Chromaflow: A GPU-Based Optical Simulation Tool for Liquid Noble Detectors and Beyond

Xiang (Alex) Li, PhD Candidate

TRIUMF, Simon Fraser University

CAP 2026

University of Ottawa

2026/06/23

What is Chroma?

- **Motivation**
  - Fast optical simulation for noble-liquid and photosensor detector R&D.
- **What it is**
  - GPU-based optical photon simulation.
  - Forked by nEXO from the original Chroma framework: github.com/BenLand100/chroma
  - Runs on NVIDIA GPUs; supports CUDA 11.x and 12.3+.
  - Compatible with HPC clusters and Apptainer/Singularity containers.
- **Geometry workflow**
  - Import full detector CAD as STL meshes and configure optical properties in data tables.
  - Avoids aggressive geometry simplifications when detailed CAD is available.

Projects

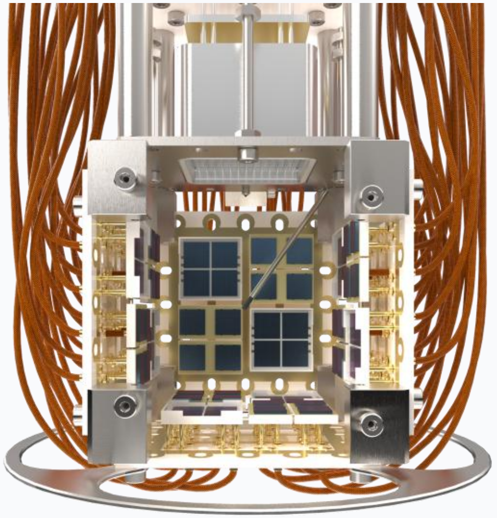
- **Chroma-flow** (formerly Chroma-simulation) users:
  - nEXO
    - LoLX, XeCube, and the UMass LXe setup
  - XLZD
- Forks / related efforts
  - DARWIN
  - EoS

Chroma source code in CUDA C and pycuda (Open source)

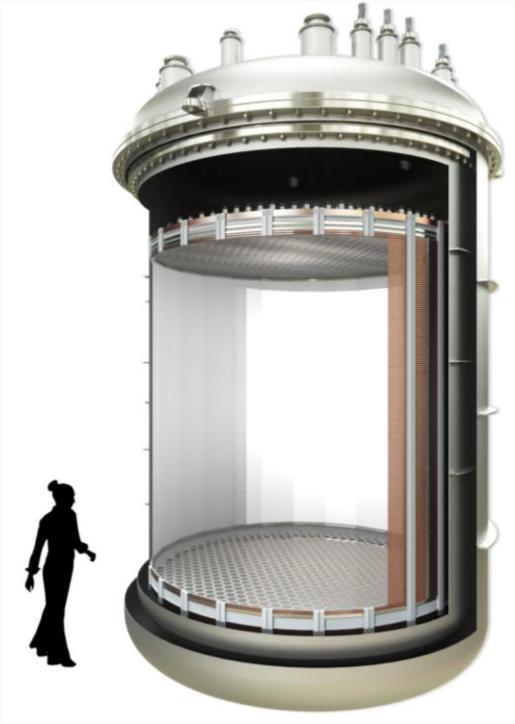
→ DARWIN, EoS (Project-specific close source)

→ XLZD, nEXO: Chroma-flow (**Open source soon!**)

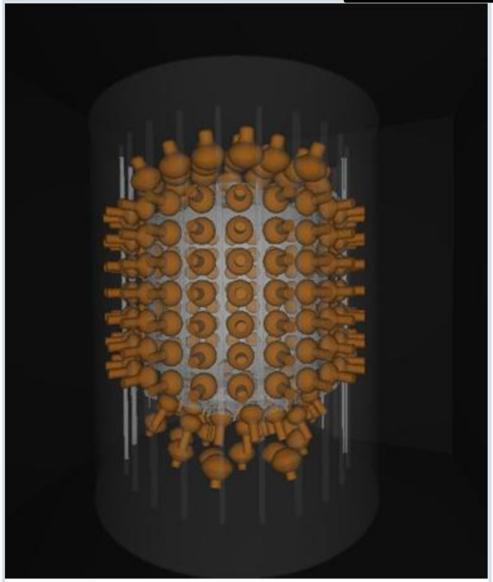
→ nEXO, LoLX, XeCube,UMass LXe setup…



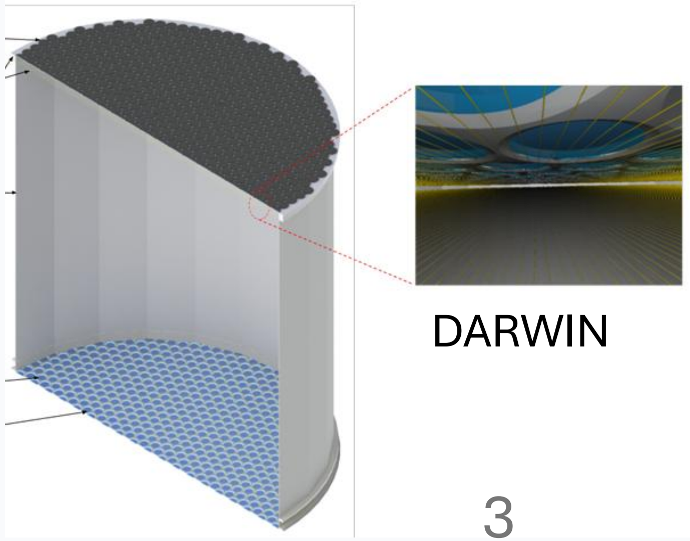
LoLX



XLZD



Eos in Chroma



DARWIN

Chroma vs Opticks

- Two widely used GPU-based optical photon simulation packages.

Opticks

Used by NEXT, LHCb, EIC and JUNO (muon group)

Pros

Fully integrated with GEANT4.

More memory efficient.

Trade-off

More complex setup.

Chroma

Used by DARWIN, EoS, nEXO, and LoLX

Pros

Easy to set up and run.

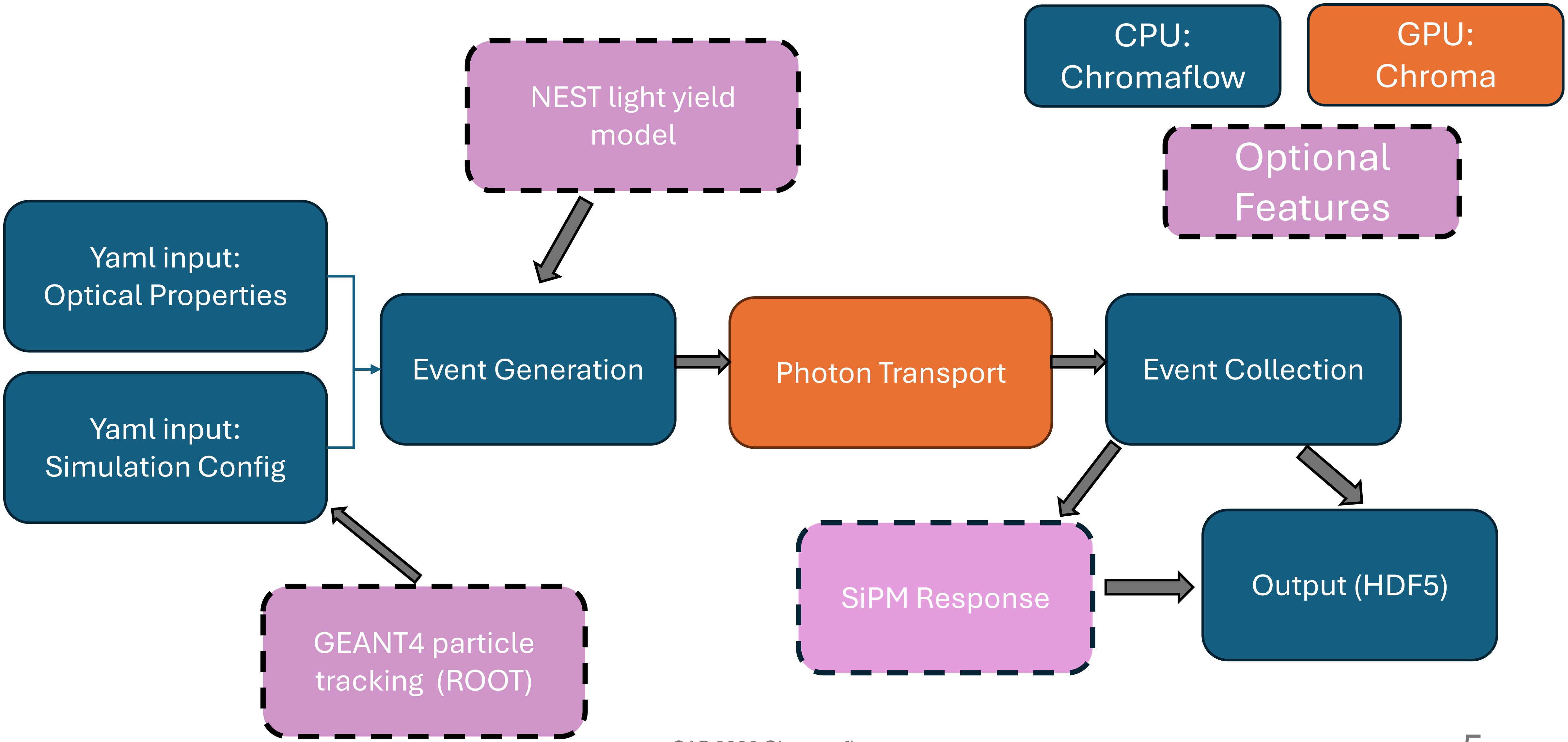
Can work without GEANT4.

Trade-offs

Separate geometries for GEANT4 and Chroma.

Higher memory use for detailed mesh geometry.

Chroma-flow workflow



Chroma-flow interface

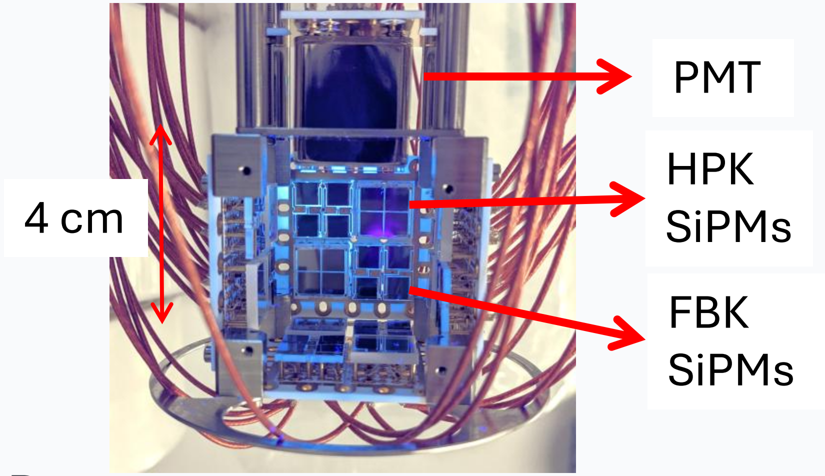
- **Inputs**
  - Geometry as STL meshes.
  - Optical properties Yaml defines each component.
    - Optical properties include wavelength shifter, diffuse/specular reflection, Fresnel equations…
  - Simulation Yaml config.
    - Similar to GEANT4 macros.
- **Photon generators**
  - Photons source from GEANT4 ROOT or custom random sampling.
  - Scintillation, Cherenkov, laser, …
- **Outputs**
  - Photon-level, event-level, and waveform-level output.
  - Waveform output can include an SiPM response model.

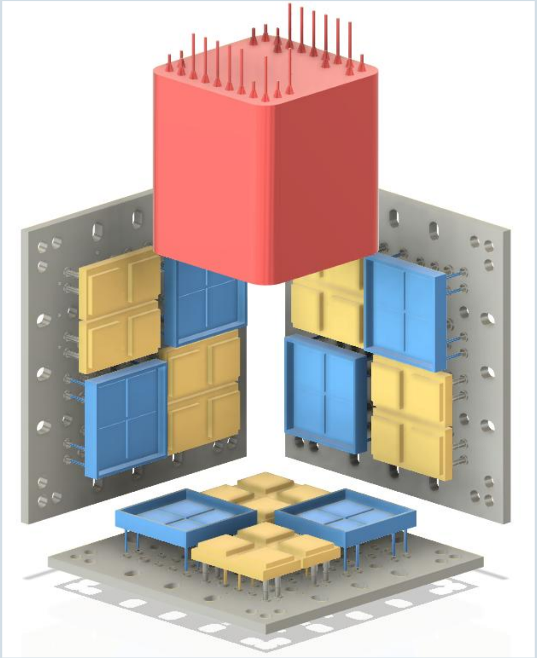
```yaml
Detector:                                         # Define parameters for the construction
  DetectorName: LoLX                              # Defines detector name
  PathToDetector: '[ChromaPath]/Geometry/LoLX2_run1_v2/'  # Location of STL files
  OpticalProperties: '[ChromaPath]/Yaml/LoLX/OpticalProperties_LoLX_tested.yaml' # Lo
  CacheName: LoLX_Tests                           # Location where geometry will be cached
  Save: False                                      # Boolean that decides if geometry will
  BuildDetector: FromSTL                          # [FromSTL, FromMesh, FromCache] Geometry
  Center: Tube                                    # Defines which component will be used to
  Detector: ['FBK_SiPM', 'HPK_SiPM', 'Photocathode']  # Defines which component(s) h
  DetectorOrientation: Tube                       # [CylinderBarrel, CylinderCap] Orientati
  FiducialVolume:                                 # Choose components that will be used to
    Top: Tube                                     # Component that defines max z position
    Bottom: Tube                                  # Component that defines min z position
    Radius: Tube                                  # Component that defines max r position
  FiducialHeightCut: 0                            # Defines standoff in the z direction wrt
  FiducialRadialCut: 0                            # Define standoff in the radial direction
  Rotation:  None
  OutsideMaterial: Vacuum                         #Sets material outside the detector
```

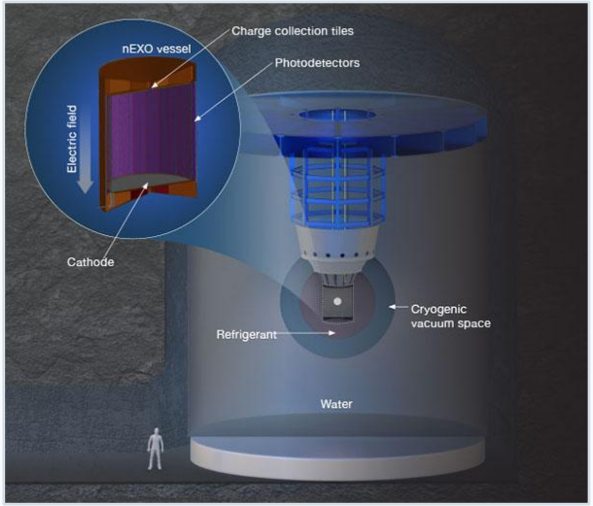
```yaml
Aluminium:
  IndexOfRefractionRe:
    0: !!python/tuple [1.549800e+02, 0.072525]
    1: !!python/tuple [1.653120e+02, 0.082340]
    2: !!python/tuple [1.771200e+02, 0.094118]
    3: !!python/tuple [1.907450e+02, 0.10843]
    4: !!python/tuple [2.066400e+02, 0.12673]
  IndexOfRefractionIm:
    0: !!python/tuple [1.549800e+02, 1.6369]
    1: !!python/tuple [1.653120e+02, 1.7793]
    2: !!python/tuple [1.771200e+02, 1.9502]
    3: !!python/tuple [1.907450e+02, 2.1283]
    4: !!python/tuple [2.066400e+02, 2.3558]
  Absorption: 0.2

Ceramic:
  Absorption: 0.8
  SpecularReflectivity: 0.1
  DiffuseReflectivity: 0.1
  AbsorptionLength: 0.0001  #mm
  ScatteringLength: 0.0001  #mm
```

Use cases

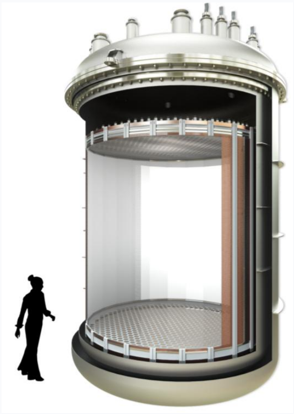
- **LoLX: Light-only Liquid Xenon experiment**
  - Modular cube array of photosensors immersed in LXe.
    - SiPMs performance in LXe.
      - See Frédéric Girard’s talk on Thursday
    - Study energy resolution.
    - Characterize external crosstalk.
- **nEXO & XLZD: 0νββ & Dark matter search**
  - Study light response and detection efficiency for detector R&D.
  - Explore background-rejection techniques.
  - Support other optical R&D developments.







nEXO: neutrinoless double-beta decay search



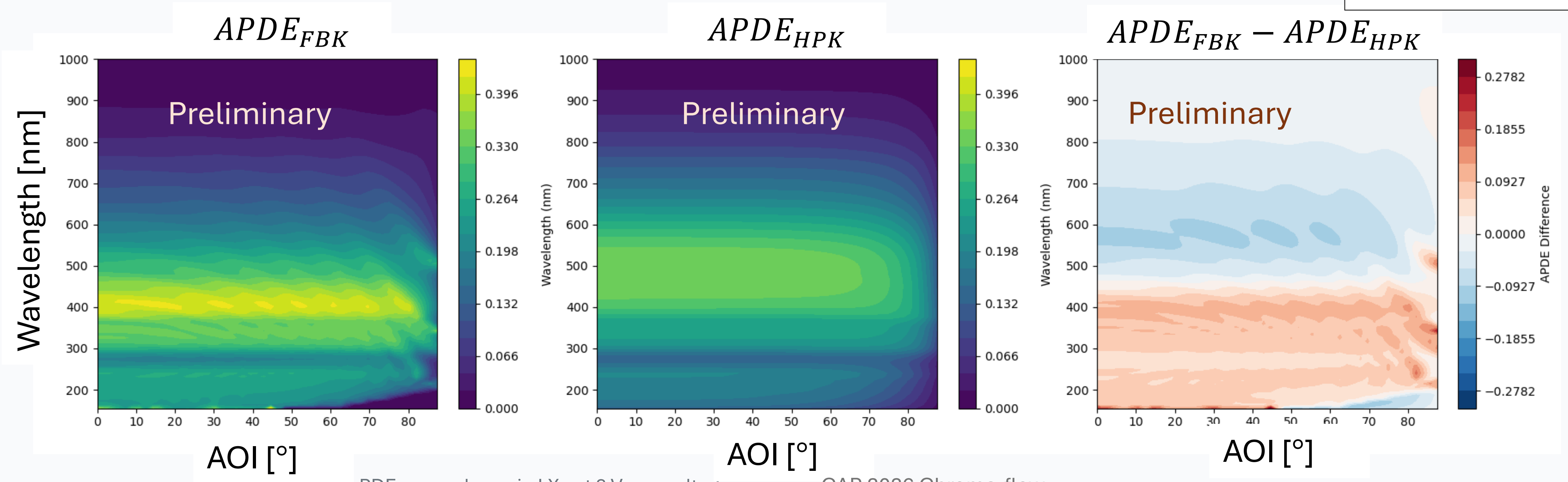
XLZD

Wavelength- and angle-dependent PDE model

- SiPM Photon detection efficiency (PDE) model: doi.org/10.1016/j.nima.2026.171526.
  - Optical effects, avalanche probabilities.
- Implications for our experiment and yours:
  - PDE is convoluted with photon transport efficiency at the event level.
  - Wavelength dependence matters for scintillation, Cherenkov, and external crosstalk studies.

FF: Fill factor.
$T(\lambda,\theta)$: Transmission.
$iPDE(\lambda,V_{OV})$: Internal PDE.
AOI: angle of incidence.

Absolute PDE: $APDE(\lambda,\theta,V_{OV}) = FF \cdot T(\lambda,\theta) \cdot iPDE(\lambda,V_{OV})$

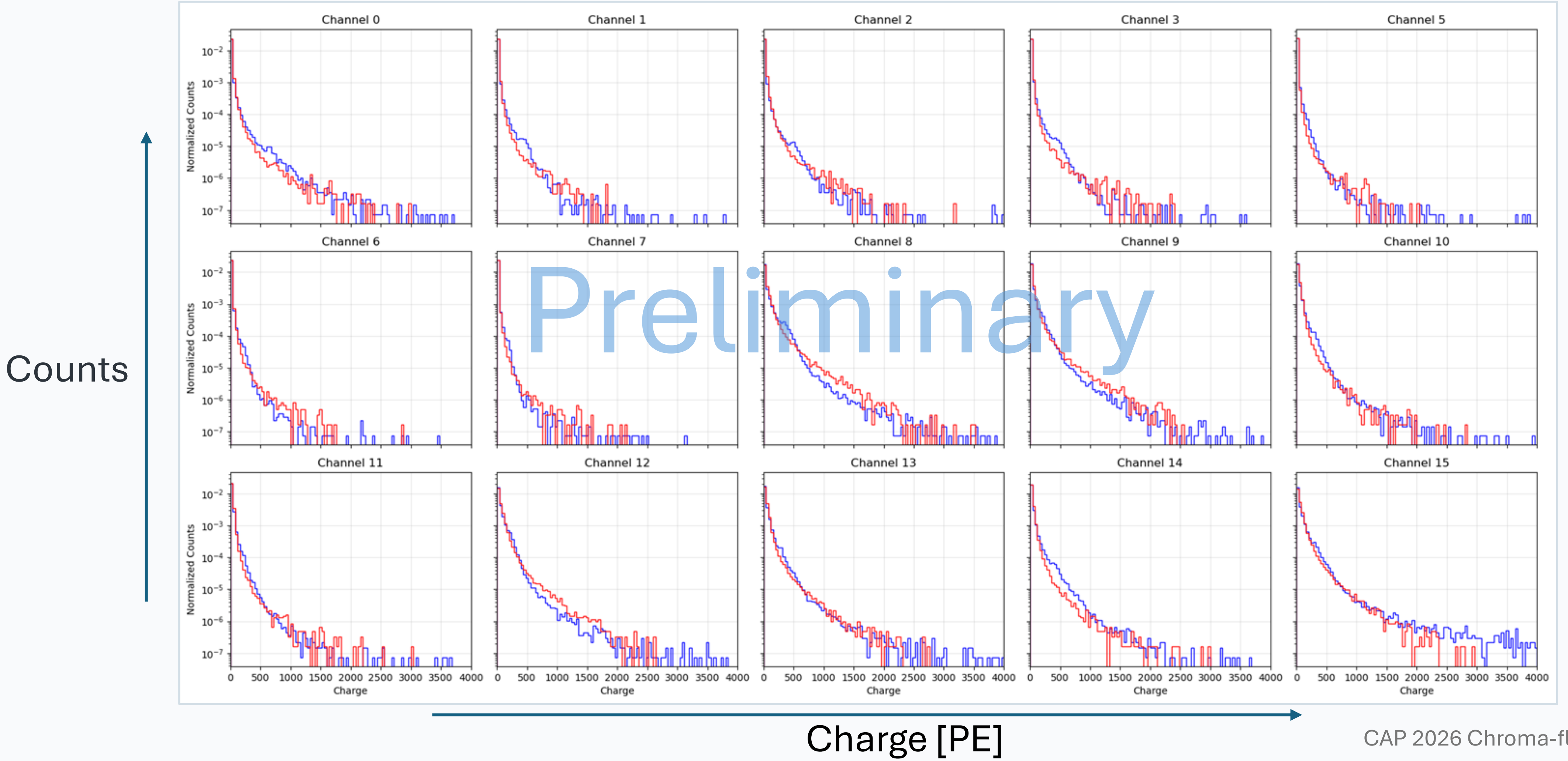


PDE maps shown in LXe at 3 V overvoltage.

Simulation–data comparison in LoLX

- Cs-137 external source: data vs simulation.

Charge distributions across channels show good overall agreement.



Red: simulation
Blue: data

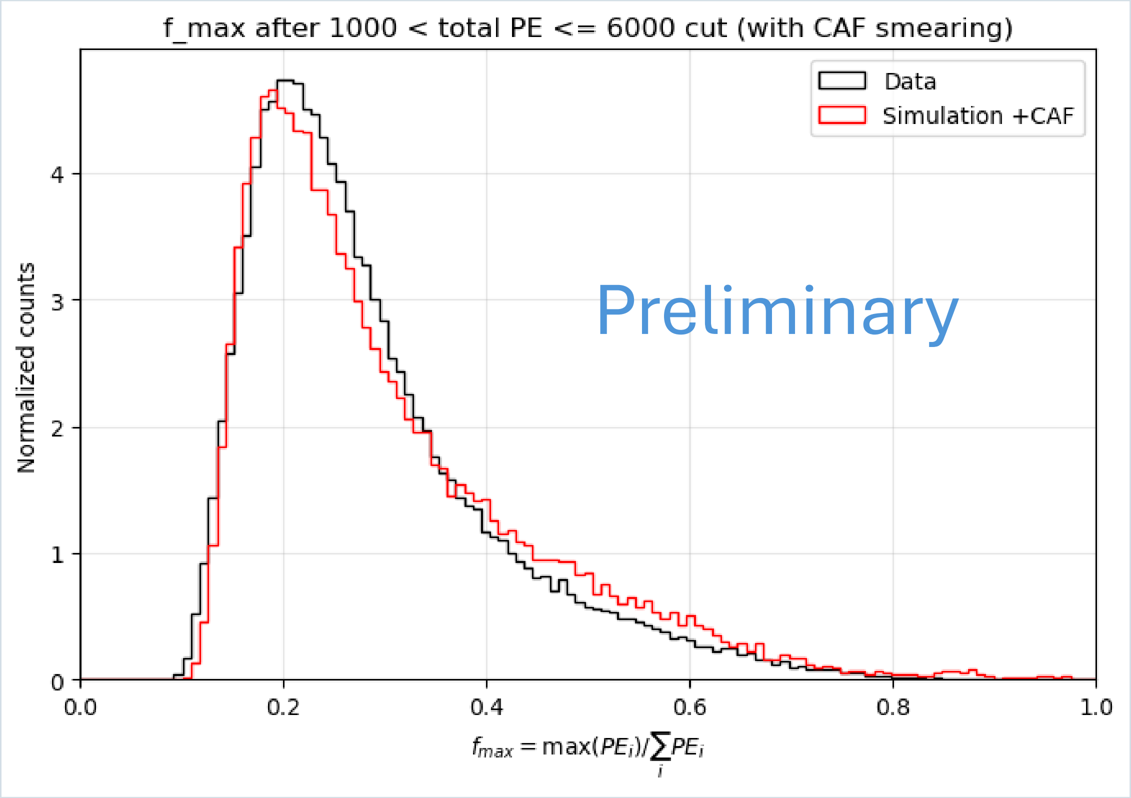
f_{max} Light distribution in LoLX

TRIUMF

- It represents the light distribution over detector volume.

$$f_{,max} = \frac{\max(\boldsymbol{Q})}{\sum_i Q_i}$$

$\boldsymbol{Q}$: Channel charge in photoelectron.

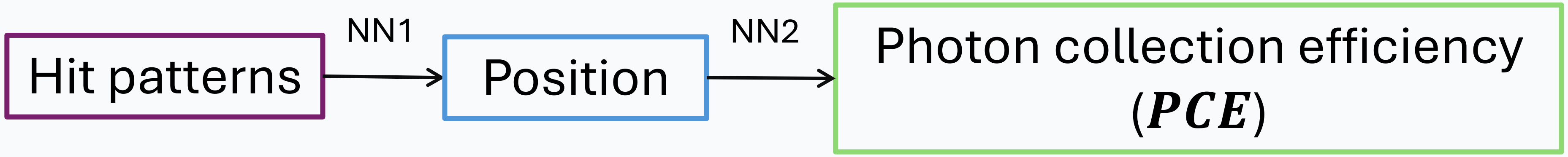


Light spread out over channels ⟷ Light too localized to one channel

Position and energy reconstruction

TRIUMF

- Simulated electron-gun energy: ~356 keV
  - NEST photon generator.
  - Scintillation photons with LXe wavelength distribution.
- Feedforward neural network (FNN)



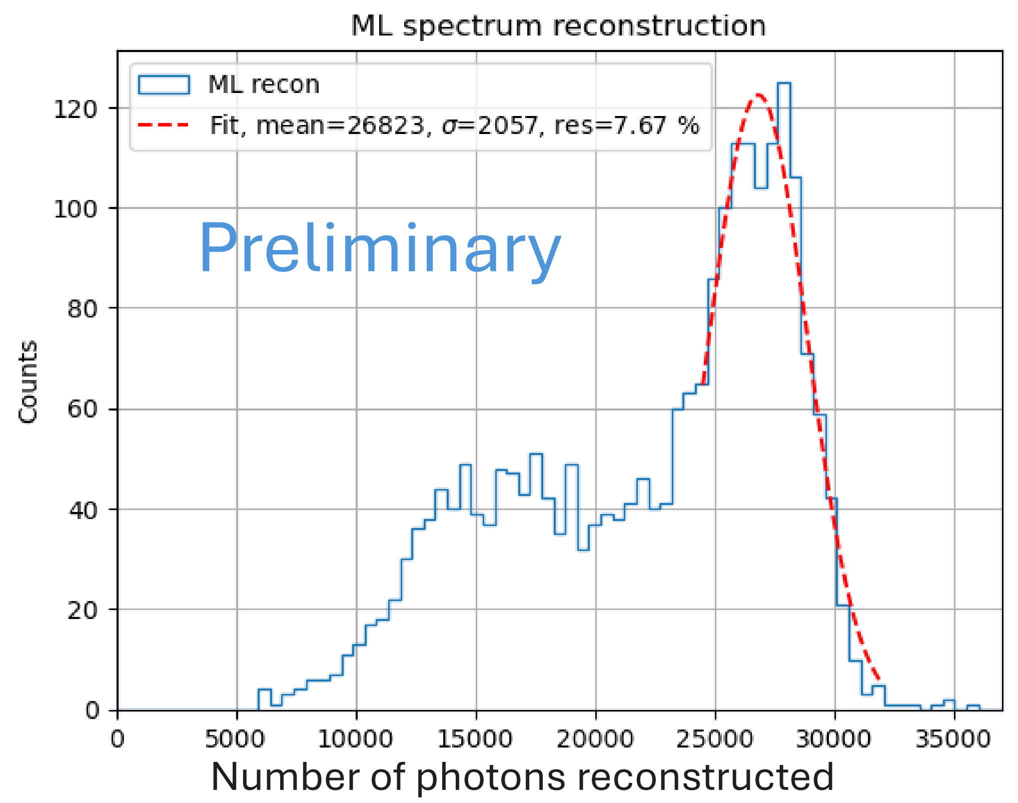
Hit patterns = $\boldsymbol{Q}\ [Q_1 \dots Q_{27}]/\sum Q_i$

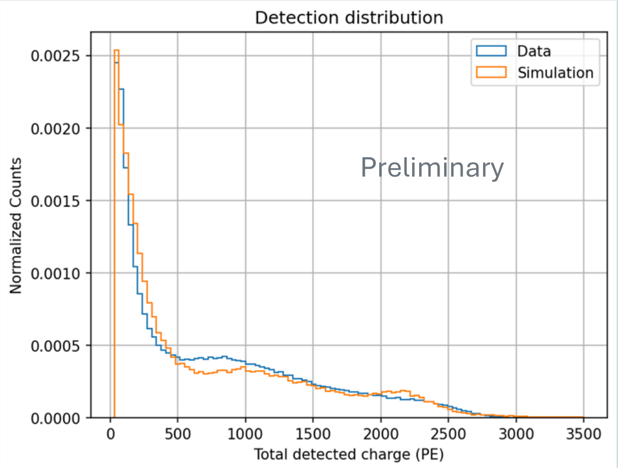
$$N_{recon} = \frac{\sum Q_i}{\sum PCE_i}$$

- $\boldsymbol{Q}\ [Q_1 \dots Q_{27}]$: Number of PE detected by each channels (array of 27 channels).
- $\boldsymbol{PCE}\ [PCE_1 \dots PCE_i]$:Photon collection efficiency for each channel (array of 27 channels).
  - Include all optical effects, transport efficiency and photon detection efficiency.

Reconstruction on real data

- Cs-137 (662 keV gamma) data reconstruction.
  - Light yield (LY): $40.6 \pm 3.1\ ph/keV$.
    - NEST γ-ER LY: 55 ~ 65 ph/keV
    - NEST β-ER LY: 40 ~ 50 ph/keV
      - https://doi.org/10.3389/fdest.2024.1480975
  - Measure Resolution: $7.7 \pm 0.7\%$ (σ/μ)
    - K. Ni (2006), reported: $7.7 \pm 0.5\%$ (σ/μ)
      - https://doi.org/10.1088/1748-0221/1/09/P09004
    - A. Minamino (2010), XMASS reported: 5.2 ± 0.2 %.
      - https://doi.org/10.1016/j.nima.2010.03.032





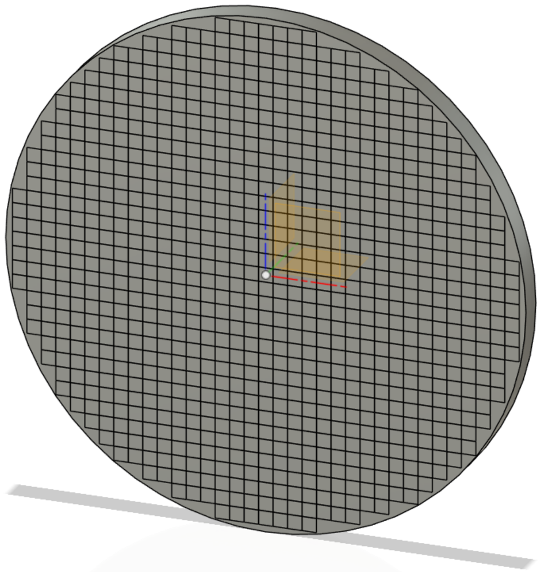
ML reconstruction

Better latest result in Fandresena's talk on Monday.

Selection: Likelihood < 0.95; events outside the 34 mm cube rejected.

Simulation in XLZD

- XLZD sensitivity.
  - Light-map (photon transport efficiency).
- Detector optimization.
- SiPMs vs PMTs options.
  - Backgroud radiation from PMT material.
  - SiPMs external cross-talk impact.

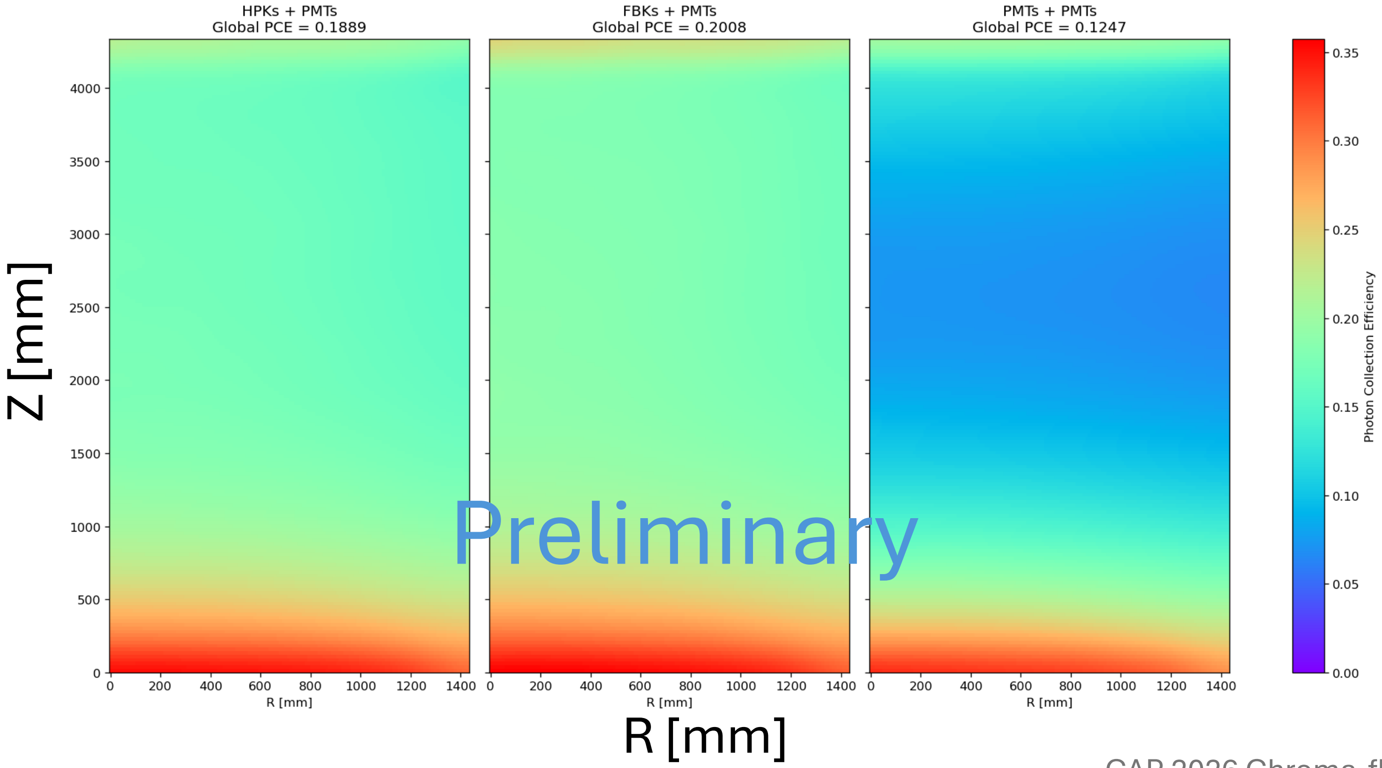


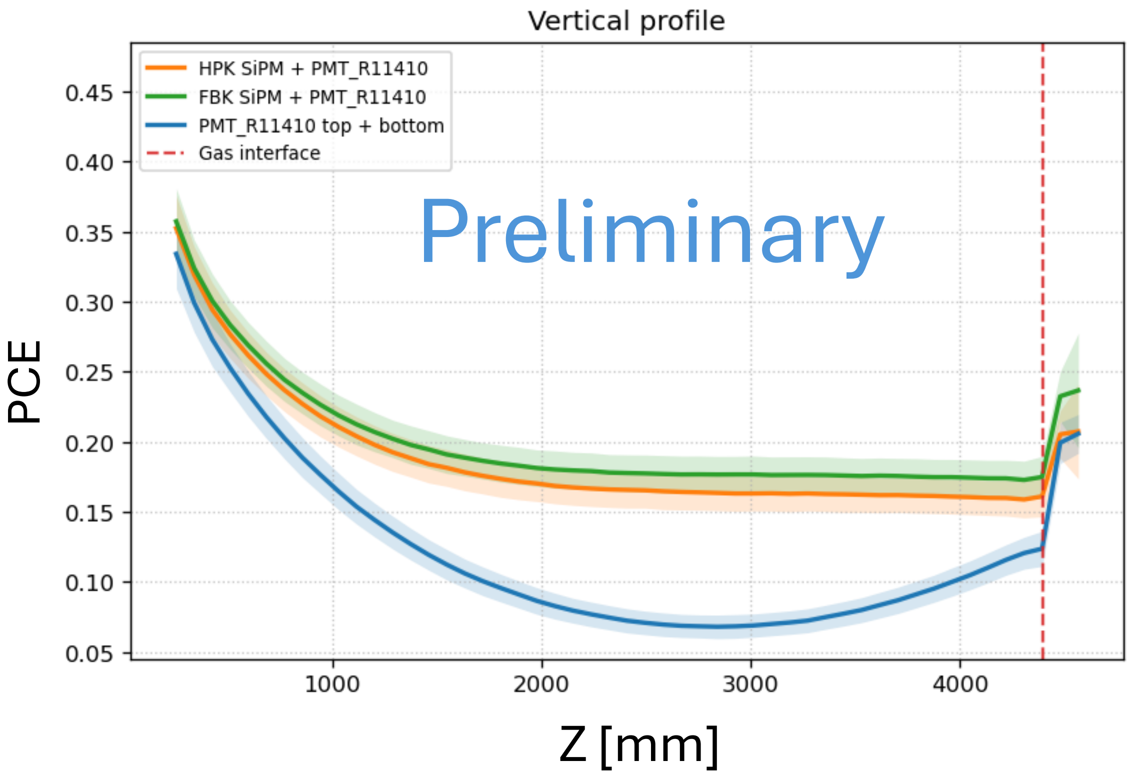
Top SiPM tiles

XLZD chroma geometry

Simulation in XLZD

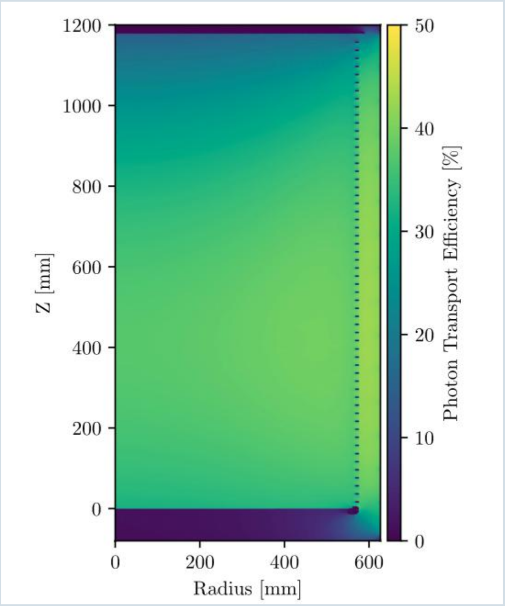
- Quick light map estimation.
- Photobomb uniform sampling over the volume.
- PTFE surfaces, SiPMs PDE model.



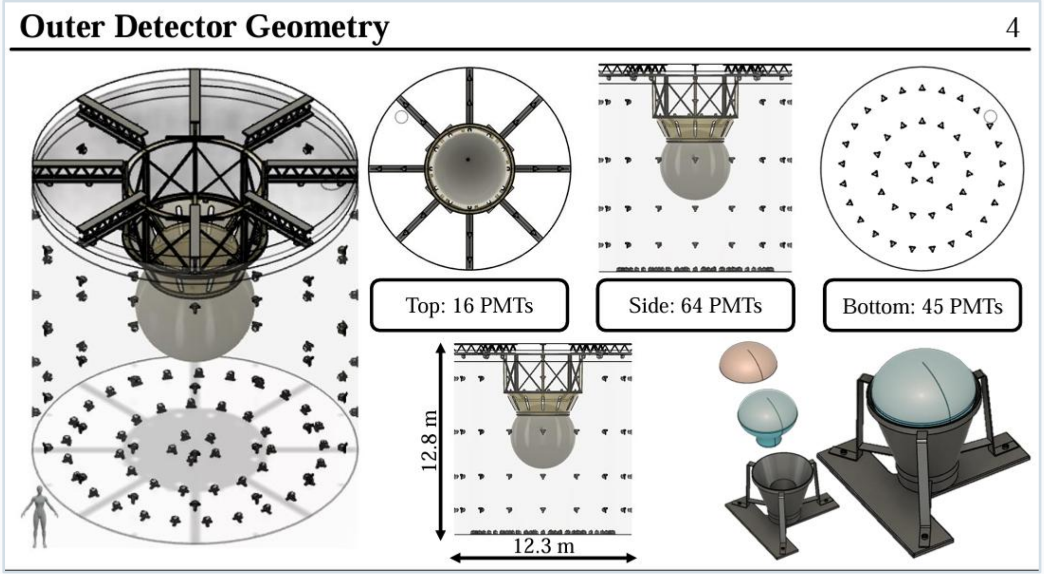


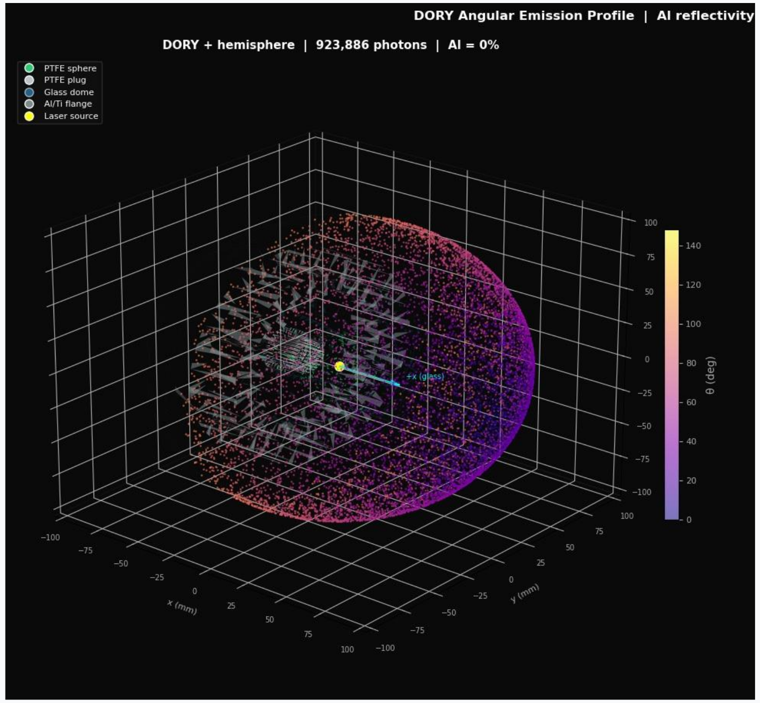
Simulation in nEXO

- nEXO sensitivity. doi.org/10.1088/1361-6471/ac3631, and doi.org/10.1103/6h4p-nk1y.
  - Light-map (photon transport efficiency) studies — credit Ako Jamil, David Gallacher, Molly Watts.
  - CNN reconstruction and MS/SS discrimination — credit Sierra Wilde.
- nEXO veto and its calibration system design.
  - Outer-detector optimization — credit Soud Al Kharusi, Samin Majidi.
  - DORY (Diffuser-Based Optical Monitoring System) — credit Samin Majidi, David Paz.



nEXO light map





DORY: Diffuser-Based Optical Monitoring System

Summary and takeaway

TRIUMF

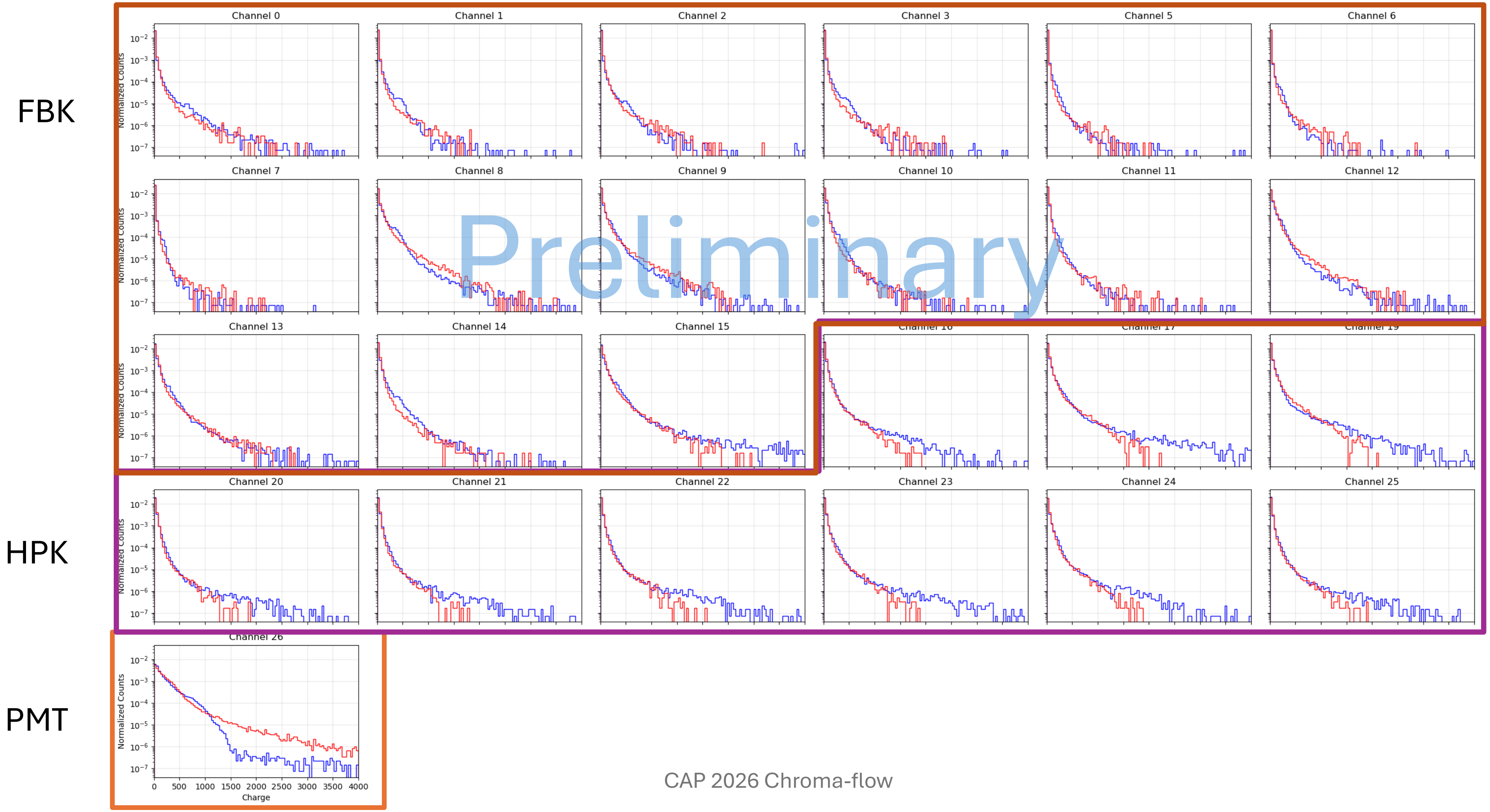
- Chroma-simulation is a soon-to-be full open-source GPU optical-photon simulation tool.
- Data and simulation show compatible light distributions across channels from LoLX experiment.
- **Current use cases:**
  - Energy and position reconstruction with light-only channels.
  - Light-map studies in large detectors.
  - Detector optimization.
- Applicable to other experiments and detector concepts.
- Chroma-flow is currently in limited-access pre-release closed beta while the manuscript is being prepared; contact me if you are interested in applying it to your project.

Thank you!
Questions?

Where to find me:
Email: xili@triumf.ca
Personal page (under construction):
https://alex-x-li.github.io/alex-li-site/

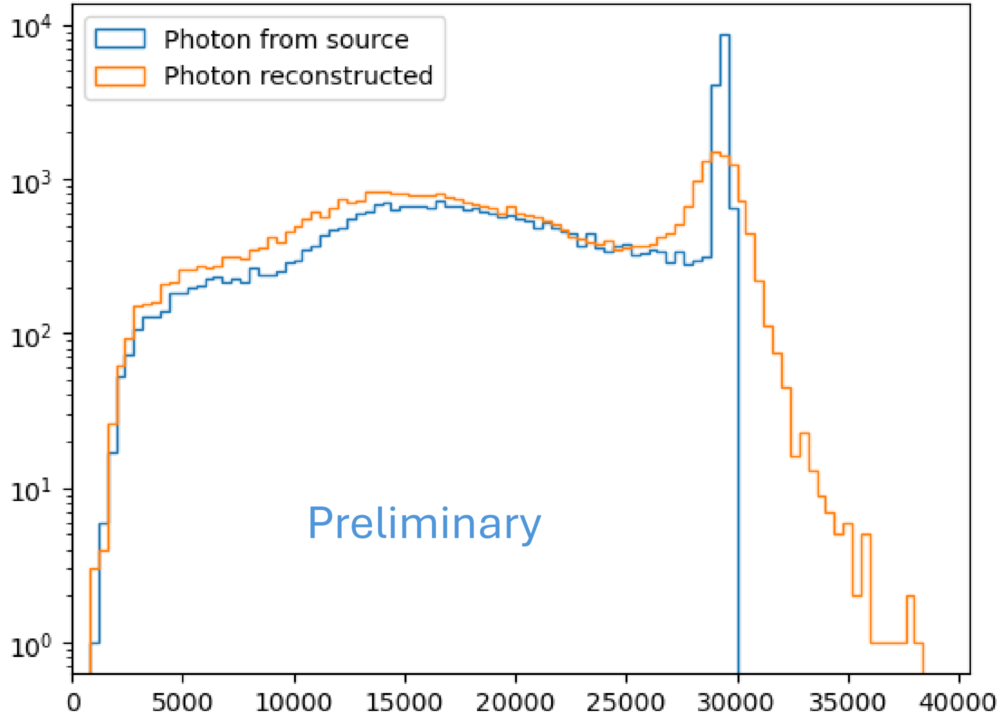
Backup

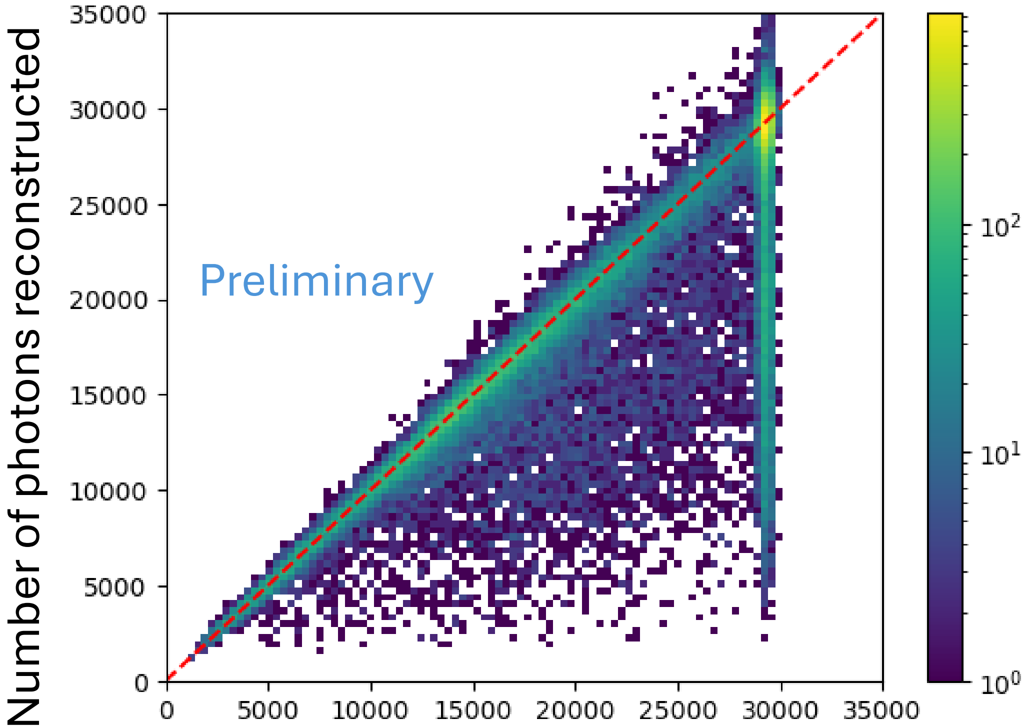
- All channels compare.



Validation within simulation – Gamma ray

- Geant 4 produced Cs-137 decay gamma ray (662keV).
- Chroma handles optical photons base on the energy deposition.





Likelihood and fiducial cut applied:
Likelihood<0.95, events outside of 34mm cube.

Neural network (NN)

- Feedforward neural network is used.
  - Simple and fast. Good for small detector with few channels.
- NN1: Mapping from source position to DAQ channel charge.

$$\boldsymbol{r_s} \mapsto \boldsymbol{q}$$

  - Where $\boldsymbol{q} = [q_0 \dots q_i]$, i is the index in DAQ channels.
  - q_i is the fraction of charge observe by channel i.

$$q_i = \frac{Q_i}{N_{gen}}$$

  - It is essential because q_i is the link to photon collection efficiency (PCE)

$$PCE = \sum q_i = \frac{\sum Q_i}{N_{gen}}$$

    - N_{gen}: Number of photon generated.
- NN2: Mapping from DAQ channels charge to source position.

$$\boldsymbol{q}_{norm} \mapsto \boldsymbol{r_s}$$

  - Where $q_{i,norm} = \frac{q_i}{\sum q_i} = \frac{Q_i}{\sum Q_i}$

Reconstruction steps

Apply the ML neural network

$$\boldsymbol{Q}_{obs} \Rightarrow \boxed{\boldsymbol{q}_{norm} \mapsto \boldsymbol{r}_{s,recon} \mapsto \boldsymbol{q}} \Rightarrow N_{s,recon}$$

- Where $\boldsymbol{Q}_{obs}$ is the charge in DAQ channel. It can be simulation or real data.
- Important to note that $\boldsymbol{q}_{norm}$ and $\boldsymbol{q}$ are different!

$$q_{i,norm} = \frac{q_i}{\sum q} = \frac{Q_i}{\sum Q_i}$$

$$q_i = \frac{Q_i}{N_{gen}}$$

- $PCE = \sum q_i = \frac{\sum Q_i}{N_{gen}}$

$$N_{s,recon} = \frac{\sum Q_i}{\mathrm{PCE}} = \frac{\sum Q_i}{\sum q_i}$$

Likelihood function

- KL-divergence base NLL function:

$$L(\boldsymbol{r_s}) = \sum_i p_i \ln \frac{p_i}{q_{i,norm}(\boldsymbol{r_s})}$$

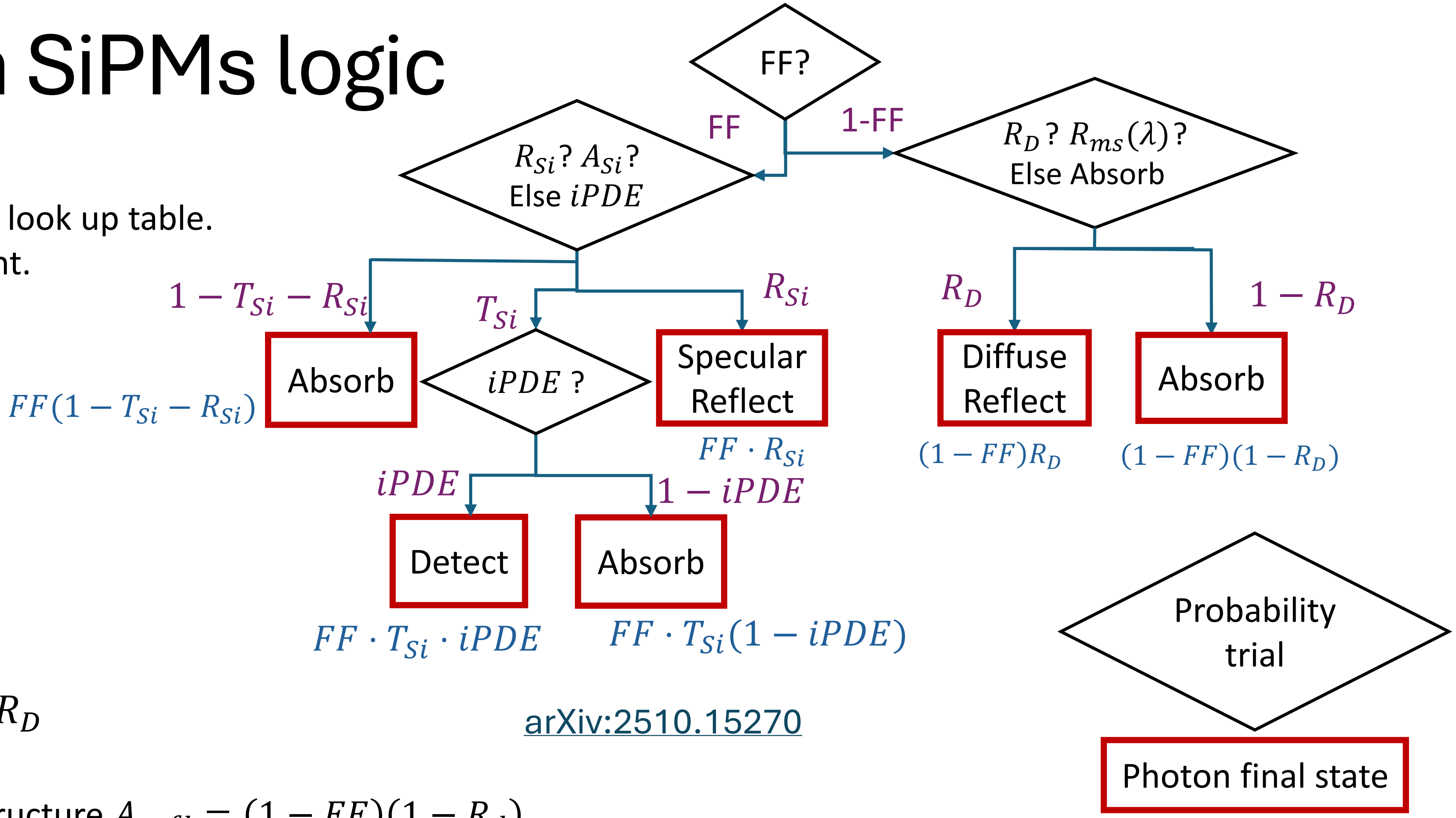
  - P_i: Charge ratio in each channel in real data.
  - Q_i: ML reconstruct charge ratio in each channel.

- From the LoLX data, charge ratio detected in channel i: $p_i = \frac{Q_{i,data}}{\sum_i Q_{i,data}}$.

- Reconstruct photon ratio in channel i, $q_{i,norm}\,(r_{recon}) = \frac{q_i}{\sum q_i}$

Chroma SiPMs logic

- Input:
  - $T_{si}, R_{Si}, iPDE,$: 2D look up table.
  - R_D, FF: Fix constant.



arXiv:2510.15270

Purple text: probability in the trials.
Blue text: effective probability.

- Effective probabilities:
- $R_{D,eff} = (1 - FF)R_D$
- $R_{Sp,eff} = FF \cdot R_{Si}$
  - Absorb in microstructure $A_{refl} = (1 - FF)(1 - R_d)$
    - For small wavelength, neglected in model just for now.
- $PDE = FF \cdot T_{Si} \cdot iPDE = FF \cdot (1 - R_{sp})iPDE$