Mpix)

- Sparse & lossy data compression for single crystal diffraction
  - Compression rate 5-100x (tuneable thanks to SNR_{pick})
  - Compression speed: 250 fps, single GPU stream
  - Decompression on CPU with background reconstruction
  - Data quality validated with XDS reduction software

- Peak-finder
  - Similar in many point to the PeakFiner8 from Cheetah (Barty, 2014)
  - Implemented on GPU @ 250 fps
  - Peak-position validated by indexing with CrystFEL

Thank you

Schematic of the processing …



Profiling (ms) Quadro A5000 / PCIe v3 16x

Kernel name	(count)	min	median	max	mean	std
copy H->D indices	1	1.775	1.775	1.775	1.775	0.000
copy H->D indptr	1	0.001	0.001	0.001	0.001	0.000
copy H->D mask	1	0.449	0.449	0.449	0.449	0.000
copy H->D radius1d	1	0.001	0.001	0.001	0.001	0.000
copy H->D radius2d	1	1.786	1.786	1.786	1.786	0.000
copy raw H->D image	**1800**	**1.679**	**2.062**	**3.405**	**2.076**	**0.145**
cast u32_to_float	1800	0.063	0.066	0.092	0.066	0.002
memset_ng	1800	0.003	0.003	0.017	0.004	0.001
corrections	1800	0.141	0.143	0.166	0.143	0.002
csr_sigma_clip4	**1800**	**3.701**	**3.858**	**4.002**	**3.851**	**0.057**
copy D->H background_avg	1800	0.001	0.002	0.002	0.002	0.000
copy D->H background_std	1800	0.002	0.002	0.002	0.002	0.000
memset counter	3600	0.005	0.006	0.036	0.007	0.001
peakfinder	**1800**	**0.252**	**0.255**	**0.264**	**0.255**	**0.001**
copy D->H counter	3600	0.001	0.001	0.004	0.001	0.000
copy D->H peak positions	1800	0.001	0.001	0.002	0.001	0.000
copy D->H peak descriptor	1800	0.001	0.001	0.003	0.001	0.000
find_intense	**1800**	**0.230**	**0.233**	**0.252**	**0.234**	**0.001**
copy D->D + cast uint32 intensity	1800	0.007	0.009	0.049	0.010	0.002
copy D->H intense pixels position	1800	0.003	0.005	0.016	0.006	0.003
copy D->H intense pixels intensity	1800	0.003	0.005	0.021	0.006	0.003

Total OpenCL execution time : 12016.502 ms

STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation prog. under grant agreement No. 870313.