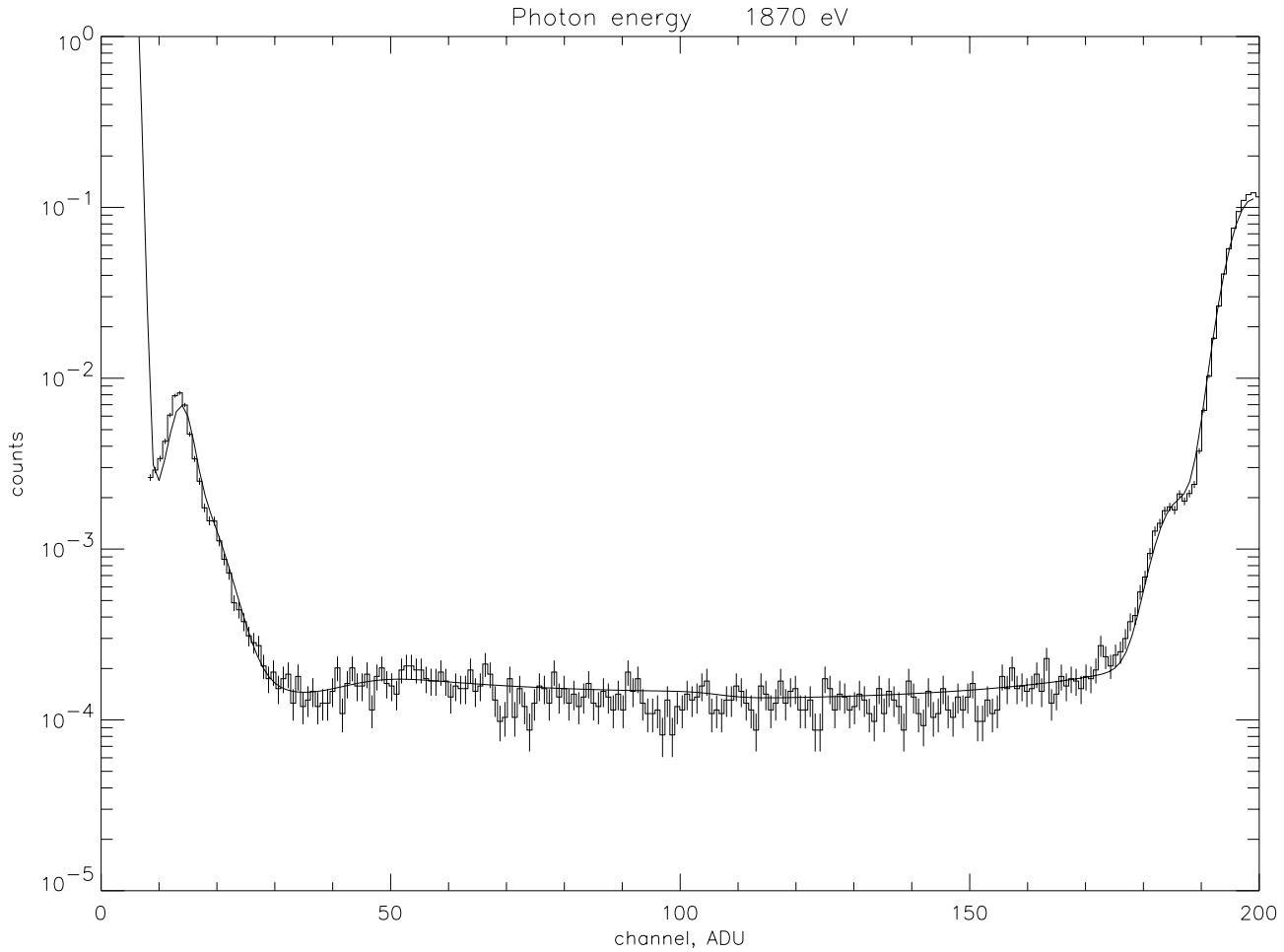
Automatic Generation of ACIS Response Matrices from Simulated Data

- Simulator developed in MIT and PSU accurately reproduce ACIS response to X-rays:



- How to go from simulated events to RMFs?

Simulated events ⟶ RMFs

- **CXC/CIAO approach:**

  a) Fit simulated response to a function at several *E*'s.

  b) Software interpolates fit parameters to compute response at any *E*.

  Labor-intensive process. Fits are required in multiple positions (in FI chips, $32 \times 4 \times 8$). Tweaks have to be applied to match in-flight calibration.

- ***PSU corrector* approach**

  Directly simulate RMFs on a pre-defined energy grid.

  Computer-intensive process ("1100 hours of CPU time to generate RMFs"). Limits user's freedom.

Simulated events ⟶ RMFs

- **CXC/CIAO approach:**

  a) Fit simulated response to a function at several *E*'s.

  b) Software interpolates fit parameters to compute response at any *E*.

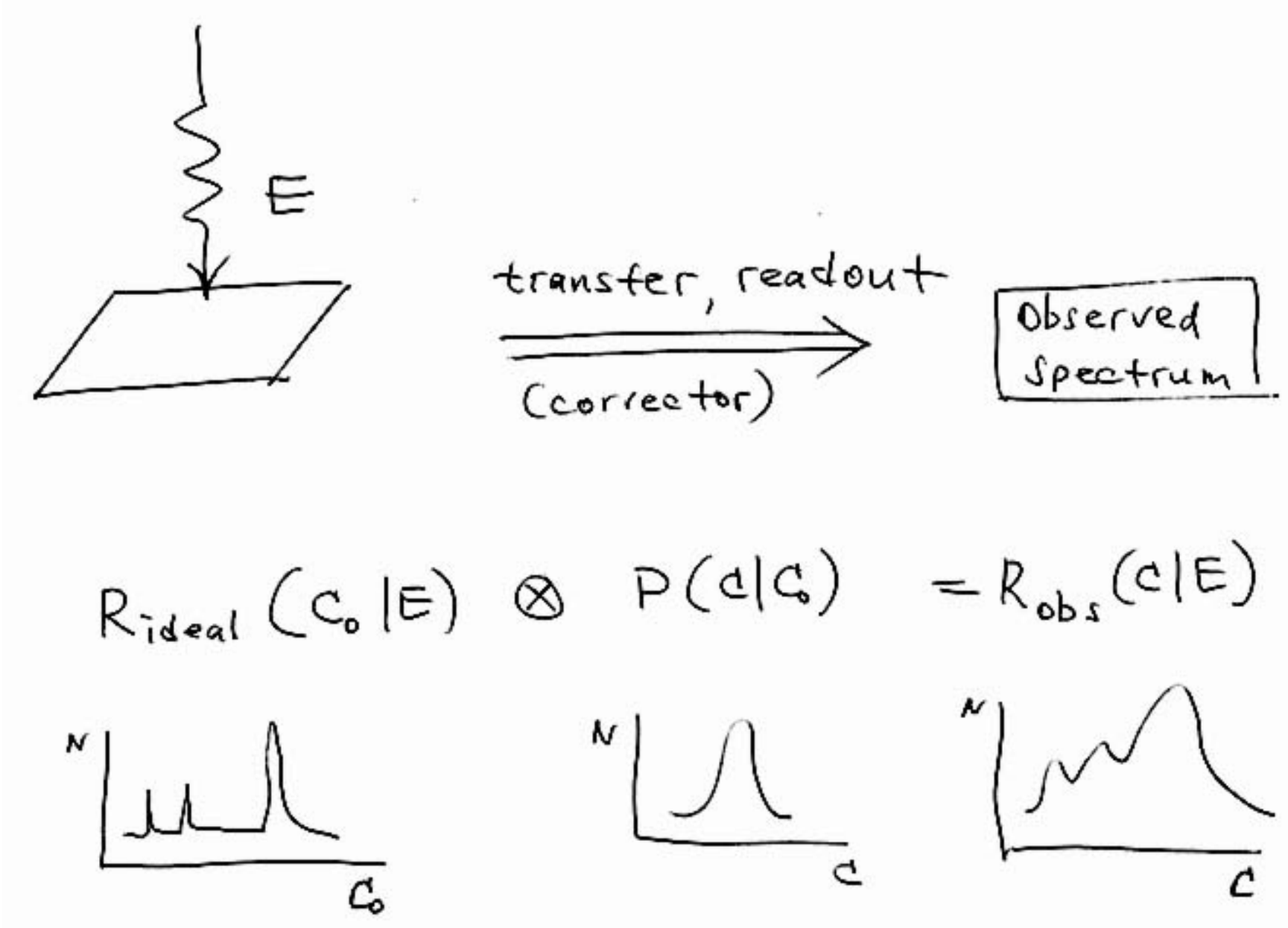
  Labor-intensive process. Fits are required in multiple positions (in FI chips, $32 \times 4 \times 8$). Tweaks have to be applied to match in-flight calibration.

- ***PSU corrector* approach**

  Directly simulate RMFs on a pre-defined energy grid.

  Computer-intensive process ("1100 hours of CPU time to generate RMFs"). Limits user's freedom.

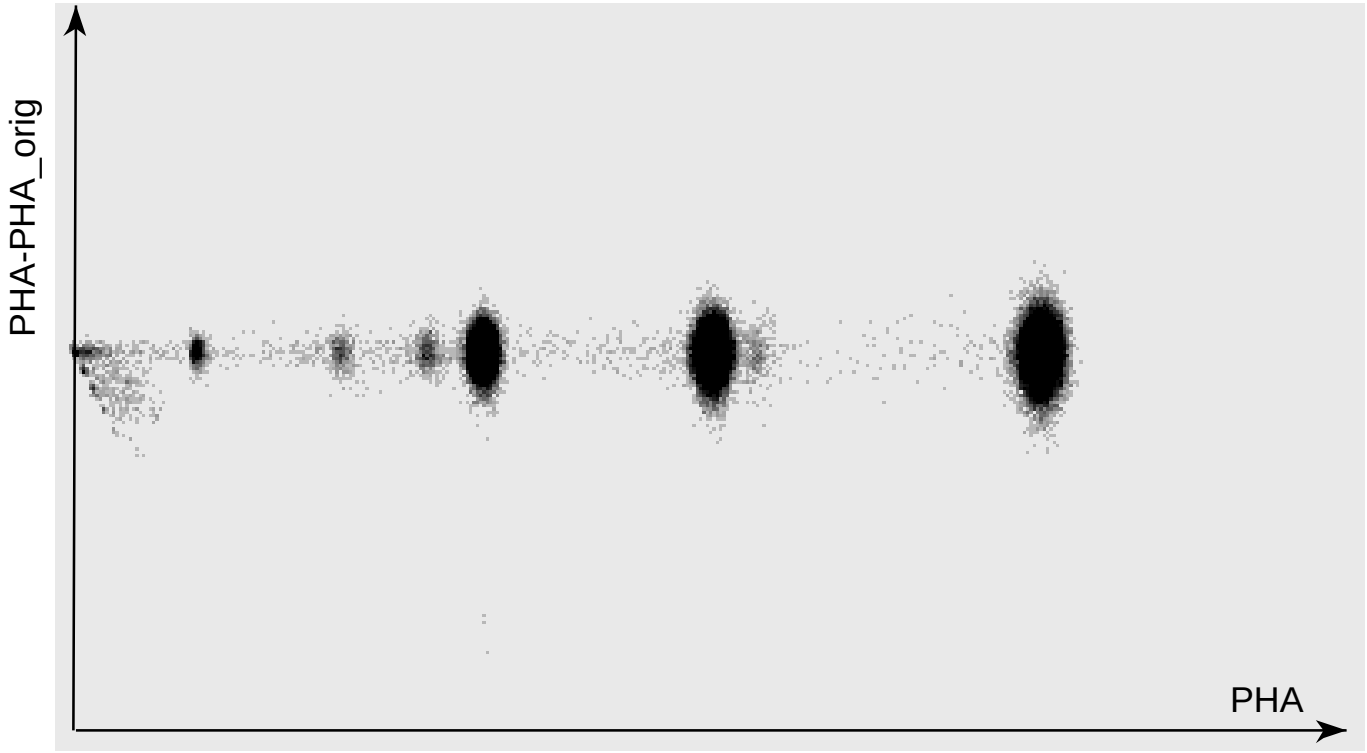
Separating response into two components



pre-CTI effect of CTI, corrector result

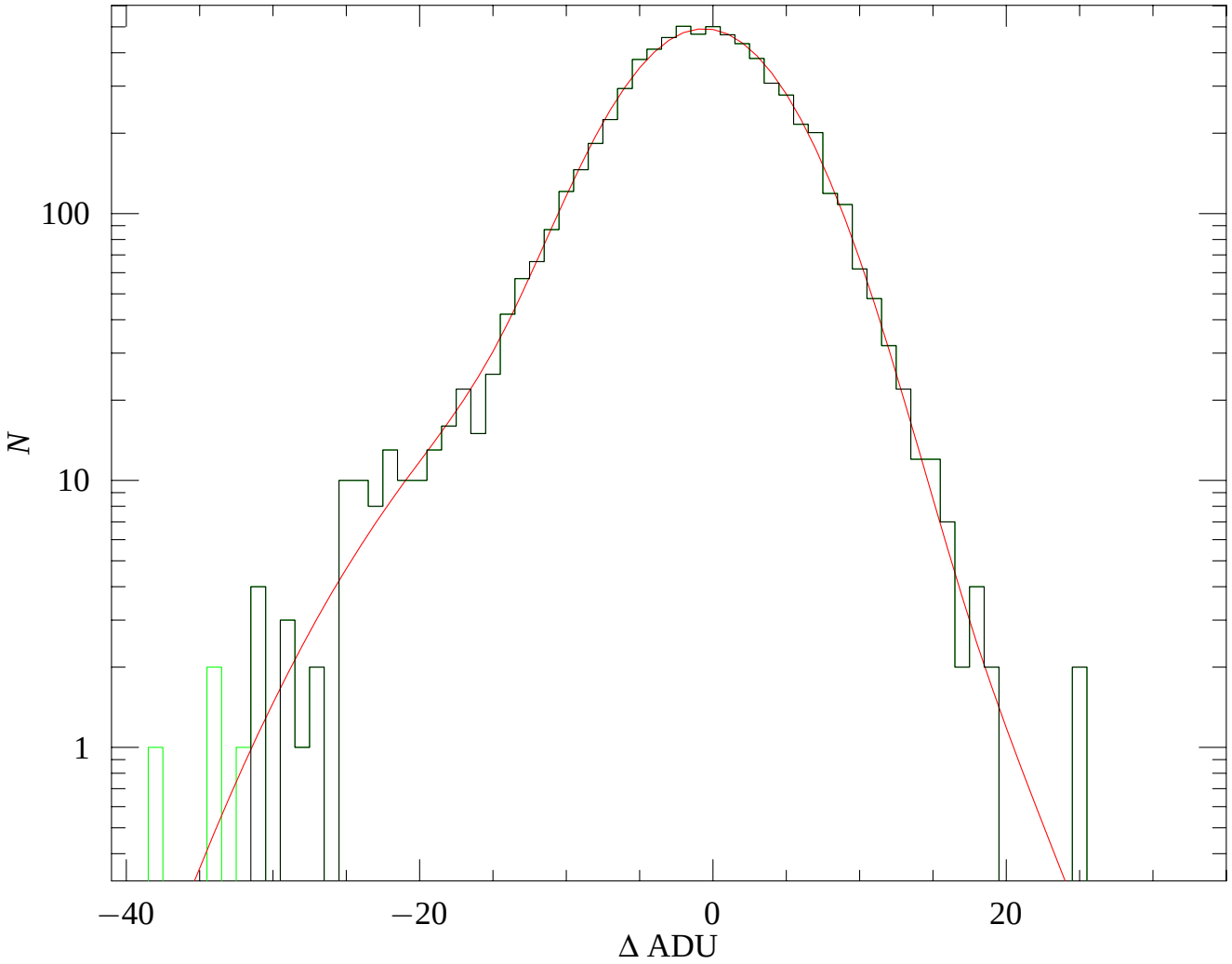
Separating response into two components

- **Pre-CTI response**
  - Complex, multicomponent function of energy;
  - BUT single matrix for all FI CCDs.
- **Effects of CTI**
  - Position-dependent;
  - BUT simple, unimodal, smooth function of PHA



CTI Scatter Matrix

- MIT ACIS Simulator has been run at 16 energies (0.25, 0.3, 0.4, 0.5, 0.7, 0.9, 1.2, 1.5, 1.8, 2, 2.6, 4, 6, 8, 10, 12 keV)
- Two-gaussian model always fits the simulated data well:



Making RMFs

pre-CTI response ⊗ CTI scatter = RMF

Global gain tweak

Position-dependent gain tweak; line width tweak.

How to compute RMFs:

- **Analytic convolution** to produce FEFs (Edgar's talk). Minor modifications to `mkrmf`. Fully tested (Schulz' talk). To be released on November 8.
- **Direct interpolation.** Fully automatic. Faster to compute. Small calibration files. Tests 50% complete. `calcrmf2` to be posted soon.