3.3 ANALYSIS

H2RG CHARACTERISATION METHODS

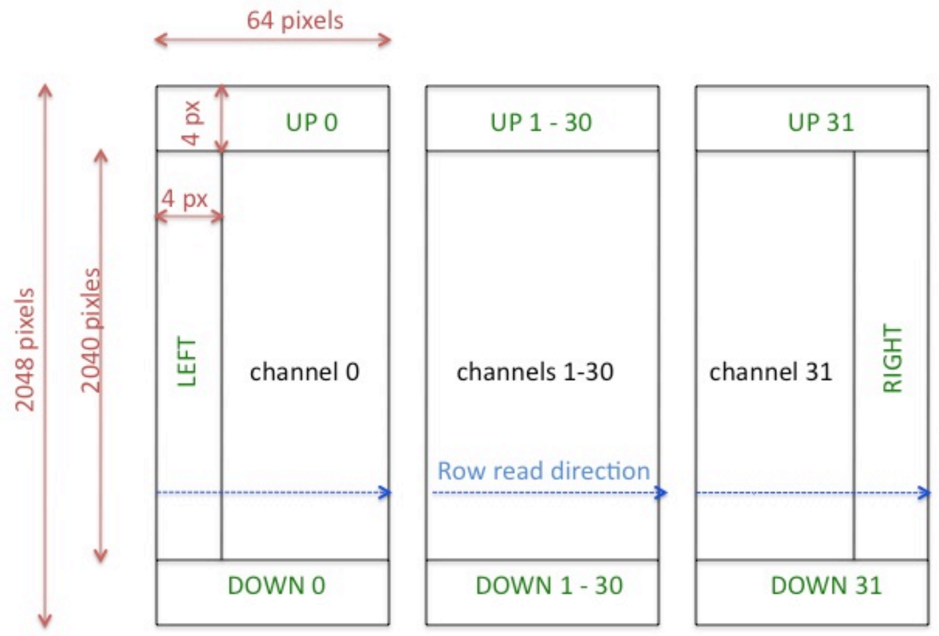
Bogna Kubik, analyst

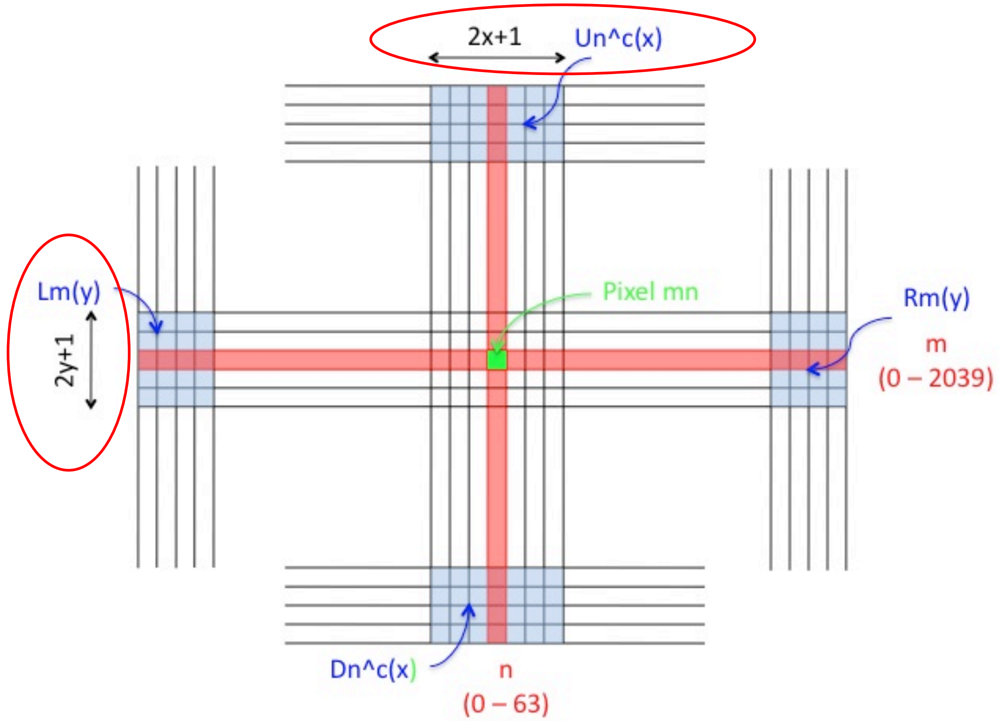
Goals

- Reference pixels correction:
  - Find and implement the optimal correction.
- Signal fit and anomaly detection:
  - Find an algorithm of signal fit and anomaly detection for on-board data processing (spectro) compliant with mission restrictions (CPU and telemetry).
- Nonlinearity correction:
  - Conception of the algorithm of nonlinearity correction for flight.
  - Define (minimal) input data to achieve required accuracy of 1% on nonlinearity correction.

REFERENCE CORRECTION (EUCL-IPN-TN-001)

Reference pixels - scheme





Up/Down = Top/Bottom → cover all channels.

Left/Right → only in the first and last channels.

Generic correction notation: $c_{mn}(x,y)$→ depends on the pixel (line, column) (m,n) and on the sliding windows size.

Optimal reference correction

- Minimizes CDS readout noise
  - So minimizes also total readout noise (spectro/photo) but the impact is lower.

Optimal sliding window size x and y

On top and bottom pixels:

c_{1n} (x) - no interpolation UP-DOWN

c_{3mn}(x) - with linear interpolation

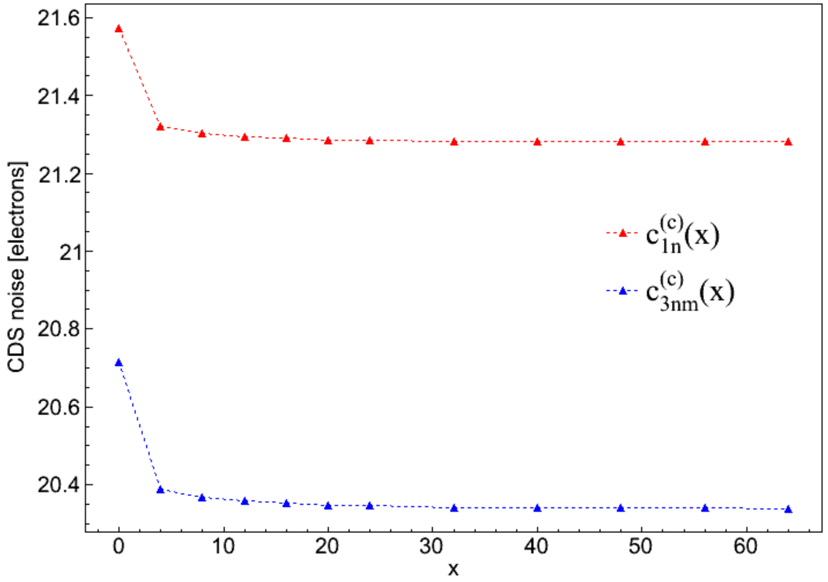
⟹ average all the channel

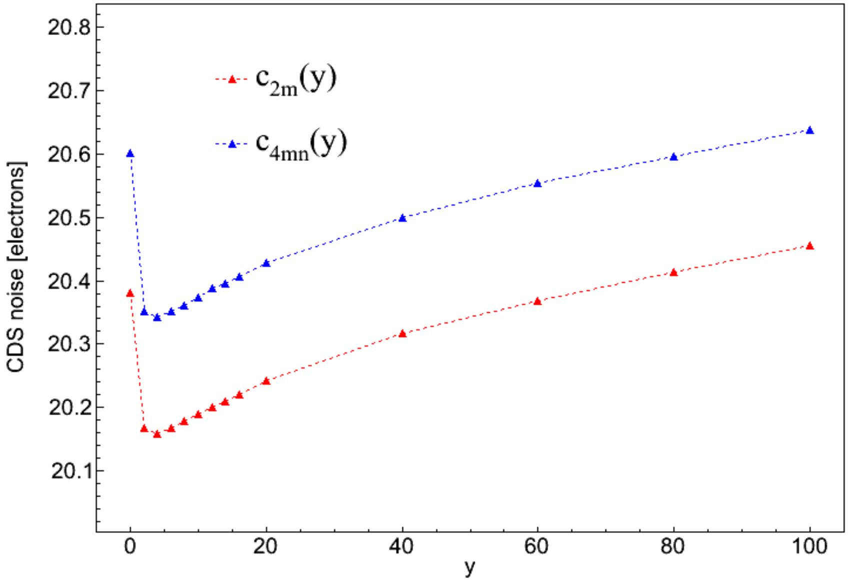
On left and right pixels:

c_{2m} (y) - no interpolation LEFT-RIGHT

c_{4mn}(y) - with linear interpolation

⟹ average over 9 lines





When the up-down and left-right corrections are mixed, the optimal sizes are the same.

All possible combinations were studied

- Correct first with Top/Bottom references
  - Left/Right are corrected in this step
  - Then correct with Left/Right references
- Compute the averages of Top/Bottom and Left/Right references independently and then correct the pixel
  - Left/Right pixels are NOT corrected with Top/Bottom.
- Interpolations
  - Between Top and Bottom
  - Between Left and Right
- In all studied cases the optimal sizes (x,y) are the same.

Optimal reference pixels correction

- Optimal reference correction (x=64, y=4) minimizes the CDS readout noise
  - Output-by-output correction using top and bottom reference pixels
    - Top/bottom pixels masked with median filter per output (|p-med| > 5 nmad)
    - Top/bottom pixels averaged per output
  - Line-by-line correction using rolling set of left and right reference pixels
    - Left/right pixels masked with median filter (|p-med| > 5 nmad)
    - Rolling set of left/right pixels averaged per corrected line

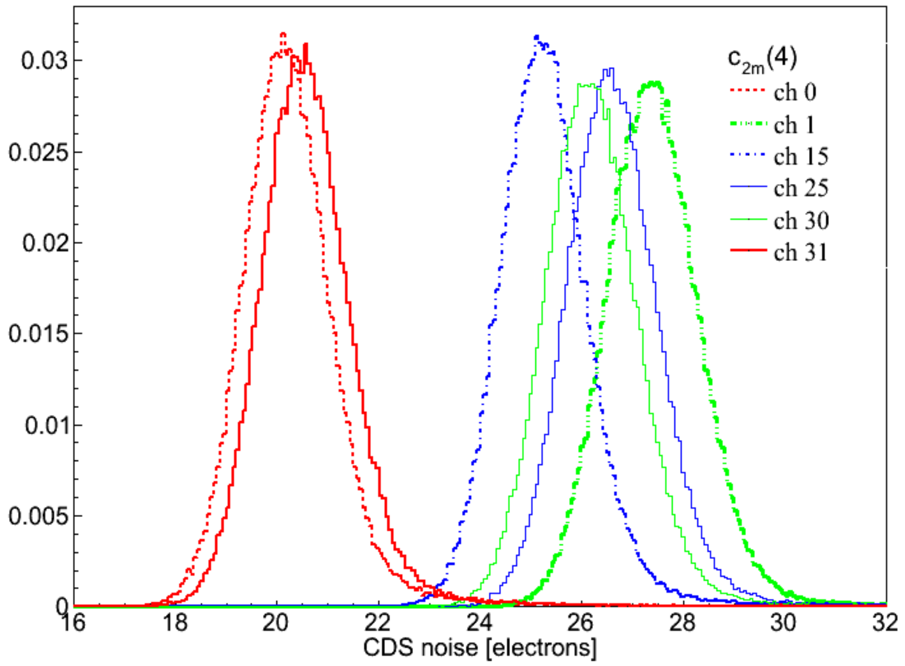
mode/correction	raw	$c_{1n}^{(20)}$	c_{2m}	$c_{3mn}^{(20)}$	c_{4mn}	$c_{5mn}^{(20)}$	$c_{6mn}^{(20)}$	$c_{7mn}^{(20)}$	$c_{8mn}^{(20)}$
CDS [e^-]	31.09	22.13	25.54	21.68	26.53	25.25	25.80	15.40	15.39
Fowler(16) [e^-]	17.30	5.55	6.87	5.49	7.11	13.18	13.19	4.32	4.32
MACC(3,16,11) [e^-]	23.67	6.56	7.67	6.53	7.91	17.94	17.94	5.38	5.38
MACC(15,16,9) [e^-]	21.61	10.69	10.83	10.67	10.89	19.23	19.22	10.39	10.39

c_{1n}: baseline correction implemented in warm electronics (2015). Uses only Up/Down references.

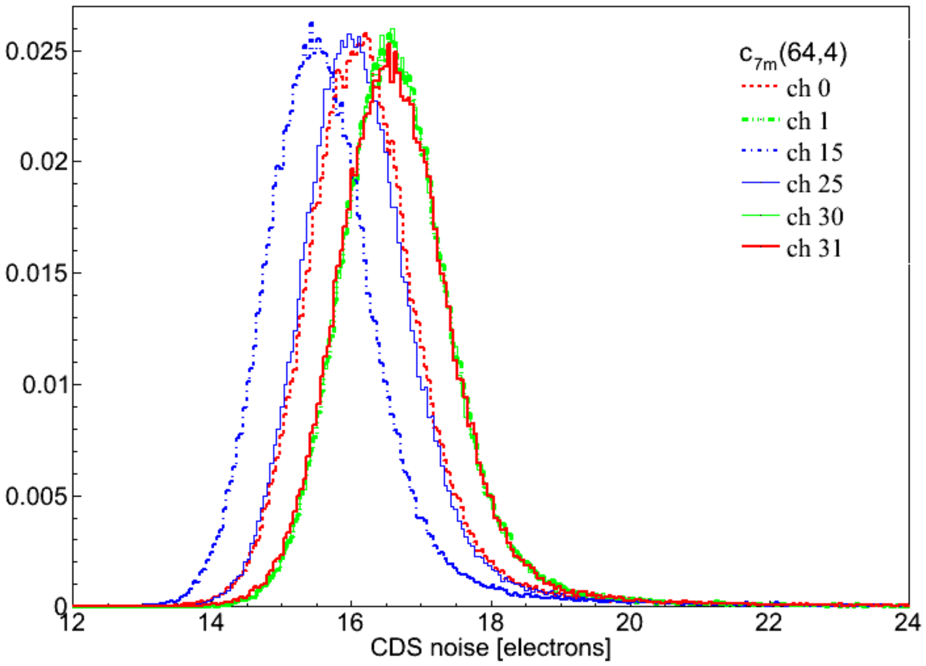
No channel position dependence

Correction with left/right pixels only

c_{2m} (4) corrects better the first and last channel



Optimal correction corrects all the channels equally well

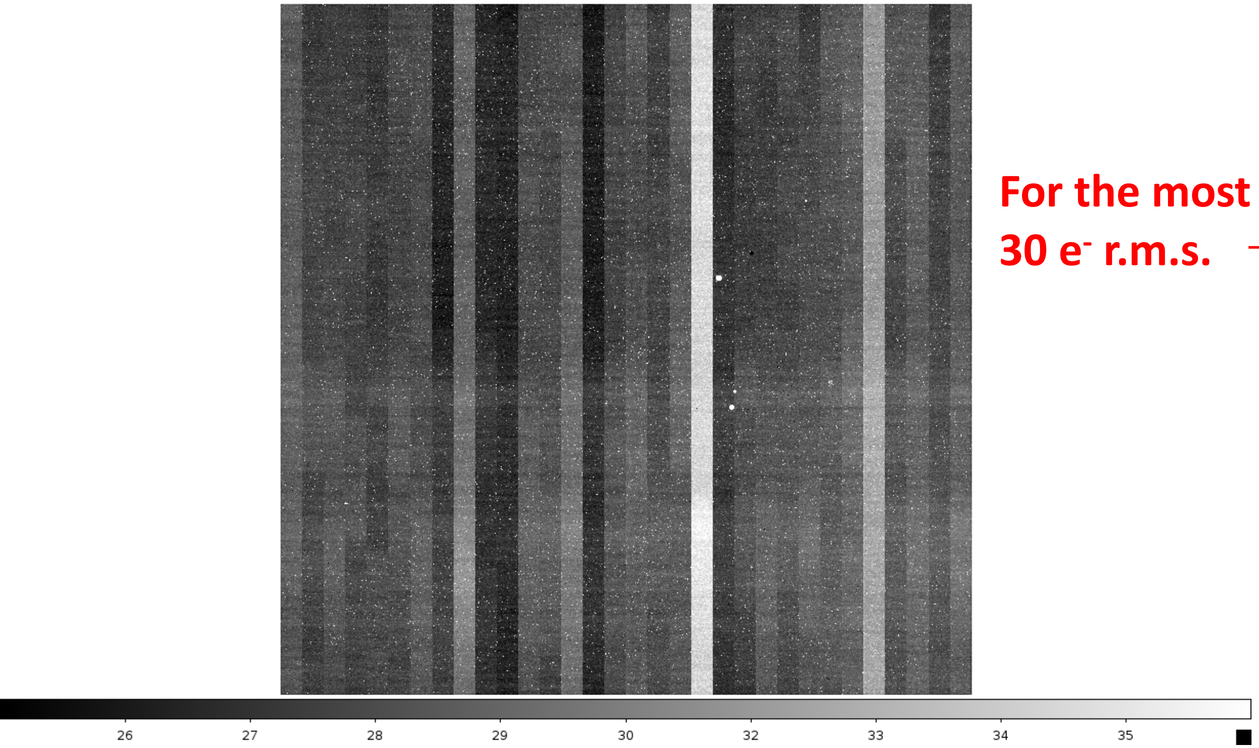


CDS noire image raw/corrected

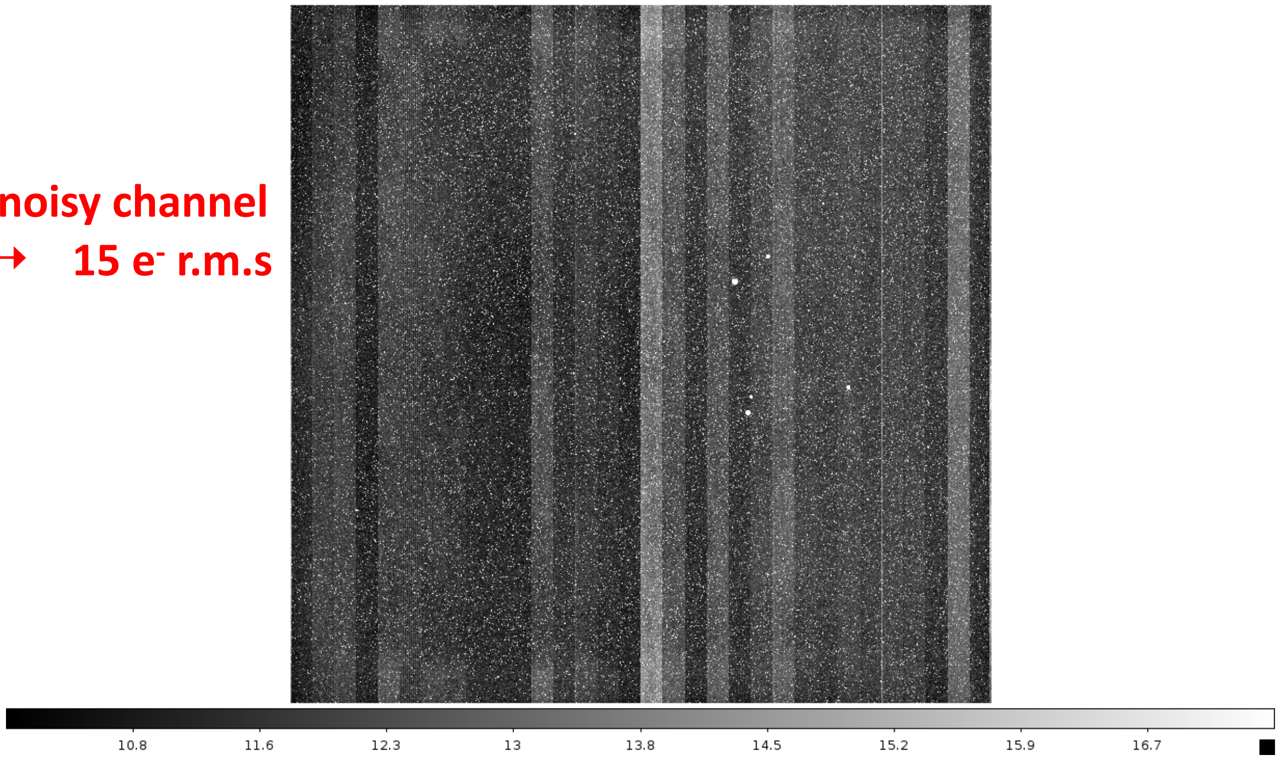
Raw CDS noise

Corrected CDS noise

image more homogeneous



For the most noisy channel
30 e⁻ r.m.s. → 15 e⁻ r.m.s



SIGNAL FITTING ALGORITHM

Need for a new signal estimator

- CPU and telemetry constraints:
  - Can not transfer all the coadded groups to the ground.
  - Signal fit and anomaly flagging on board.
  - Can not be assured by a simple LSF (too many iterations on data).

- Need for a new algorithm that estimates the signal and flags anomalies in a single iteration on data.

New signal estimator

- New algorithm of signal estimation and anomaly detection (QF) (Kubik et al. PASP 128, 968 (2016))
  - Proposed for the on-board Euclid-NISP spectrometric readout.
  - Used in ground testing as reference algorithm.

$$\hat{g} = \frac{1+\alpha}{2f_e}\left[\sqrt{1+\frac{4f_e^2\sum_{i=1}^{n_g-1}(\Delta G_i+\beta)^2}{(n_g-1)(1+\alpha)^2}}-1\right]-\beta$$

$$QF = \frac{2f_e}{(1+\alpha)}[(n_g-1)\hat{g_x}-(G_n-G_1)]$$

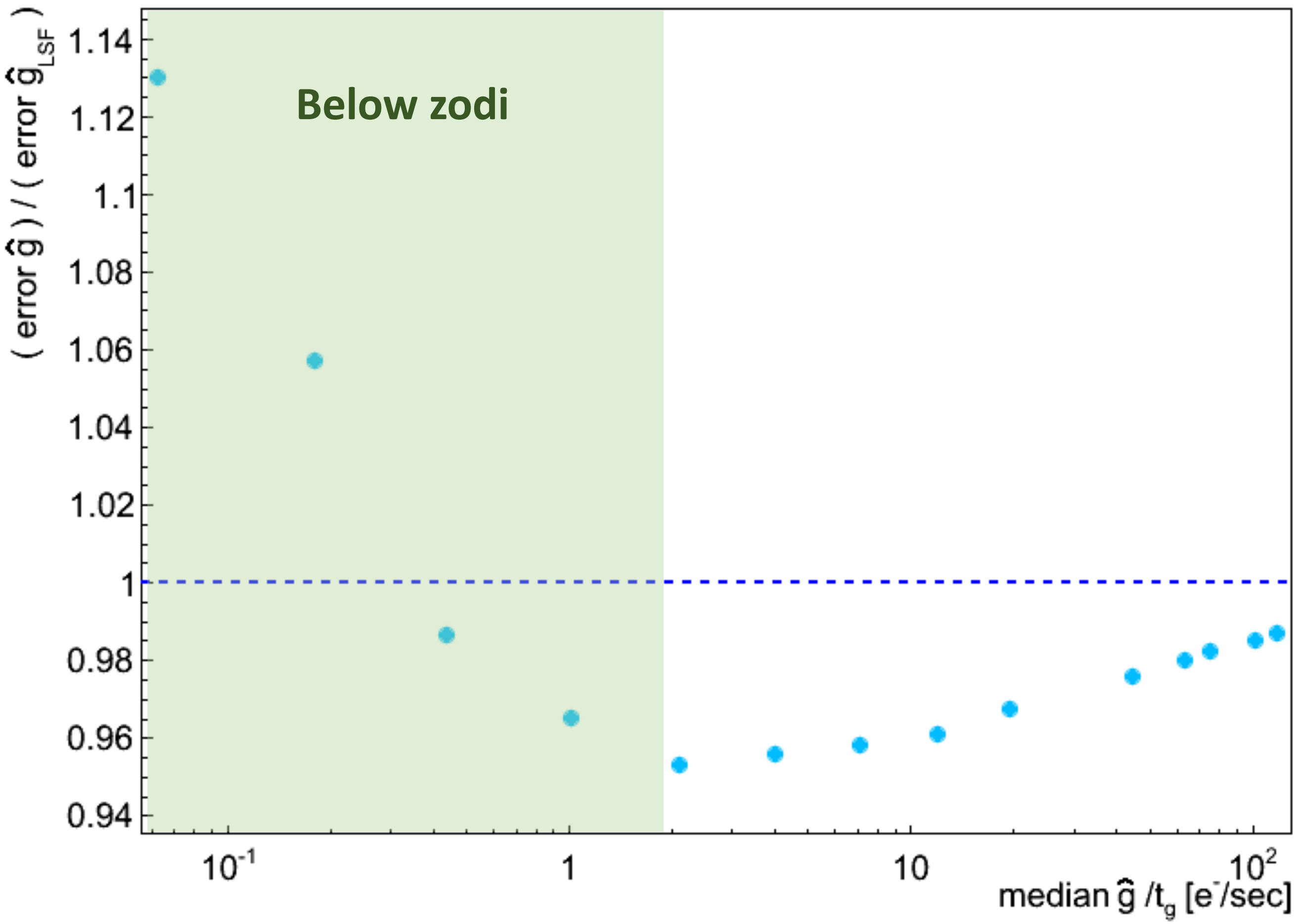
- Technical / mathematical points new in this method:
  - Takes into account the correlations between reads from coadding (coefficient α - same for all pixels, RO mode dependent)
  - Assumes Poisson noise (not Gaussian as commonly used)

Signal fit algorithm - advantages

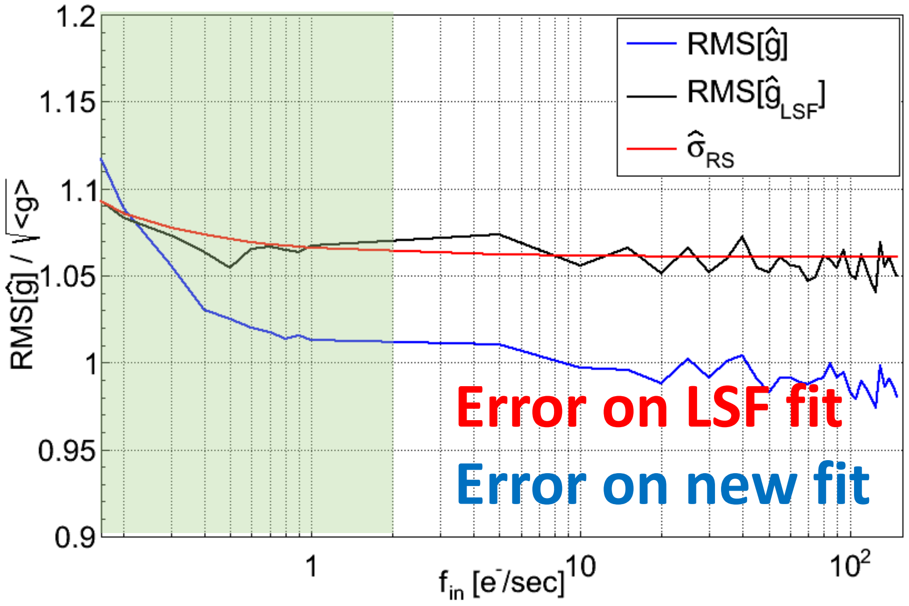
- Error on the flux lower than least square fit (LSF)
- Provides a quality factor (QF) for anomaly detection compliant with CPU restrictions: QF computed in the same iteration on data that the flux estimate.

Advantage 1: Flux error lower than LSF

Data



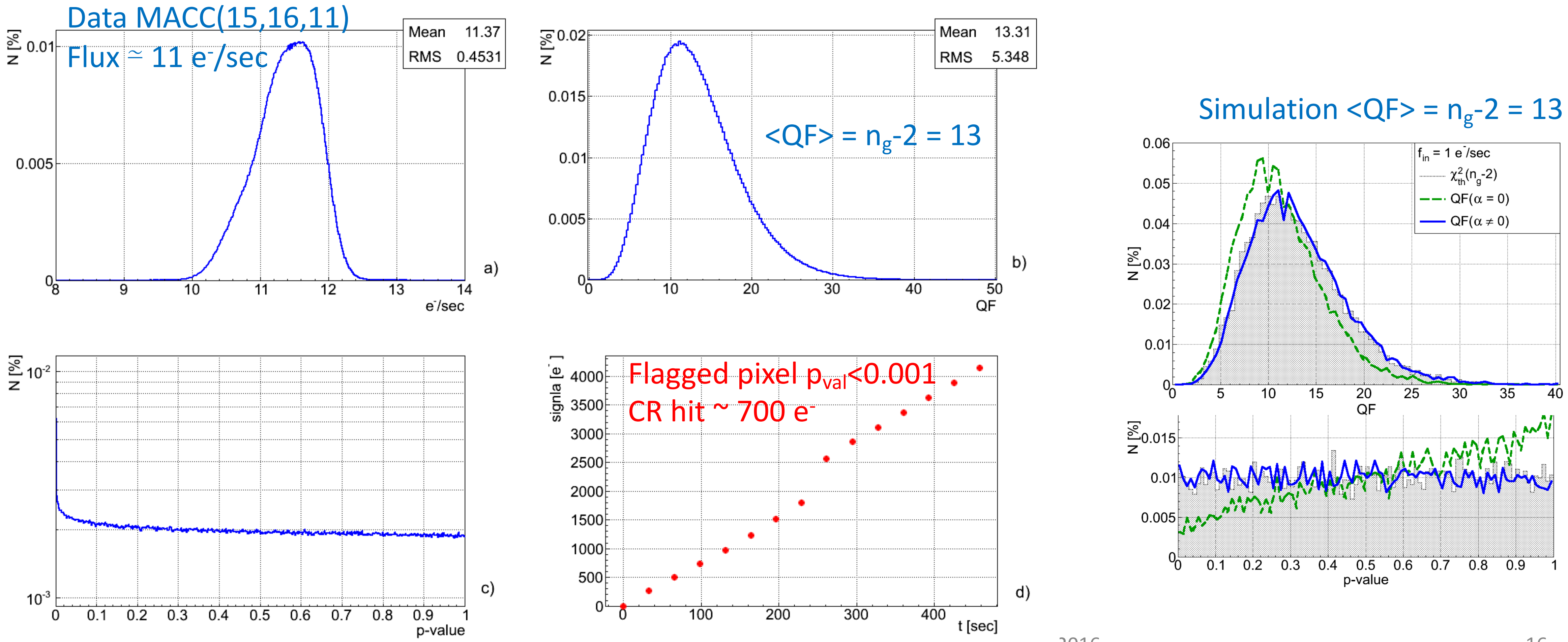
Simulation



With the new method the error ~5% lower than with LSF.

Advantage 2: anomaly detection

- Quality factor (QF) computation compliant with CPU restrictions



Signal fit algorithm - issues

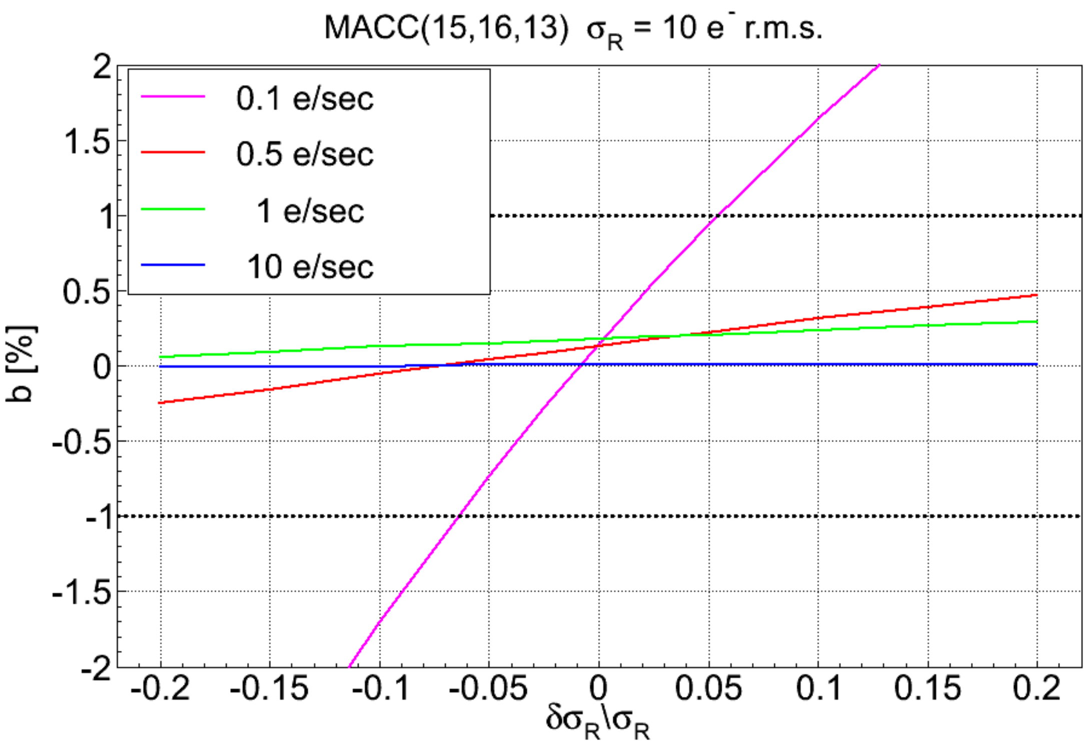
- Sensitive to detector properties (readout noise σ_R, gain f_e)
  - Need to measure the pixel properties with good accuracy (better than 20%).
  - Probably need to monitor and update them during flight.

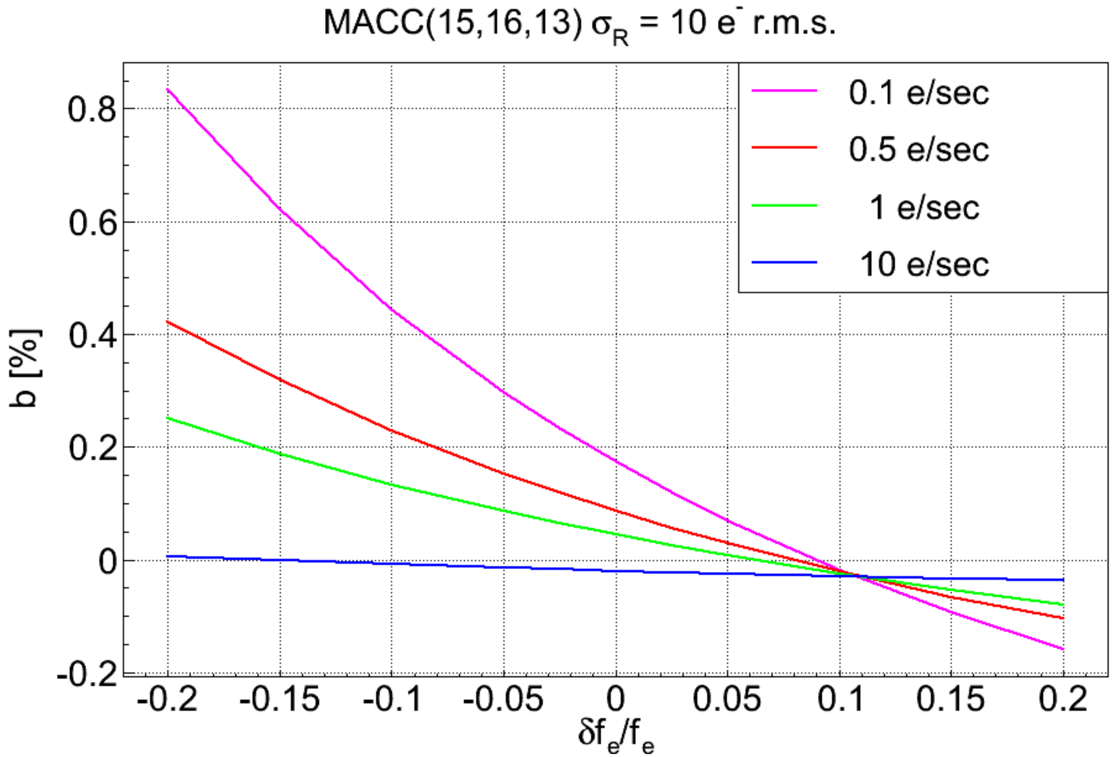
- Sensitive to nonlinearity (must be corrected before anomaly flag)
  - Flux and quality factor estimated in orbit must be corrected on ground for nonlinearity effects.

Sensitivity to detector properties

- Flux estimator is sensitive to the readout noise and gain of the integrating array

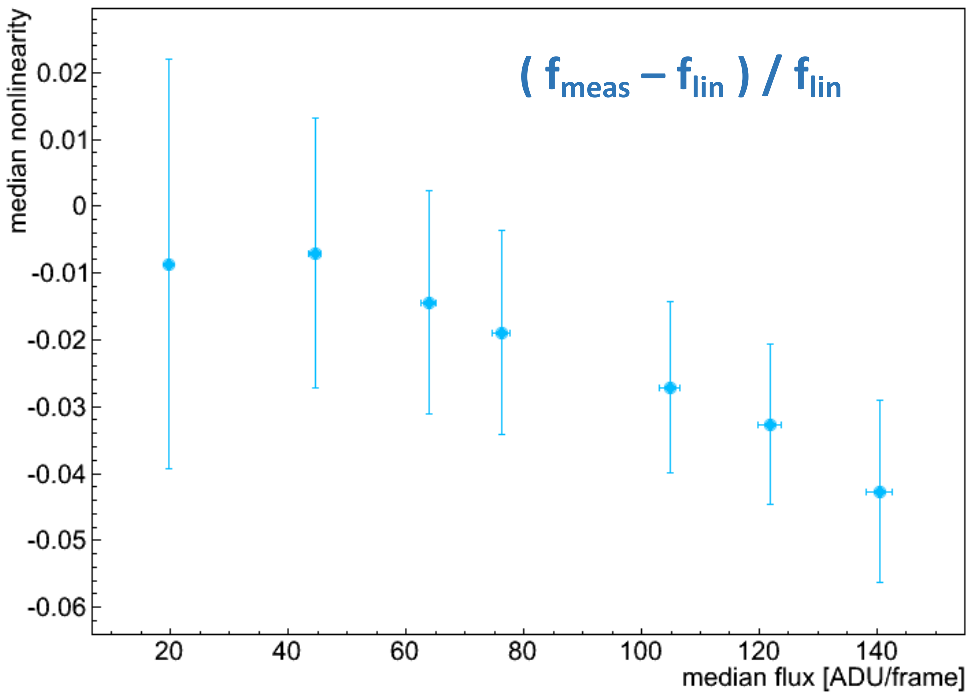
  ⟹ Possible biases if an average value is used on board for pixels in the tails of the distributions of σ_R and f_e (bias lower than 1% for scientific fluxes).

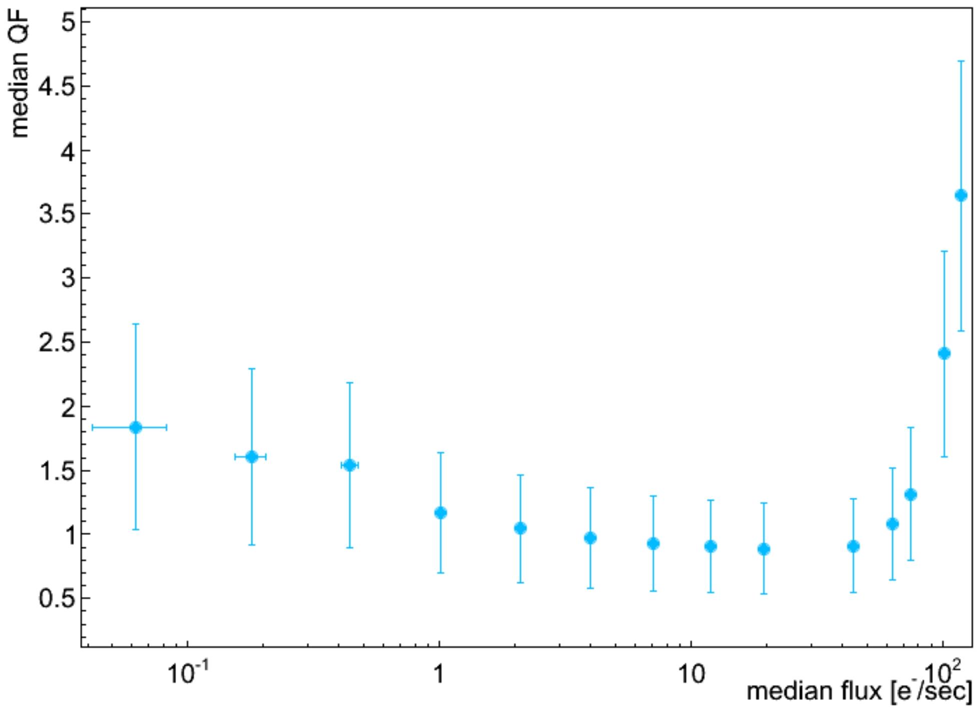




Sensitivity to nonlinearity (data)

- On ramps not corrected for nonlinearity, with increasing illumination:
  - Measured flux is lower than expected.
  - Normalized (divided by n_g-2) QF is higher than expected.

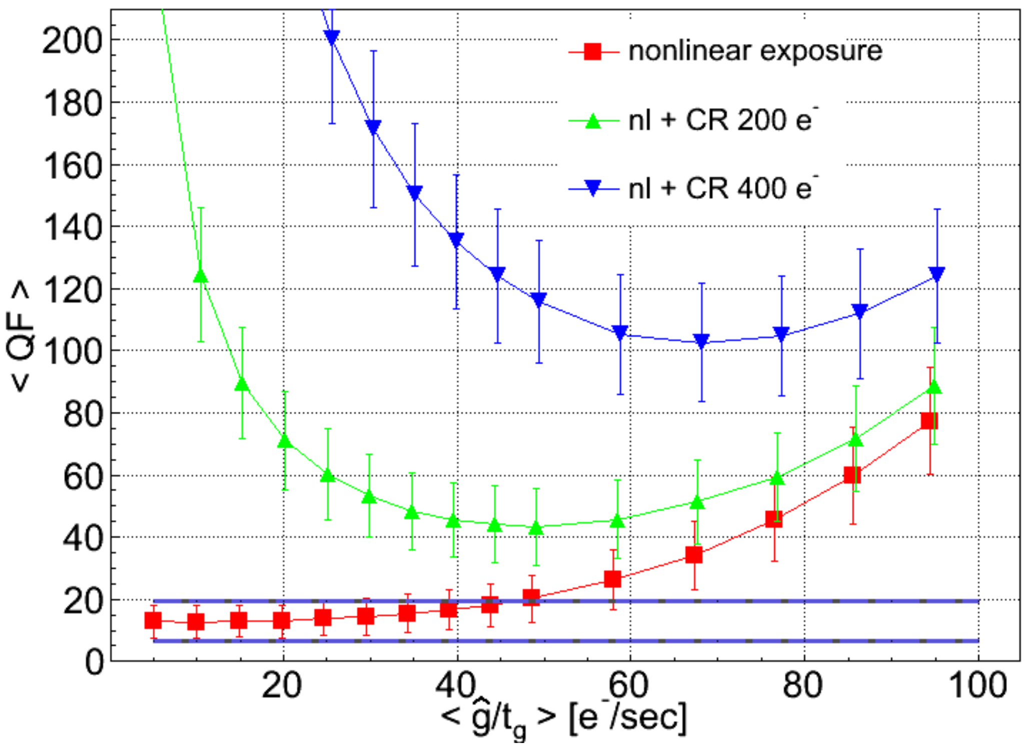




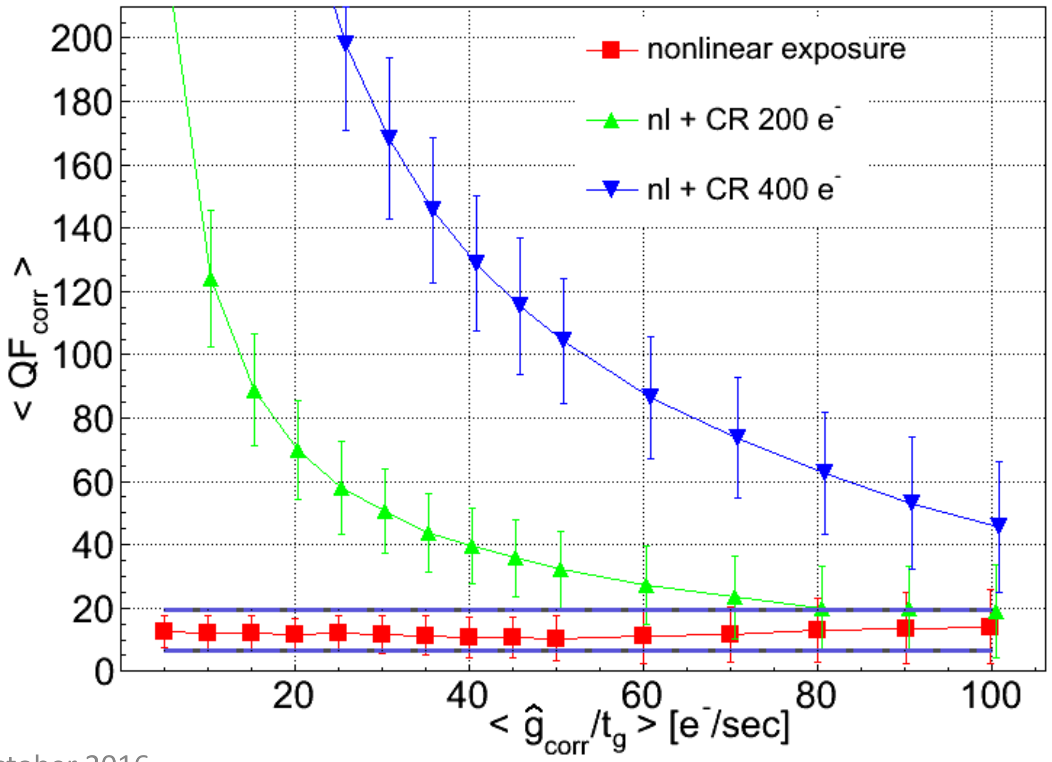
Sensitivity to nonlinearity (simulation)

- Both: flux estimator and QF are sensitive to the nonlinear response
  - QF must be corrected prior to anomaly flagging
  - Flux must be corrected anyway

Simulation: before correction for nonlinearity



Simulation: after correction for nonlinearity

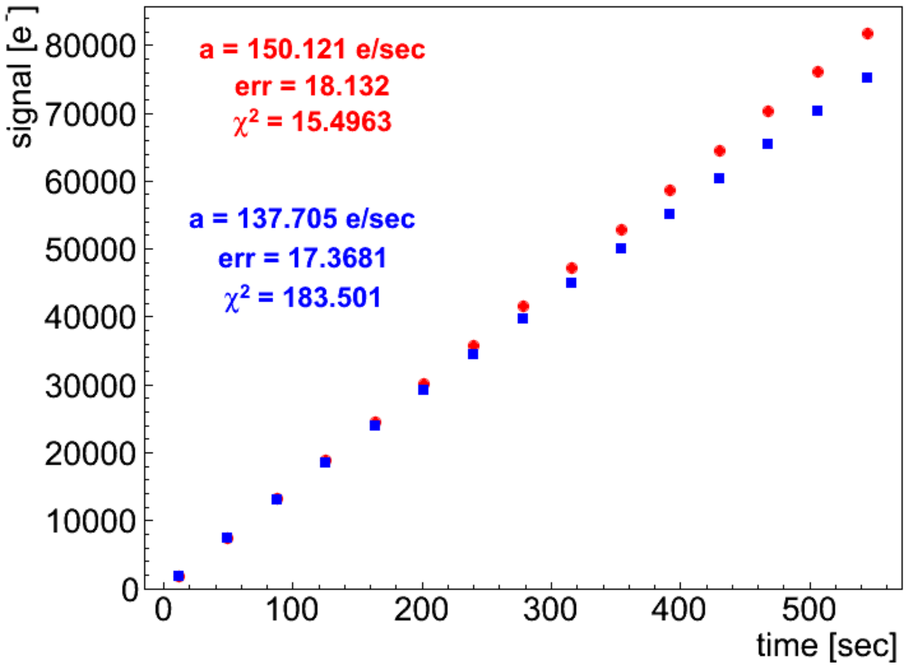


NONLINEARITY CORRECTION
example with spectro MACC(15,16,11)

Nonlinearity correction – principle

- Goal: construct maps of coefficients to correct the FIT to the response of the pixels (g)

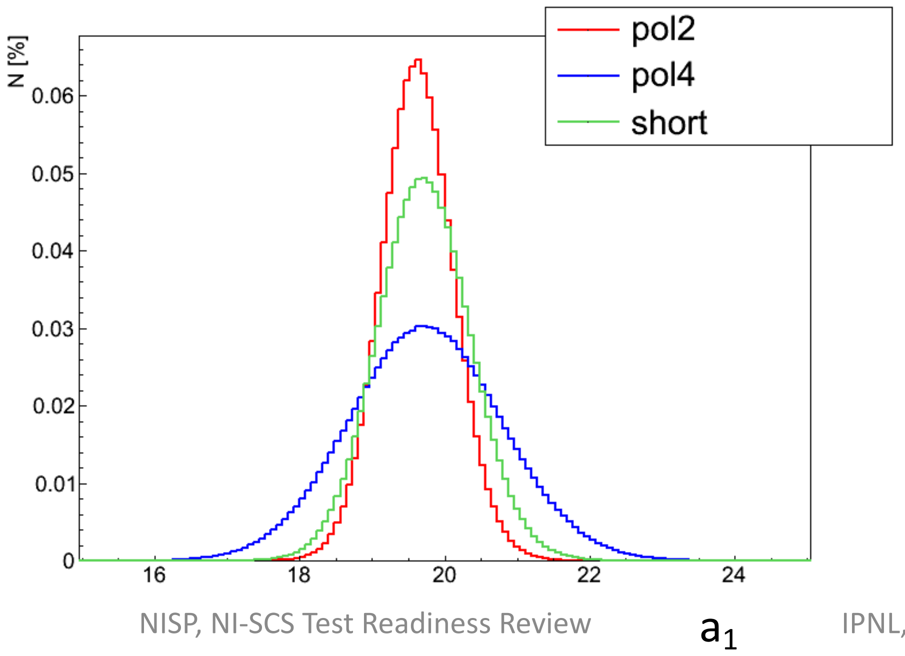
- Example:
  - In absence of anomalies the fit on board will give
    - g_{nl} = 137.705
    - QF_{nl}= 183
  - Must be corrected to
    - g_{lin} = 150.121
    - QF_{lin}= 15

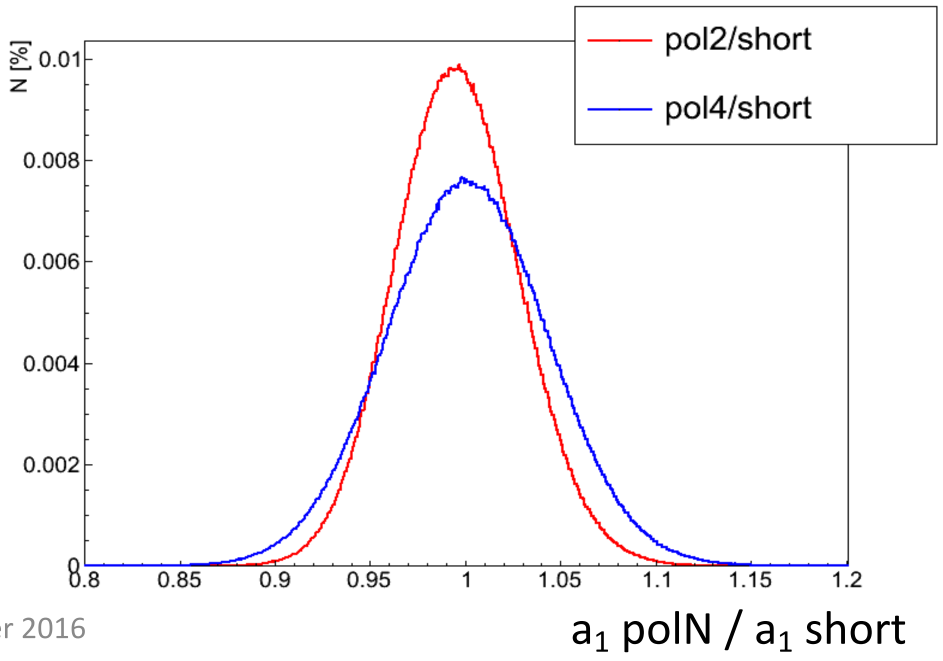


- On the ground: construct the maps $g_{nl} \rightarrow g_{lin}$ and $QF_{nl} \rightarrow QF_{lin}$ for all the range of fluxes (science + calibration) with precision $\leqq$1%.

How to define the true linear flux?

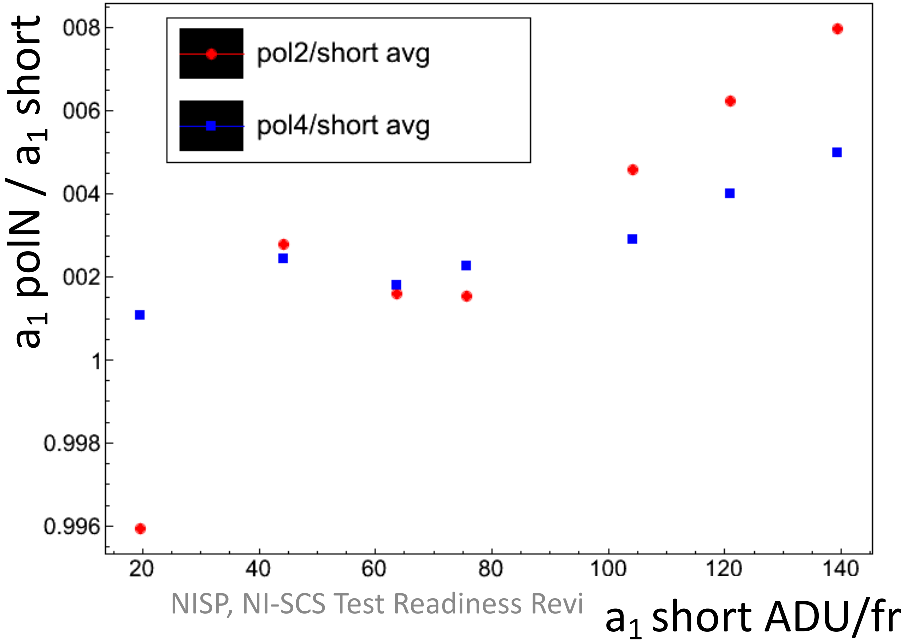
- Result of a polynomial fit to the ramp a_1
  - $Y(t) = a_0 + a_1 t + a_2\, t^2$ (POL2)
  - $Y(t) = a_0 + a_1 t + a_2\, t^2 + a_3 t^3 + a_4\, t^4$ (POL4)
- Result of the linear fit to the beginning of the ramp (5 groups of 10 frames coadded)
  - $Y(t) = a_0 + a_1 t$ (SHORT)

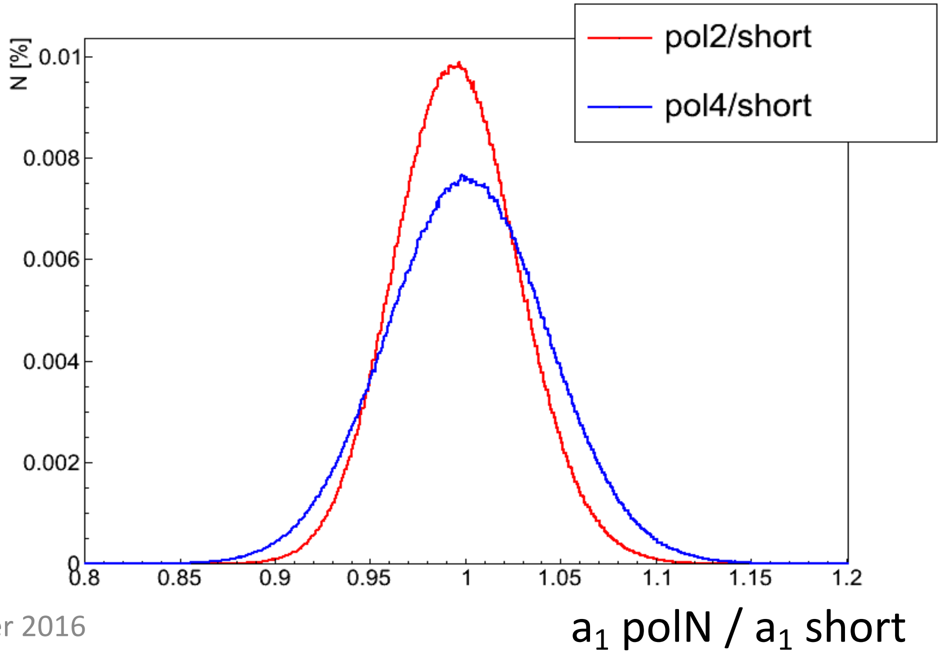




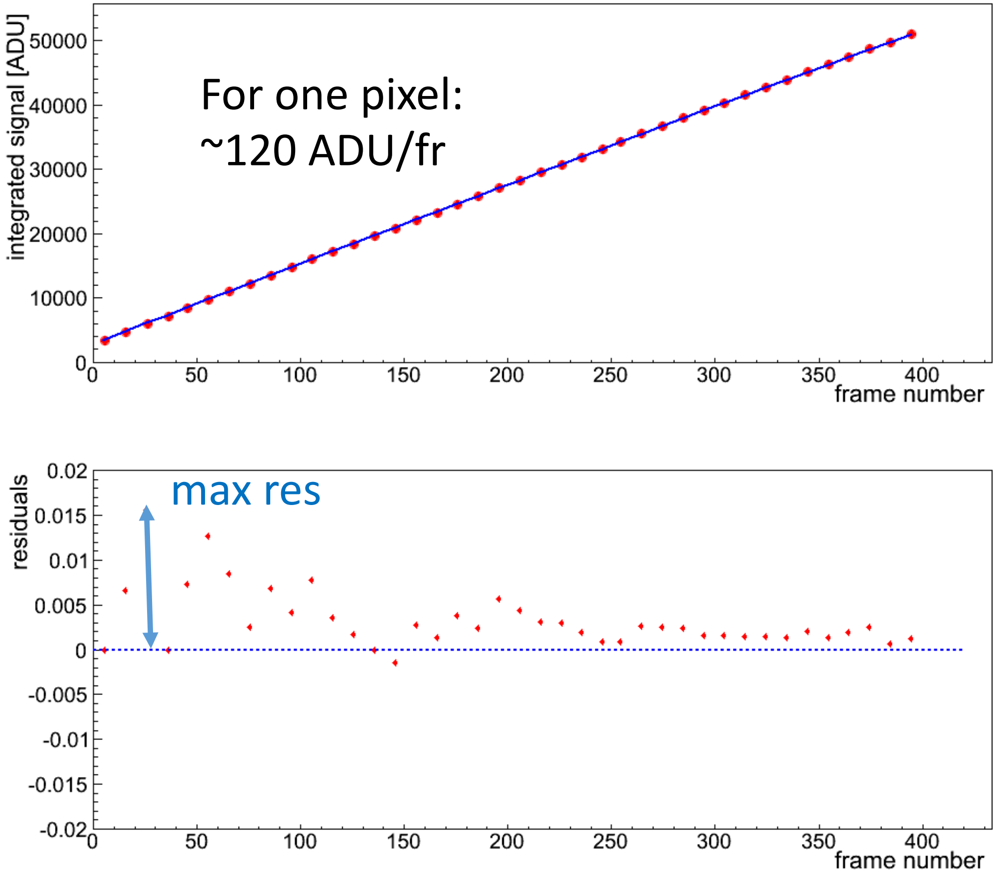
How to define the true linear flux?

- The three definitions are not equivalent
- Flux-dependent bias
- Which of them shall be used as reference linear flux?
  - Compare a_0 and check compatibility with baseline
  - Compare residuals to fit

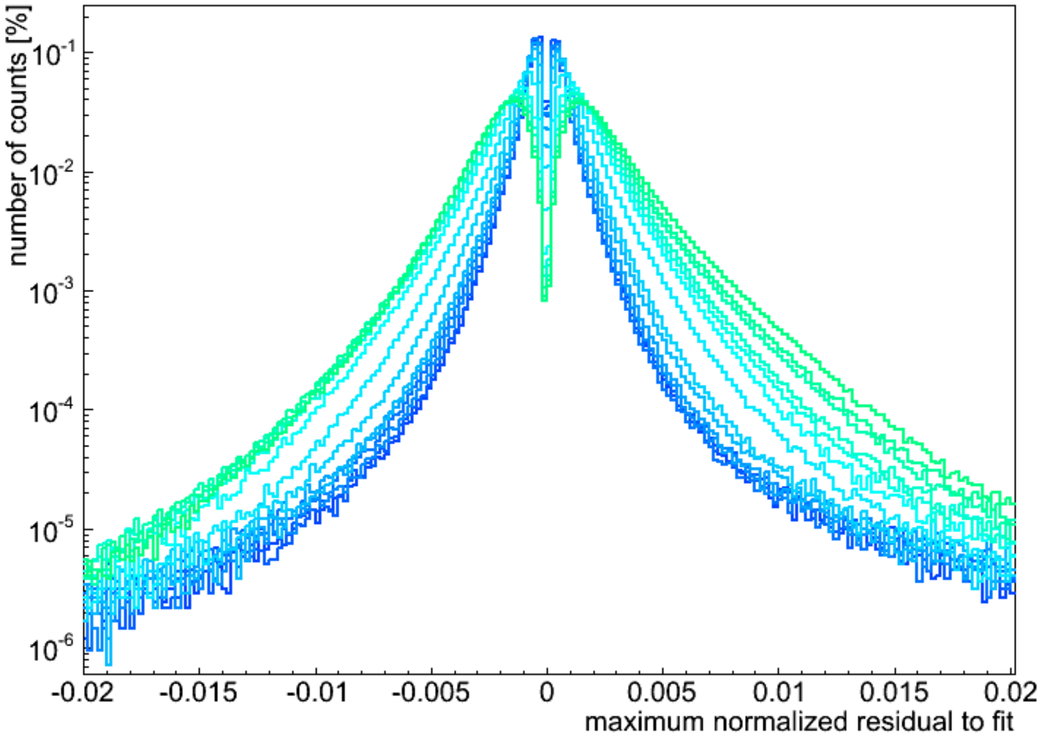




Maximum residual to the fit



Distribution for all pixels as function of illumination (different colors):



Median |maximum residual| to the fit

When comparing residuals:

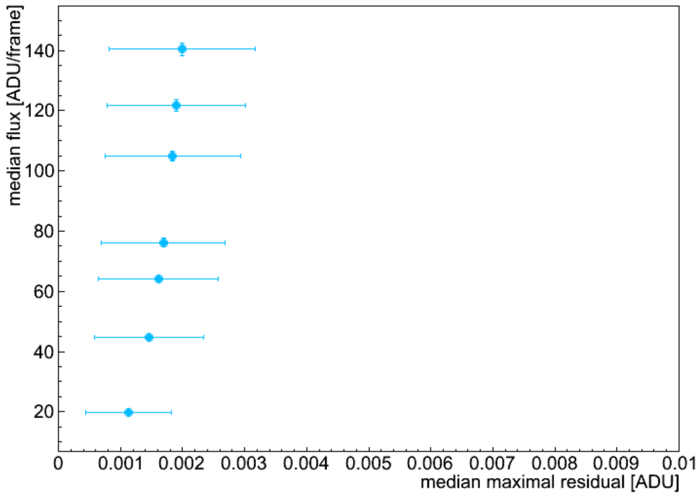
pol(N) were computed on whole the ramp (minus discarded points)

short fit only on 5 groups

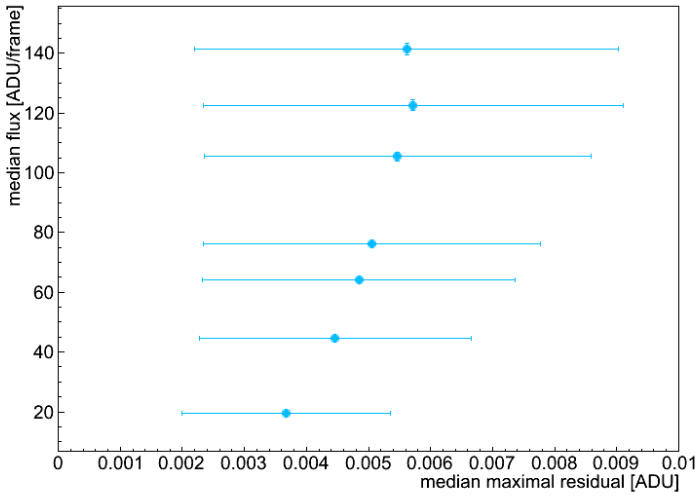
⇨ Residuals comparison is not straight forward.

⇨ Short fit has lowest residuals in the used range (50 frames).

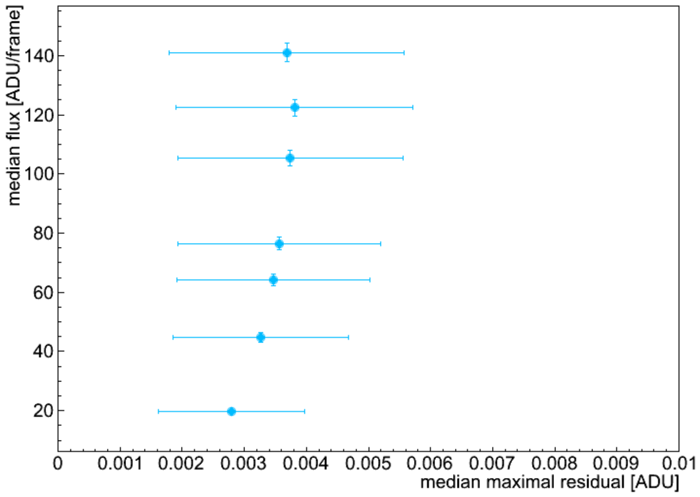
Short fit



Pol 2



Pol 4



Which fit to estimate the linear expected flux?

How many (%) pixels have

a) residual > 1%

b) a_0 incompatible with baseline

Flux [ADU/fr]	Percent of masked pixels					
	| max res | > 0.01			| a0 - baseline| > 5 kTC noise		
	short	pol2	pol4	short	pol2	pol4
20	0.23	5.5	1.6	0.07	0.18	0.11
45	0.34	10.6	3.4	0.11	0.37	0.20
65	0.34	13.7	4.6	0.18	0.86	0.42
75	0.49	15.5	5.3	0.21	1.23	0.57
100	0.63	19.2	6.7	0.71	2.93	1.99
120	0.71	21.5	7.4	1.14	3.52	2.91
140	0.86	21.6	7.4	2.13	4.43	3.94

Which fit to estimate the linear expected flux?

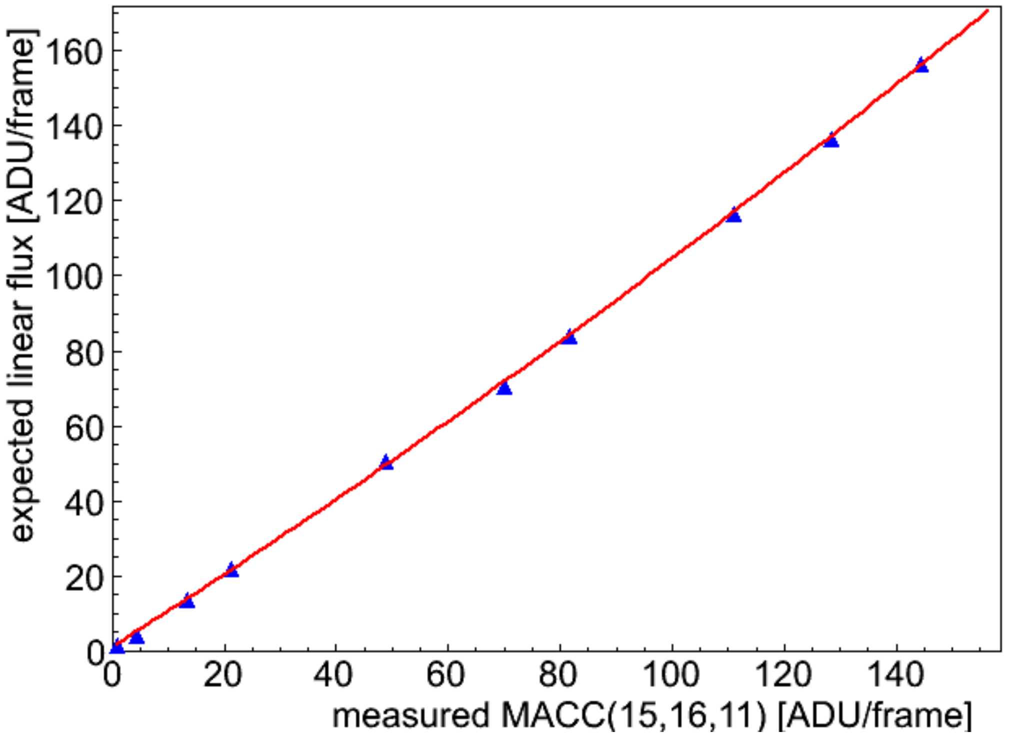
- In the range of fluxes 20 – 140 ADU/frame (equiv. to 20 – 140 e/sec)
  - All the dynamical range was explored without hard ADC saturation with exposures UTR(400)
  - If 95% of pixels must have a0 compatible with baseline and residuals < 1% then
  - Poly 2 fits the ramps up to 20 e/sec
  - Poly 4 fits the ramps up to flux 70 e/sec
  - Short fit to first 5 groups is a good estimator of the linear flux over 20-140 e/sec range.
- We take the short fit as the estimate of the expected linear flux.

Look-up tables construction

- Lookup table is a function that maps the flux measured in flight (on non-corrected exposures) to the expected linear flux.
- Used for the flight
- Fluxes used to construct 3 lookup tables:

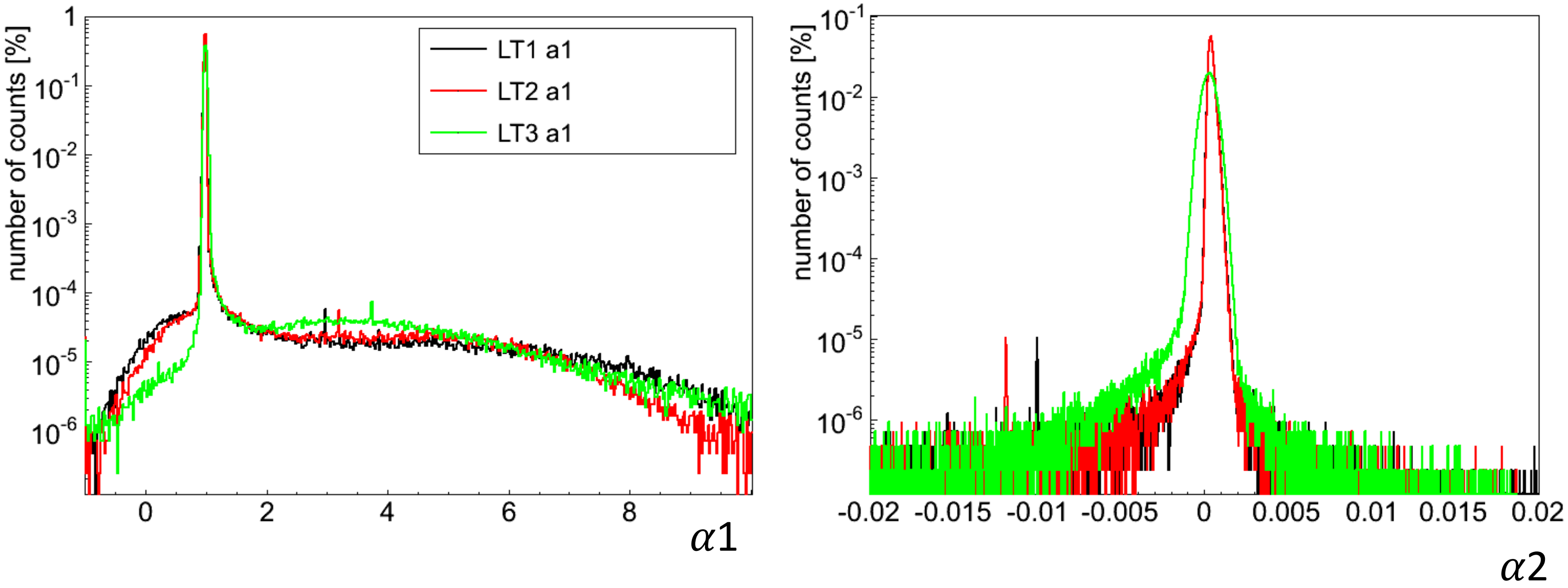
ADU/fr	1	4	10	20	45	65	75	100	120	140
LT1				✔	✔	✔	✔	✔	✔	✔
LT2	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
LT3		✔	✔	✔	✔	✔				

Look-up table fit example



- Look-up table per pixel fit:
  - $F_{lin} = \alpha_0 + \alpha_1 F_{MACC} + \alpha_2 F_{MACC}^2$
- And correction:
  - $\alpha_0 + \alpha_1 F'_{MACC} + \alpha_2 F'_{MACC}{}^2 \rightarrow F_{corr}$

Look-up tables – results {$\boldsymbol{\alpha}_1$, $\boldsymbol{\alpha}_2$}

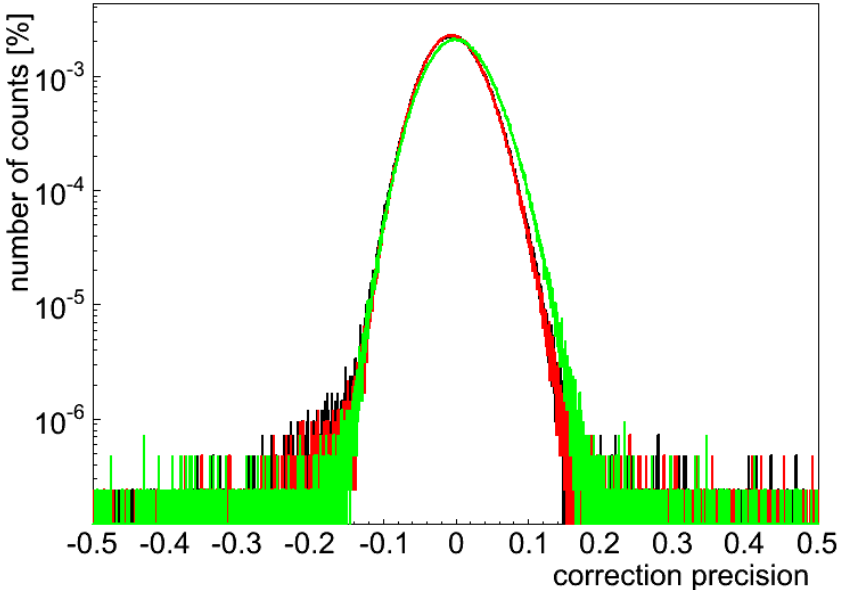


ADU/fr	<α_1>	<fit err α_1>	RMS[α_1]	<α_2>	RMS[α_2]	<fit errα_2> 10^3
LT1	1.01	0.01-0.02	0.48	0.0004	0.0005	0.1
LT2	1.01	0.01-0.02	0.46	0.0004	0.0005	0.1
LT3	1.03	0.02-0.04	0.51	0.0003	0.0007	0.5-0.8

- Look-up tables differ depending on used fluxes but the differences are within the fit errors (for average values)

Nonlinearity correction bias ε

- For f = 20 e/sec $\Longrightarrow$ S = $f*t_{exp}$ = 10 000 e $\Longrightarrow$ σ = 1% (Poisson)
- Correction bias for the majority of pixels ~ 0.3% $\Longrightarrow$ within Poisson error.

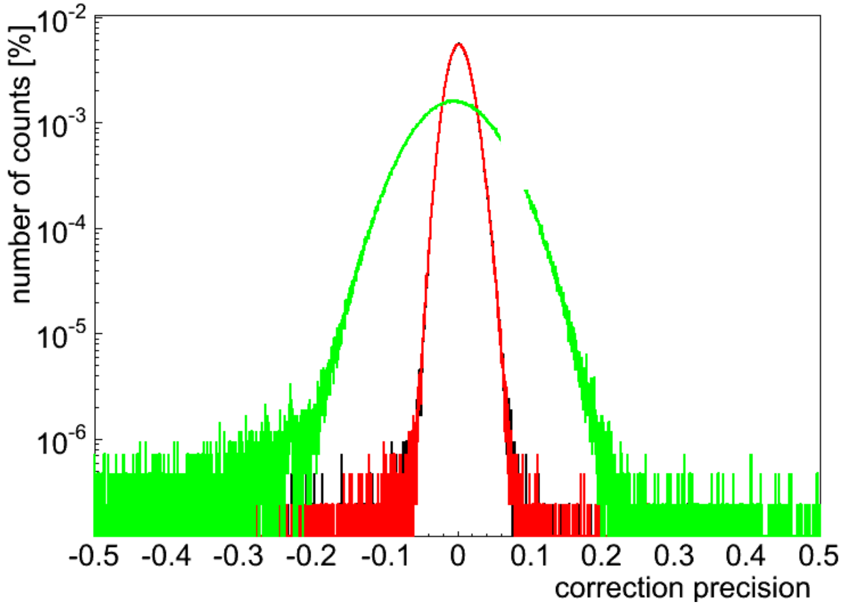


$$\varepsilon = (f_{corr} - f_{lin}) / f_{lin}$$

ADU/fr	$<a_{corr}>$	$RMS[a_{corr}]$	$<\varepsilon>$	$RMS[\varepsilon]$
LT1	19.45	4.66	-0.0038	0.0361
LT2	19.45	4.76	-0.0037	0.0357
LT3	19.58	4.82	0.0025	0.0386

Nonlinearity correction bias ε

- For f = 120 e/sec $\Longrightarrow$ S = f*t_{exp} = 60 000 e $\Longrightarrow$ σ = 0.4% (Poisson)
- Correction bias for the majority of pixels ~ 0.3% $\Longrightarrow$ within Poisson error.
- What is the precision on $\boldsymbol{\varepsilon}$? (It is not the spatial RMS of the ε distribution over the image)



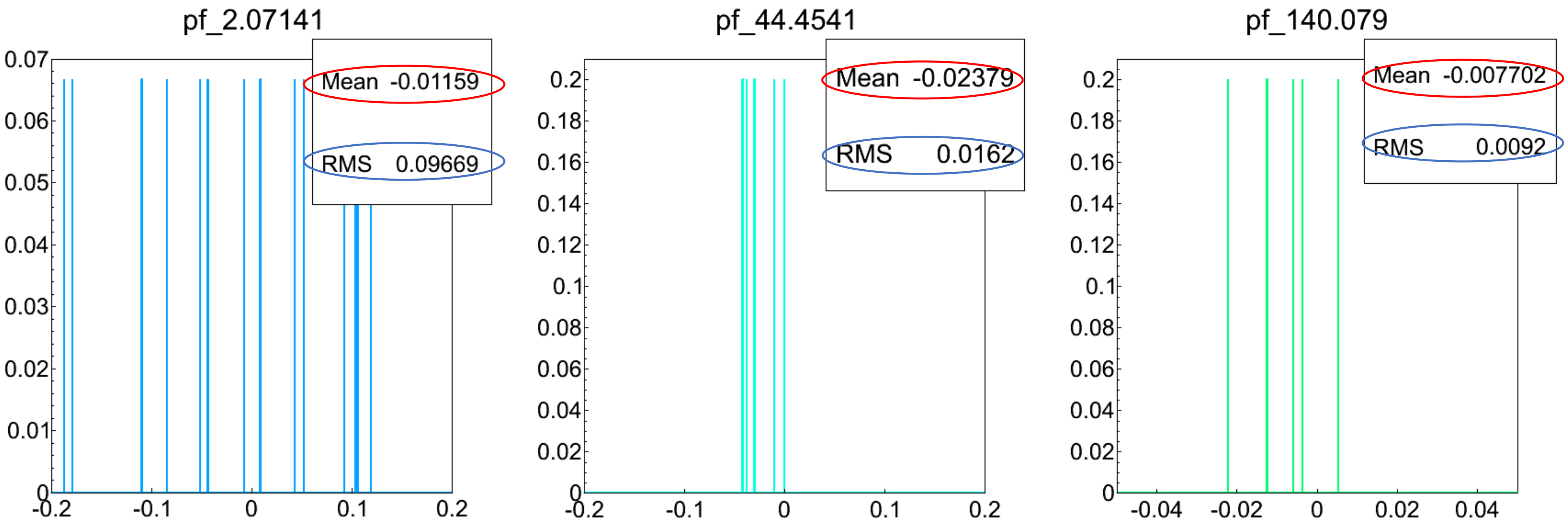
$\varepsilon = (f_{corr} - f_{lin}) / f_{lin}$

ADU/fr	$<a_{corr}>$	RMS[a_{corr}]	$<\varepsilon>$	RMS[ε]
LT1	122.174	4.668	0.0030	0.0148
LT2	122.173	4.612	0.0030	0.0148
LT3	121.320	7.216	-0.0038	0.0505

Bias ε(flux) and its precision – one pixel

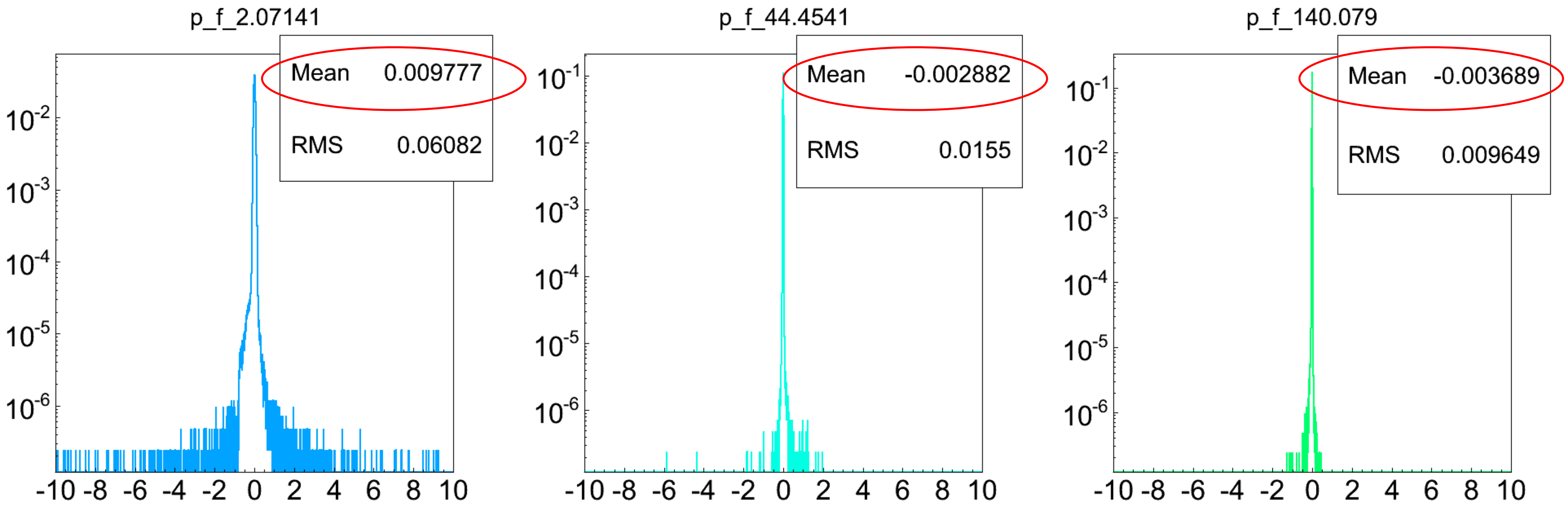
Average ε per pixel over n=5 to 15 values. (same nominal illuminations)

RMS/sqrt(n) = precision on the mean bias



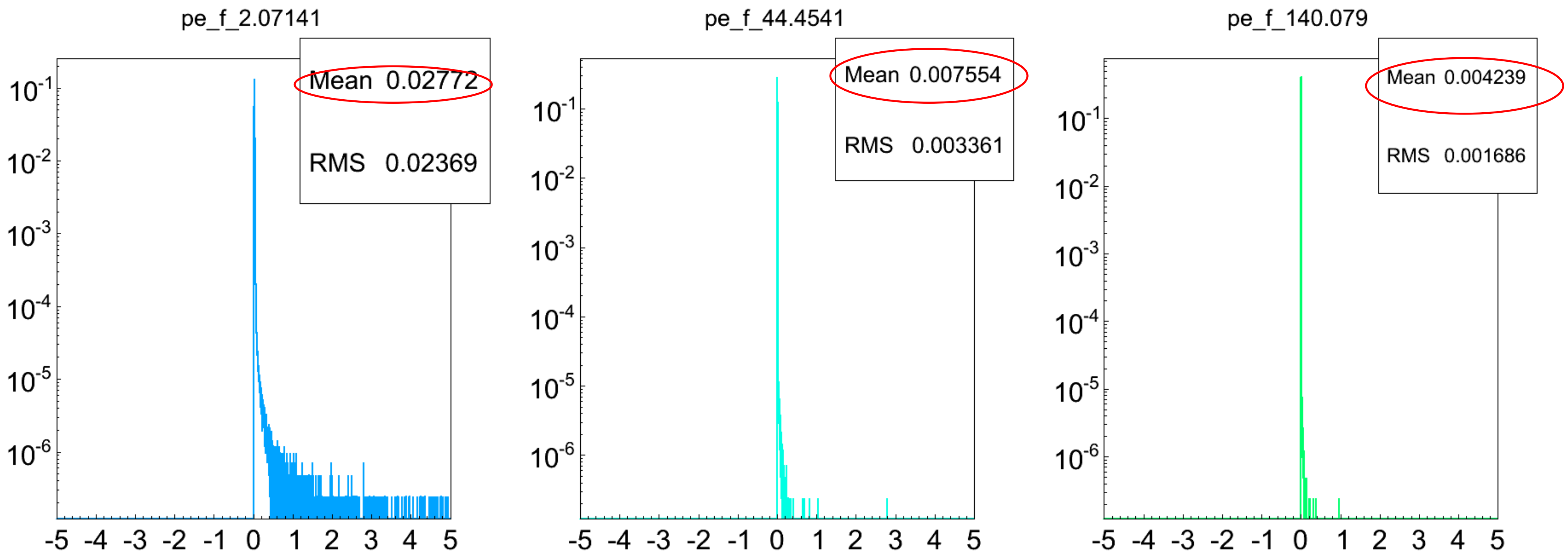
Bias < 1% at high flux, statistically limited at low flux (Poisson noise).

Distribution of ε(flux) over the array LT1



Mean bias compatible with zero for all fluxes.
At high flux the bias is lower than 1% with precision better than 0.5% (see next slide).

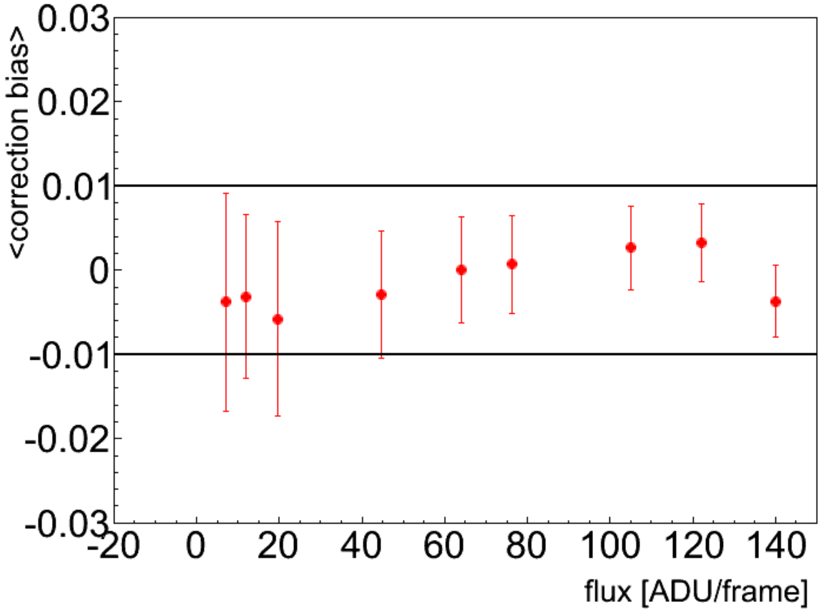
Distribution of precision on ε(flux) LT1



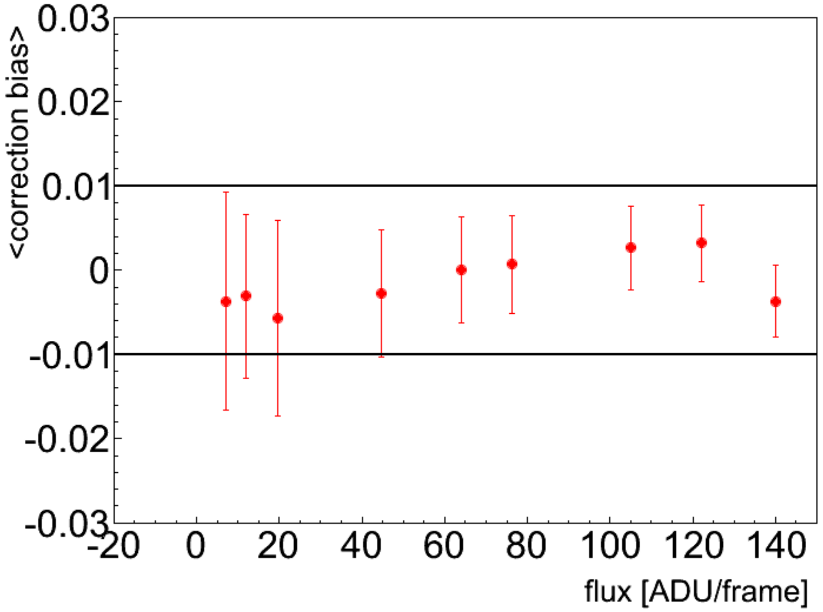
At low flux the accuracy on the mean bias per pixel worse than 1%. (Poisson noise).
At high flux the precision is better than 0.5%.

Spatial mean bias <ε> and it's precision

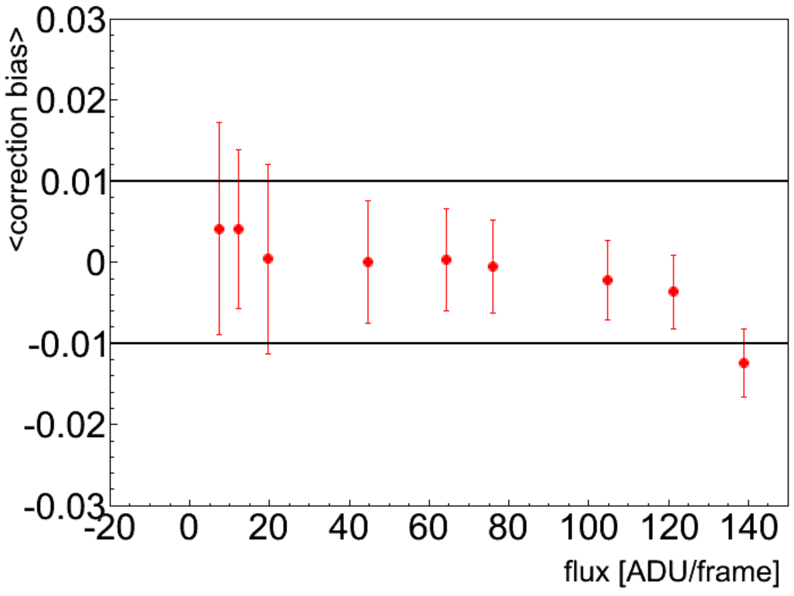
LT1 (7 med fluxes)



LT2 (10 all range fluxes)



LT3 (5 low fluxes)



- 5 low fluxes (LT3) are not enough to fit with accuracy the lookup table. Higher fluxes have to be corrected (somehow).

CONCLUSIONS

Conclusions

- Reference pixels correction:
  - Full study done.
  - Optimal correction found and implemented.
- Signal fit and anomaly detection:
  - Signal fit algorithm established (for spectrometric readout) and implemented
  - Need precise inputs of detector properties σ_R and f_e .
  - Anomaly detection feasible, need to adjust thresholds.
- Nonlinearity correction:
  - Algorithm established and implemented.
  - Nonlinearity correction bias and precision depend on input data $\Rightarrow$ optimization of the needed input under study.