Digital signal processing, fast simulation and calibration

RootInteractive tutorial, 10/03/2022

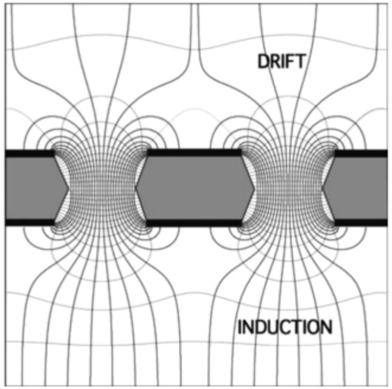
Marian Ivanov, Yiota Chatzidaki

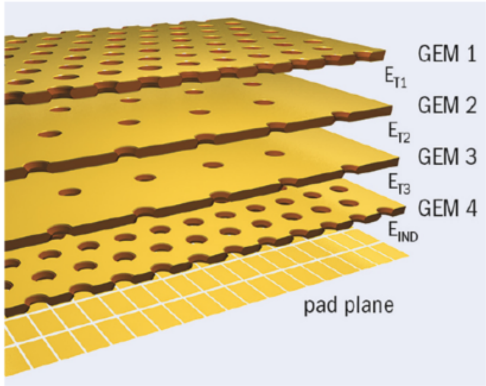
https://alice.its.cern.ch/jira/browse/ATO-559
https://alice.its.cern.ch/jira/browse/ATO-568

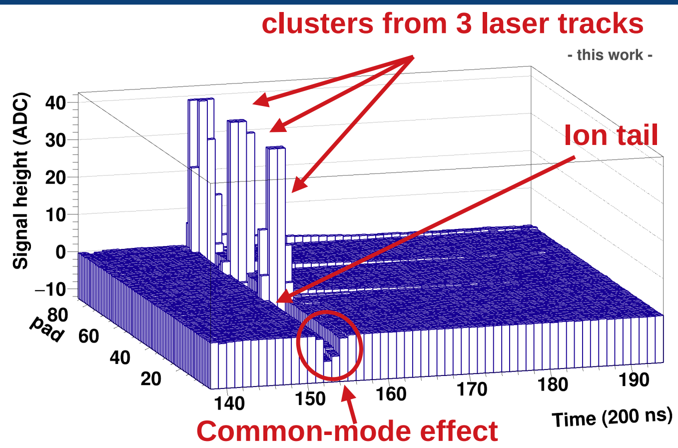
https://indico.cern.ch/event/1135398/contributions/4764024/subcontributions/370740/attachments/2402507/4114266/CMITSimulationsGEMTPC.html

https://indico.cern.ch/event/1135398/contributions/4764024/subcontributions/370740/attachments/2402507/4109039/CMITSimulationsGEMTPC.mp4

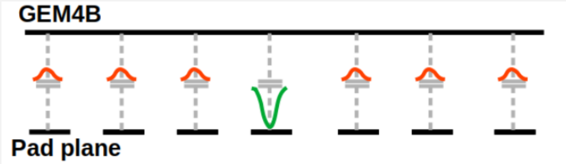
Simulation of the Common-mode (CM) and Ion-tail (IT) in the GEM-based ALICE TPC



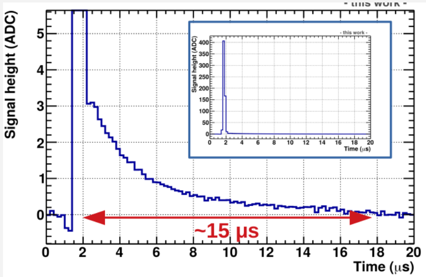




- **Common-mode effect:**
  - Source: capacitive coupling between GEM and pad-plane
  - Impact: negative undershoot when positive signal is detected
  - Characteristics: simultaneous effect, pad-dependent magnitude



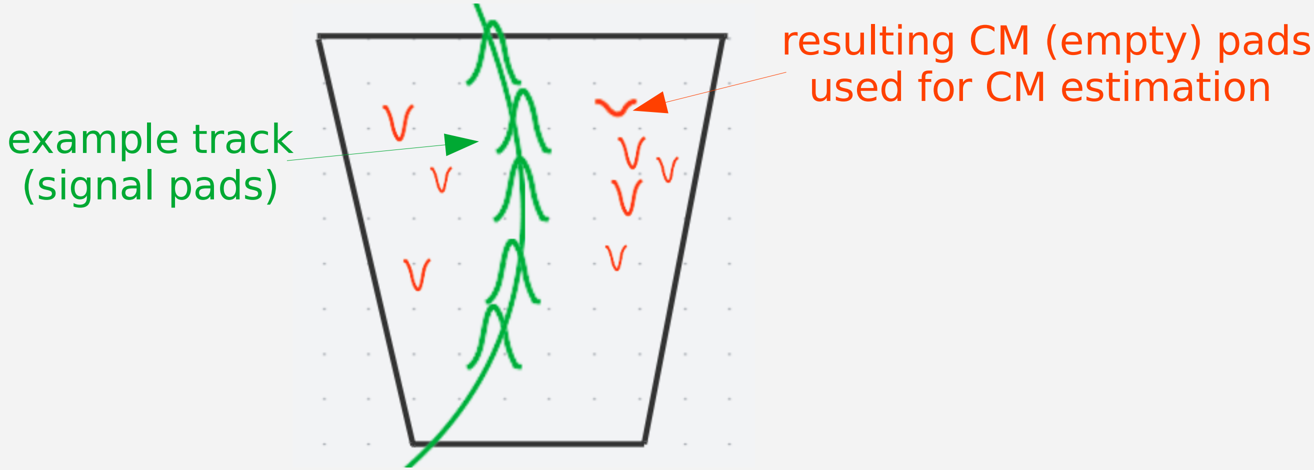
- **Ion-tail:**
  - Source: ions created in GEM4 holes or induction gap
  - Impact: positive tail only in signal pads
  - Characteristics: tail height and slope are pad-dependent



- CM+IT deteriorate the TPC performance → **online correction needed**
- Goal of studies: **quantify impact of CM&IT, develop/optimize corrections**

CM correction methods (schematic description)

- We want to calculate the per-pad baseline shift because of the CM on a timebin basis
- Each pad will have a different response due to the pulser variations



“Neighbors” method:

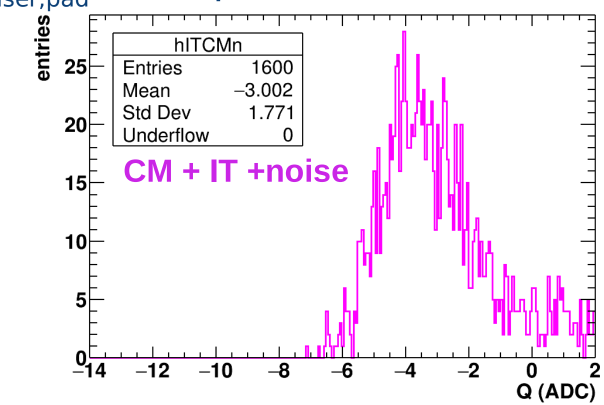
- Calculate the CM shift by identifying the “empty pads” for each timebin
- The “empty pads” are identified using a 2-step algorithm

mean baseline shift

$$Q^{\mathrm{CM}}_{\mathrm{pad}}(t) = Q^{\mathrm{norm}}_{\mathrm{pulser,pad}} \left\langle \frac{Q_{\mathrm{pad}}(t)}{Q^{\mathrm{norm}}_{\mathrm{pulser,pad}}} \right\rangle_{\text{empty pads in CRU}}$$

“Histogram” method:

- Histogram the $Q_{pad}/Q_{pulser,pad}$ of all pads for one timebin
- Max = average CM
- “LTM” method:
  - improved results
  - param. **fractionLTM**

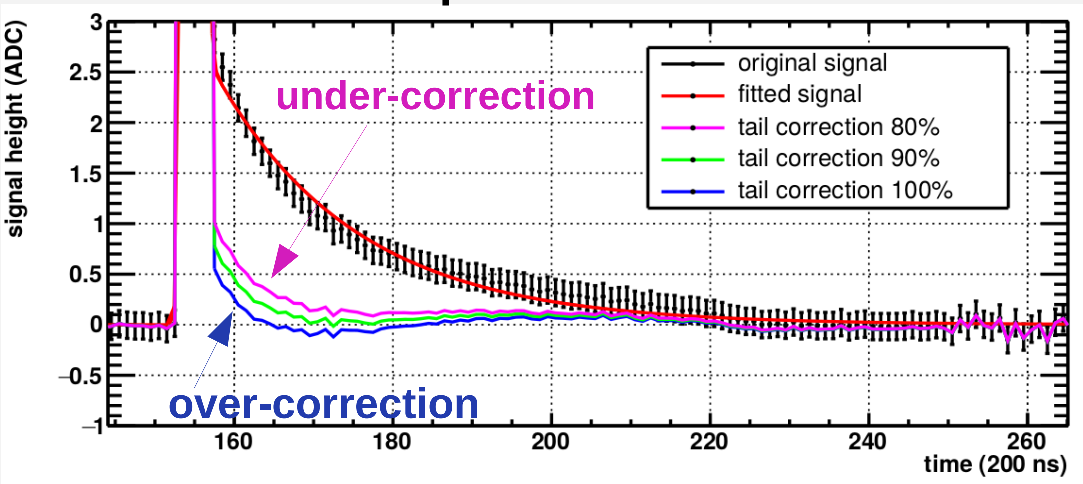


IT correction methods (brief description)

Exponential filter IT method

- Filter applied on a channel (=pad) basis
- Three parameters need to be provided:
  - k_0 : same for all pads – value to be decided based on simulations
  - IT slope and IT fraction:
    - either pad-by pad (if Kr calibration successful) or
    - average correction
  - Performance studies with toy MC

Example correction:



Simulation setup

Data-driven toy MC simulation for CM&IT effects and their correction in TPC

Goal: quantify impact of CM&IT, optimize corrections (mean bias, RMS)

- 1600 pads with noise, CM fraction, IT parameters sampled from real data
- O(6000) events (each 4000 timebins). For each event:
  - Clusters are randomly generated in pad/time space
  - Signal max. of each cluster following the LHC15o data
  - Signal spread over 3 timebins & 3 pads following a Gaussian distribution
- IT, CM are generated
- Noise, pedestal, rounding is added
- Pedestal correction
- CM correction. Scan correction settings/methods
  - Neighbors
  - Simple histogram
  - LTM histogram
- 2^{nd} order correction
  - Mean CM method
  - Mean IT method
- IT correction. Scan correction settings/methods
  - pad-by-pad
  - median
- For each event, we vary the different simulation and correction settings

Macroses and data format

https://gitlab.cern.ch/alice-tpc-offline/alice-tpc-notes/-/blob/master/JIRA/ATO-559/cruFilter.C

- Definition of special classes for the pad characteristics (noise, tail parameters, CM parameter)
- Functions for signal, CM, IT generation
- Functions for CM & IT corrections

https://gitlab.cern.ch/alice-tpc-offline/alice-tpc-notes/-/blob/master/JIRA/ATO-559/cruFilterSimul.C

- Setting up the simulation settings (nEvents, nSetups, correction parameter ranges...)
- Running the simulation and storing mean bias and RMS for each setup

general

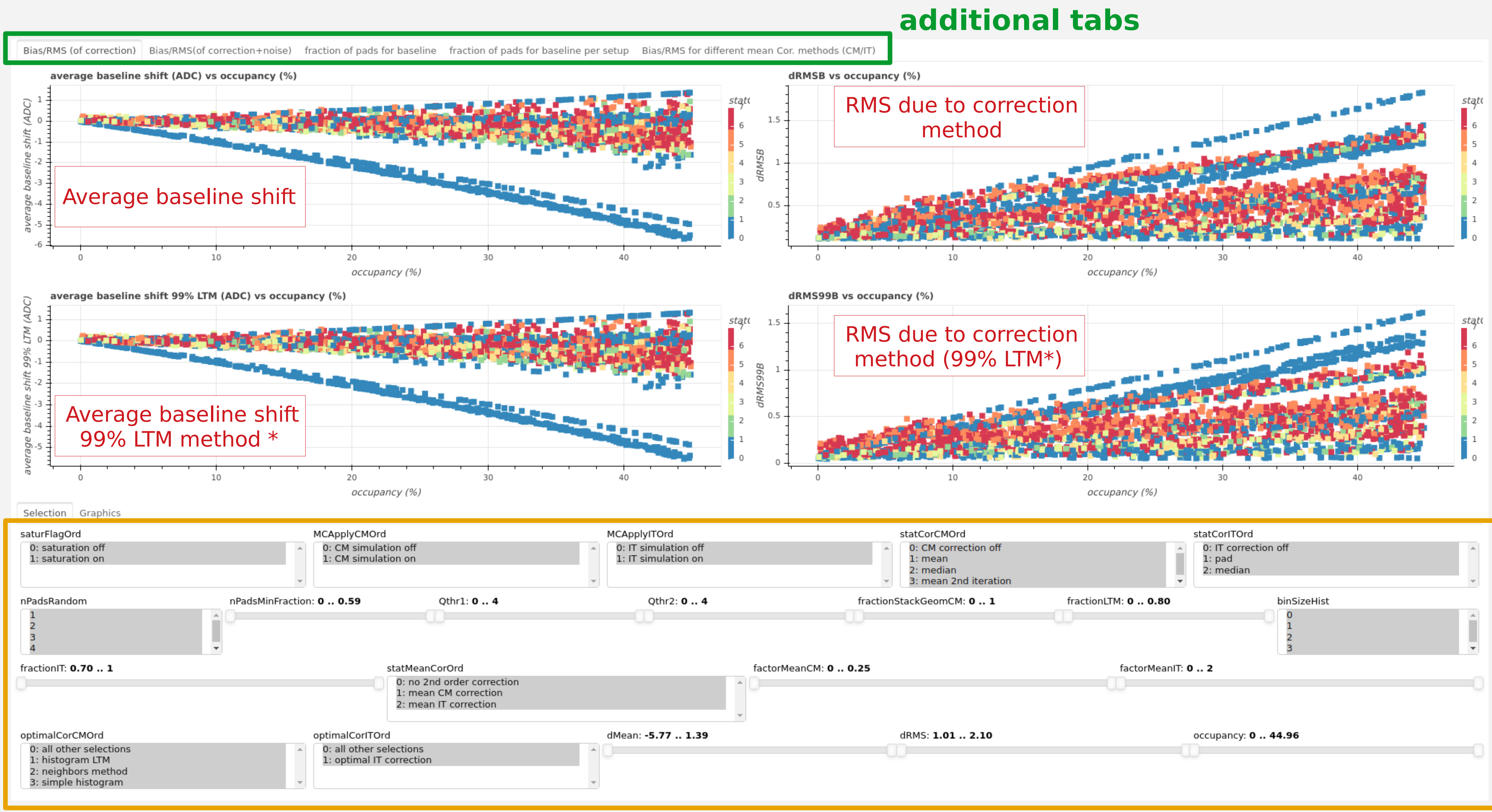
CM settings

IT settings

output

```
// write out information
(*pcstream)             <<  "signalDeltaStat"      <<
"id="                   <<  randomID               <<
"nSignal="              <<  nSignal                <<     // n clusters
"saturFlag="            <<  saturFlag              <<     // saturation flag if saturation is allowed
"saturFraction="        <<  saturFraction          <<     // fraction of saturated pads
"occupancy2="           <<  occupancy2             <<     // fraction of pads/timebins above thresh 2 ADC
//CM
"MCApplyCM="            <<  rStatMCApplyCM         <<     // apply or not CM in MC
"statCorCM="            <<  rStatCorCM             <<     // correct type of CM
// neighbors method
"nPadsRandom="          <<  rNPadsRandom           <<
"nPadsMin="             <<  rNPadsMin              <<
"Qthr1="                <<  rQthr1                 <<
"Qthr2="                <<  rQthr2                 <<
"fractionStackGeomCM="<<  rFractionStackGeomCM   <<     // fraction of pads (geometrically) checked for baseline estimation
"nOK="                  <<  nOK                    <<     // number of pads accepted for baseline estimation
// LTM method
"fractionLTM="          <<  rFractionLTM           <<
"binSizeHist="          <<  rBinSizeHist           <<     // *minBinWidth
// IT
"MCApplyIT="            <<  rStatMCApplyIT         <<     // apply or not IT in MC
"statCorIT="            <<  rStatCorIT             <<     // correct type of IT
"fractionIT="           <<  rFractionTail          <<     // fraction of correction
"statMeanCor="          <<  rStatMeanCor           <<     // statistic for applying mean correction
"factorMeanCM="         <<  rFactorMeanCM          <<     // multiplicative factor for mean corr. method 1
"factorMeanIT="         <<  rFactorMeanIT          <<     // multiplicative factor for mean corr. method 2
"dMean="                <<  dMean                  <<
"dRMS="                 <<  dRMS                   <<
"ordStat.="             <<  &ordStat               <<     // truncstat
"\n";
```

Dashboard layout



Widgets

general widgets

- **saturFlagOrd:** saturation effects on/off
- **MCApplyCMOrd:** CM simulation on/off
- **MCApplyITOrd:** IT simulation on/off
- **statCorCMOrd:** CM effect correction method
  - 0 = CM correction off
  - 1-4 = neighbors methods (mean, median, mean 2nd iter., median 2nd iter.)
  - 5-7 = histogram methods (simple hist., COG hist., LTM hist.)
- **statCorITOrd:** IT correction method
  - 0 = IT correction off
  - 1 = pad correction
  - 2 = median correction

CM widgets

- **nPadsRandom:** number of pads checked (only for neighbors method)
- **nPadsMinFraction:** nPadsMin/nPadsRandom (fraction of required pads, only for neighbors method)
- **Qthr1, Qthr2:** threshold values
- **fractionStackGeomCM:** (geometrical) fraction of pads checked (for neighbors/histogram method)
- **fractionLTM:** fraction for LTM histogram
- **binSizeHist:** bin size for histogram methods (binSizeHist*0.0625ADC)

Widgets

- **fractionIT:** (k_0) fraction for IT correction
- **statMeanCorOrd:** 2^{nd} order correction method
  - 0 = no 2^{nd} order method
  - 1 = mean CM method
  - 2 = mean IT method
- **factorMeanCM:** scaling factor for mean CM method
- **factorMeanIT:** scaling factor for mean IT method

IT/ 2^{nd} order corr. widgets

- **OptimalCorCMOrd:** optimal CM correction methods (specifying param. Ranges)
  - 0 = All other selections
  - 1 = optimal histogram LTM
  - 2 = optimal neighbors methods
  - 3 = optimal simple histogram method
- **OptimalCorITOrd:** optimal IT correction method (specifying param. Ranges)
- 0 = all other selections
- 1 = optimal IT correction
- **Dmean:** mean baseline shift
- **DRMS:** RMS
- **Occupancy:** range of occupancy

remaining widgets

Observations (demonstrated in dashboard)

- **CM on/IT off scenario:**
  - Impact of CM (@40% occupancy): mean~–5 ADC, RMS increase ~1 ADC (18:30-19:30)
  - Bias if 2nd order correction is not applied, @40% occupancy: mean~–0.5 ADC (22:15-23:40)
  - Mean, median, mean 2nd iter.: require better tuning than median 2nd iter., but can achieve comparable results (24:25-25:50)
  - Histogram “max” method: larger RMS compared to histogram “LTM” (0.2 ADC vs 0.4 ADC increase @ 40% occupancy) → bigger difference in case of CM on/IT on scenario (25:50-28:00)
  - Bin size not important in hist methods (between 0.0625 – 0.3 ADC) (28:00-28:20)
  - “Neighbors” and “LTM” methods: similar performance when parameters tuned (28:20-29:40)
- **CM off/IT on scenario:**
  - Impact of IT (@40% occupancy): mean~+1.2 ADC, RMS increase~1.2 ADC (29:40-31:40)
  - “Median” method: corrects mean bias but leads to an increased RMS~0.6 ADC increase
  - “Pad” method: a value f~85% fully restores mean bias and RMS (32:00-33:20)
- **CM on/IT on scenario:**
  - Significant bias if 2nd order correction is not applied, @40% occupancy: mean~-1.5 ADC
  - Complete restoration of baseline bias without increase in RMS with 2nd order correction
  - Slightly improved performance of 2nd order mean IT correction, which can also be achieved with mean CM by adjusting the factorMeanCM (33:40-35:00)

How the jupyter notebook works

Get jupyter notebook from my directory in lustre (or directly from gitlab):

`/lustre/alice/users/pchatzid/NOTES/alice-tpc-notes/JIRA/ATO-559/parameterScan.ipynb`

- Import modules
- Read input file, set aliases
- Create dataframe
- Create additional variables in the dataframe
- Set metadata
- Create dashboard

Backup

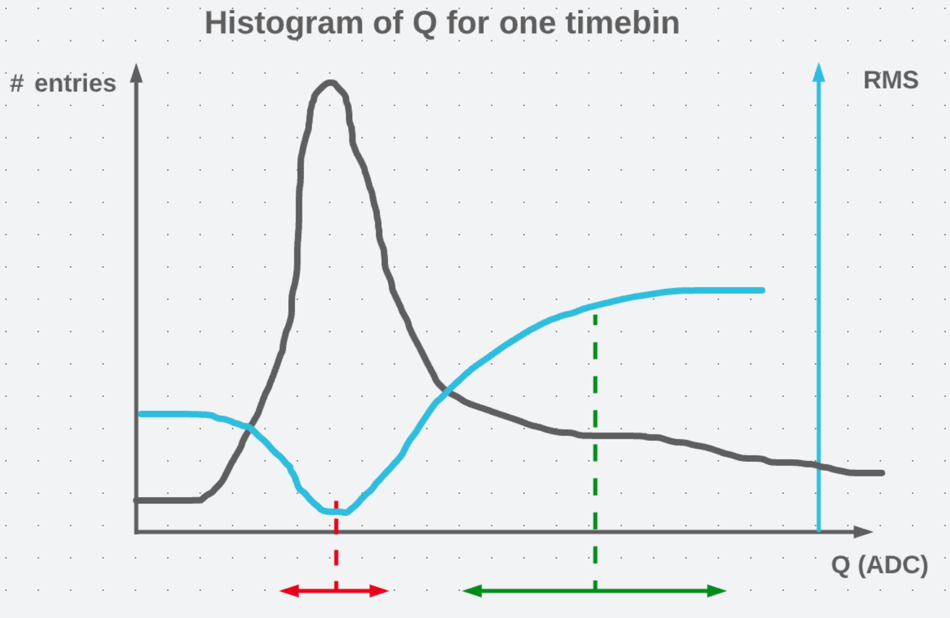
CM correction methods

- **"Neighbors" CM method**
  - Calculate the CM shift by identifying the "empty pads" for each timebin
  - To identify a pad as "empty", its charge Q_{pad} must
    - (a) be below a threshold value ($Q_{pad} < \mathbf{Q_{thr1}}$) and
    - (b) be comparable to that of "neighboring" pads
      ($|Q_{pad}/k_{pad} - Q_{rand}/k_{rand}| < \mathbf{Q_{thr2}}$ for at least $\mathbf{N_{min}}$ of the $\mathbf{N_{random}}$ random pads checked)
  - Estimators mean or median of the selected empty pads can be used to get $Q_{baseline}$
  - A 2[nd] iteration (modifying (a) to $Q_{pad} - Q_{baseline} < Q_{thr1}$) can improve the results

- **"Histogram" CM method**
  - Histogram the Q_{pad}/k_{pad} of all pads for one timebin, using appropriate **binSize**
  - The hist. maximum should correspond to the average baseline shift
  - Improved results with the "histogram LTM" method, using appropriate **fractionLTM**

Histogram LTM method - logic

- Moving from left to right, the RMS is calculated at each bin using f% of total histogram entries (f%/2 from left, f%/2 from right)
- Find minimum RMS
- Calculate mean of f% of entries around minimum RMS



Implementation in `TStatToolkit` (http://alidoc.cern.ch/AliRoot/master/_t_stat_toolkit_8h_source.html)

Improvement in performance is demonstrated later

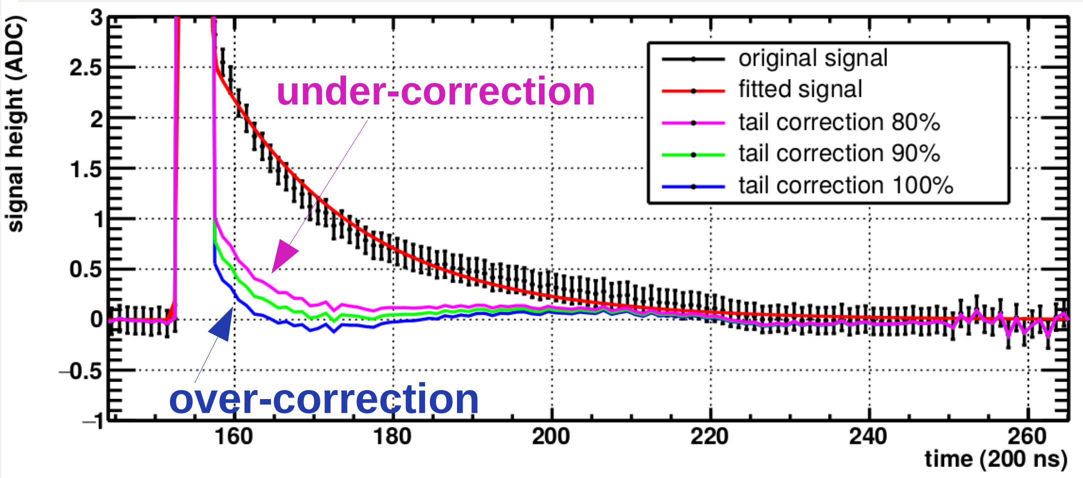
IT correction methods

Simple exponential filter on a channel-by-channel basis

```
// constant to be set before
float k0; // multiplicative correction factor, same for all pads, say 90%
for each padID{
    float k1 = GetFractionIT(padID); //k_fraction, from a static map
    k1*=k0; // scale IT fraction
    float k2 = GetExpSlopeIT(padID); // = exp(-k_slope), from a static map
    float Q_correction = 0;
    for each timebin{
        Q_out = Q_in - k1*(1-k2)*Q_correction;
        Q_correction+=Q_in;
        Q_correction*=k2;
    }
}
```

3 parameters:
- **IT fraction from map**
- **IT slope from map**
- **k0 fixed for all pads****

free parameter to compensate for difference between (slow, precise) de-convolution and (fast, approximate) exponential tail correction



Assuming Kr calibration is successful:
- **use per-pad IT fraction & IT slope ("pad" corr.)**
- **otherwise use median values ("median" corr.)**

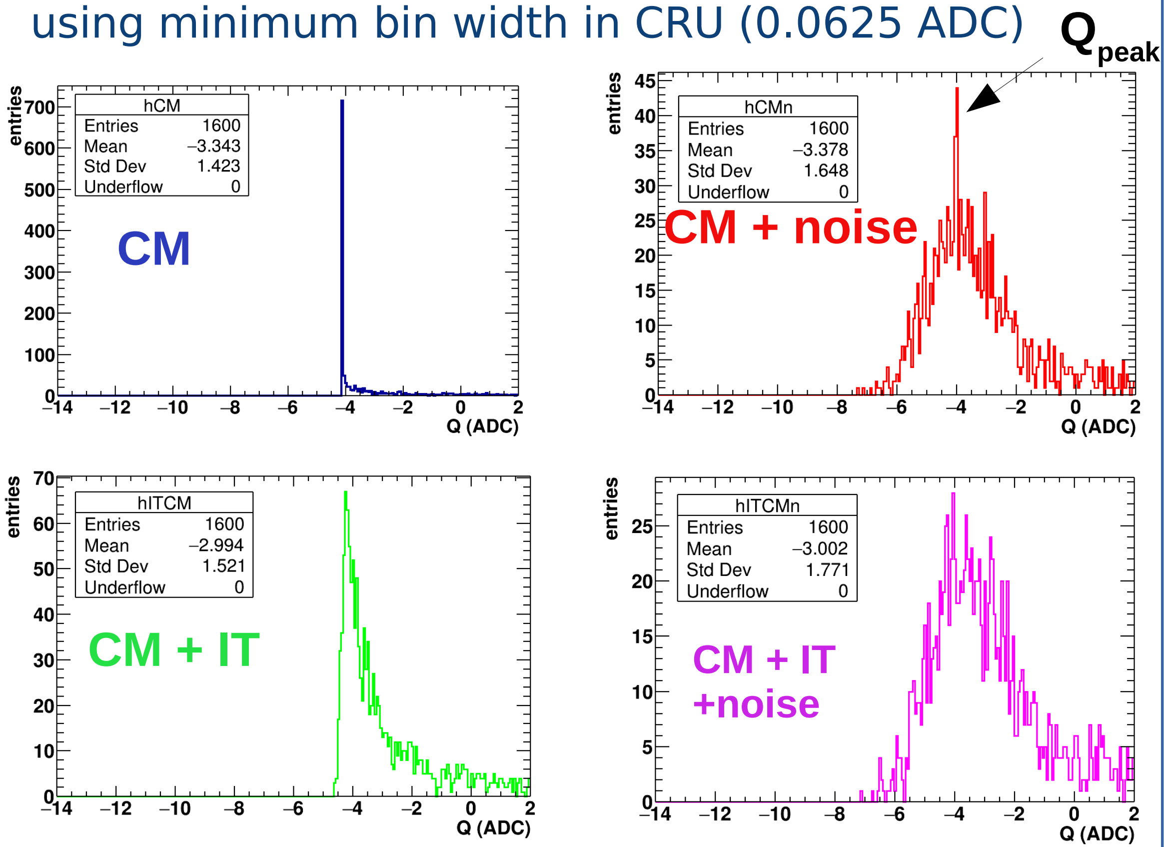
Histogram method for CM correction

Instead of finding the empty pads and calculating the mean baseline shift for each tb

(1) Create a histogram in the FPGA of the $Q_{pad}(t)/k_{pad}$

(2) The empty pads will be found around the peak

Example for 30% occupancy
using minimum bin width in CRU (0.0625 ADC)



Different options for baseline estimation:

- Use Q_{peak} as baseline
- Use weighted (in 3–5 bins) mean
- Use trimmed mean in ±1-1.5σ_{noise} region
- Keep padIDs contributing to the peak and calculate mean Q

Considerations:

- Too small bin width → peak position not well-defined
- Too large bin width → also ''signal'' pads included in the peak

Neighbors method for CM correction

```
// Some constants to be set before
int nPadsCRU; // #pads in the current CRU
int nPadsRandom; // #pads, say 10, at random distance to the current to pad. To be
    compared for additional check if the current pad is an empty pad
int nPadsMin; // Minimum #pads, say nPadsRandom/2, required to have Q very close to
    current pad.
float Q_thr1; // Optimized threshold 1 with toy MC; comparable to 2*noise
float Q_thr2; // Optimized threshold 2 with toy MC; comparable to noise
//
// Main CM correction algorithm
for each timebin{
    // calculate mean baseline of empty pads
    vector<float> Q_pad_array; // array to hold Q of empty pads of a CRU
    for each pad{
        float k_pulser = GetNormQPulser(padID); // from 2D pad-by-pad map
        // simple check if the current pad is empty
        if (Q_pad <= Q_thr1){
            // additional check if the current pad is empty pad
            int nPadsOK = 0;
            for (int i = 0; i < nPadsRandom; i++){
                int padRandomID = GetRandomPad(); // randomly select a pad
                float k_pulser_rndm = GetNormQPulser(padRandomID);
                if (abs(Q_pad/k_pulser - Q_padRndm/k_pulser_rndm) < Q_thr2)
                nPadsOK++;
            }
            // if empty, add charge of current pad into array, scale accordingly
            if (nPadsOK >= nPadsMin) Q_pad_array.push_back(Q_pad/k_pulser);
        }
    }
    // calculate mean baseline of the non-signal pads
    Q_baseline = mean(Q_pad_array); // or median
    // apply Common-mode correction
    for each pad{
        float k_pulser = GetNormQPulser(padID); // from 2D pad-by-pad map
        Q_pad = Q_pad - Q_baseline*k_pulser;
    }
}
```

Main equation (per timebin):

$$Q^{CM}_{pad}(t) = Q^{norm}_{pulser,pad}\left\langle \frac{Q_{pad}(t)}{Q^{norm}_{pulser,pad}} \right\rangle_{\text{empty pads in CRU}}$$

mean baseline shift

empty pads in CRU

Empty pad definition:

(1) Check if $\mathbf{Q_{pad}(t) < Q_{thr1}}$ (~2 ADC)

(2) Compare $Q_{pad}(t)/k_{pad}$ to that of N_{random} pads:

- If for at least N_{min} pads:

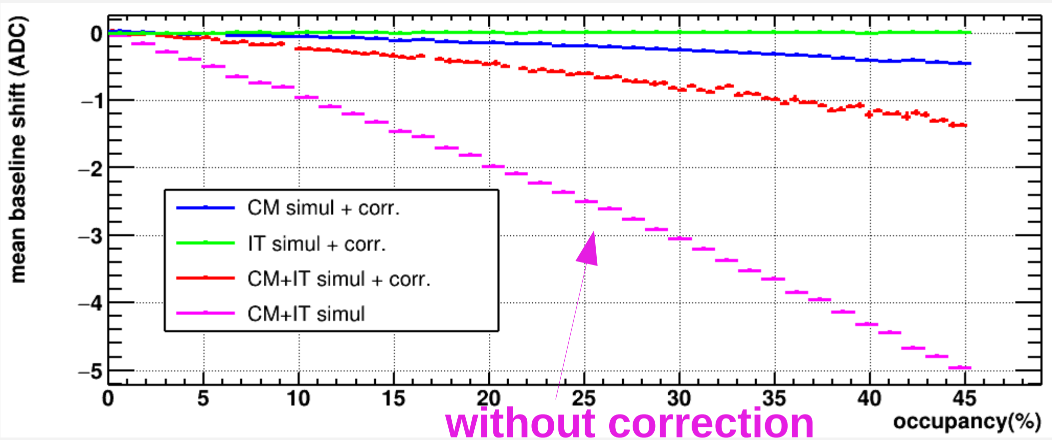
$$|Q_{pad}(t)/k_{pad} - Q_{rand}(t)/k_{rand}| < Q_{thr2} \text{ (~2 ADC)}$$

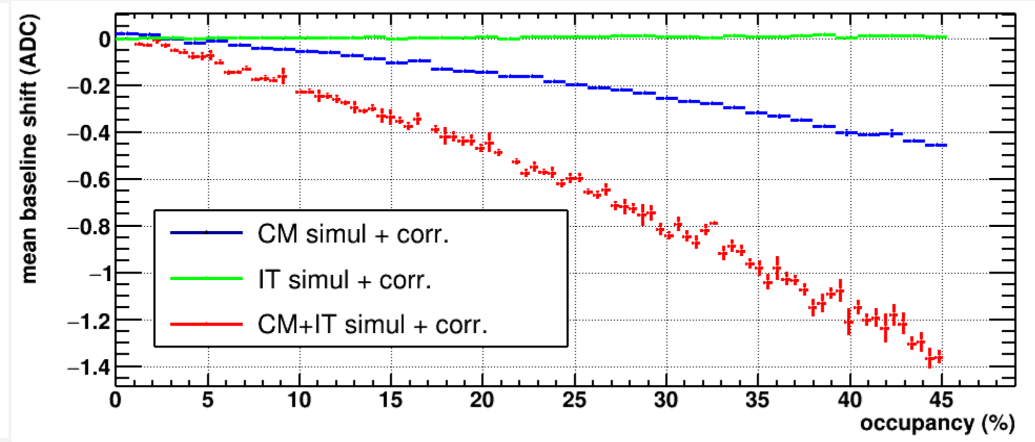
Then the pad is considered empty

(3) Apply a 2nd iteration modifying (1) as $Q_{pad}(t) - Q_{baseline} < Q_{thr1}$ (optional)

$N_{min}/N_{random}, Q_{thr1}, Q_{thr2}$ require optimization

Remaining biases (two different origins)





Remaining biases in case of:

- **CM simulation & correction (–0.4 ADC @40%occupancy)**
  - Caused due to “signal contamination” (low-signals biasing the CM estimation)
- **CM+IT simulation & correction (–1.2 ADC @40%occupancy)**
  - Caused due to “signal contamination” (as before)
  - Caused due to “double-correction of IT”

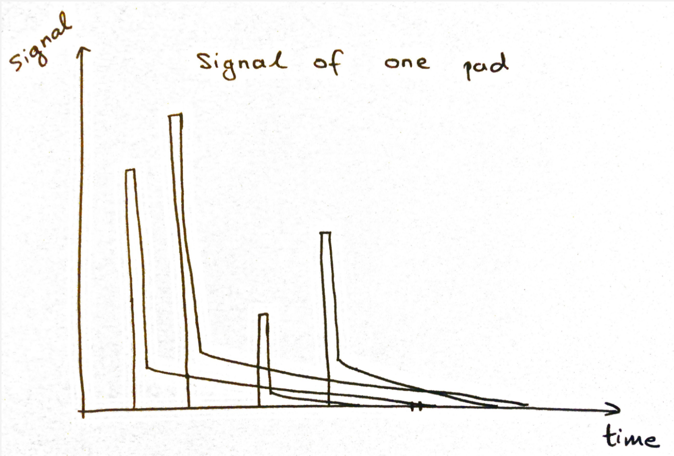
The CM correction (applied before the IT correction), corrects for the **mean baseline** which consists of:

- the CM
- the **“mean IT”** which shifts the baseline towards less negative values

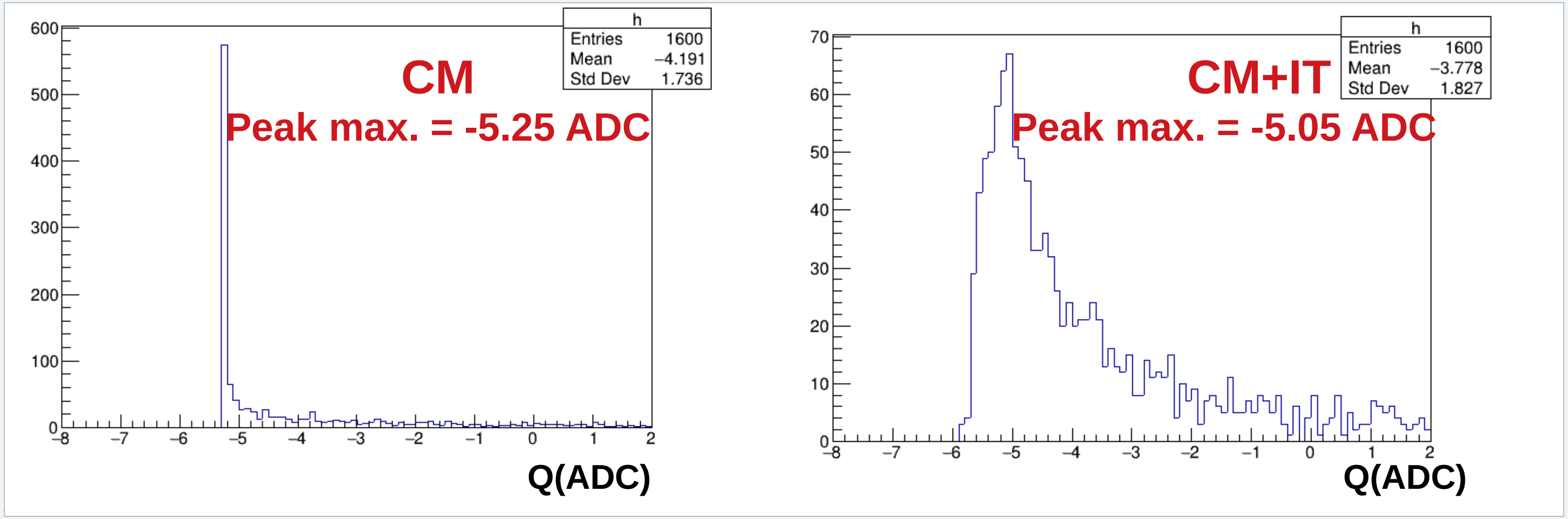
Understanding the double correction of the IT

- For a given timebin, the pad "sees" the IT of signals from previous timebins

  → The baseline is shifted towards higher values



- Plotting the Q of all pads for one timebin, the Q distribution is shifted accordingly



2^{nd} order correction methods for remaining biases

- **"Mean CM correction":** simple rescaling of CM to mitigate residual bias due to signal contamination

```
for each time{
    Fill histQ with Q_pad/k_pad values
    float meanBase = histQ→GetMaxPos();
    for each pad{
      Q_pad += |meanBase|*k_pad;
      Q_pad += factorMeanCM*|meanBase|;
    }
}
```

Standard CM correction (as an example, histogram method)

factorMeanCM: same for all pads

- **"Mean IT correction":** exponential filter to mitigate double correction of IT

```
kAmp  = factorMeanIT*meanFractionIT; kTime = meanSlopeIT; tailSlopeUnit = exp(-kTime);
cumul = 0;
for each time{
    Fill histQ with Q_pad/k_pad values
    float meanBase = histQ→GetMaxPos();
    for each pad{
      Q_pad += |meanBase|*k_pad;
      Q_pad += kAmp*(1-tailSlopeUnit)*cumul;
    }
     cumul+=|meanBase|; cumul*=tailSlopeUnit;
}
```

factorMeanIT: same for all pads

Standard CM correction (as an example, histogram method)

meanFractionIT, meanSlopeIT: mean of fractionIT and slopeIT distributions

Either can be used as a 2^{nd} order correction. Effect demonstrated in dashboard

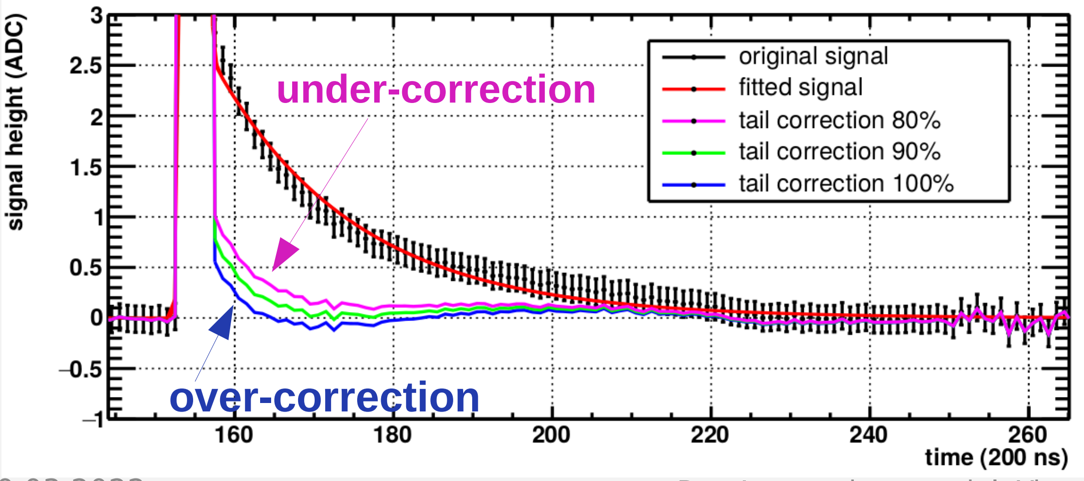
Ion-tail correction - reminder

Simple exponential filter as an approximation of unfolding

```
// constant to be set before
float k0; // multiplicative correction factor, same for all pads, say 90%
for each padID{
     float k1 = GetFractionIT(padID); //k_fraction, from a static map
     k1*=k0; // scale IT fraction
     float k2 = GetExpSlopeIT(padID); // = exp(-k_slope), from a static map
     float Q_correction = 0;
     for each timebin{
          Q_out = Q_in - k1*(1-k2)*Q_correction;
          Q_correction+=Q_in;
          Q_correction*=k2;
     }
}
```

3 parameters:

- **IT fraction from map**
- **IT slope from map**
- **k0 fixed for all pads****



free parameter to compensate for difference between (slow, precise) de-convolution and (fast, approximate) exponential tail correction

Observations (demonstrated in dashboard)

- **CM on/IT off scenario:**
  - Impact of CM (@40% occupancy): mean~–5 ADC, RMS increase ~1 ADC
  - Bias if 2nd order correction is not applied, @40% occupancy: mean~–0.5 ADC
  - Mean, median, mean 2nd iter.: require better tuning than median 2nd iter., but can achieve comparable results
  - Histogram “max” method: larger RMS compared to histogram “LTM” (0.2 ADC vs 0.4 ADC increase @ 40% occupancy) → bigger difference in case of CM on/IT on scenario
  - Bin size not important in hist methods (between 0.0625 – 0.3 ADC)
  - “Neighbors” and “LTM” methods: similar performance when parameters are tuned
- **CM off/IT on scenario:**
  - Impact of IT (@40% occupancy): mean~+1.2 ADC, RMS increase~1.2 ADC
  - “Median” method: corrects mean bias but leads to an increased RMS~0.6 ADC increase
  - “Pad” method: a value f~85% fully restores mean bias and RMS
- **CM on/IT on scenario:**
  - Significant bias if 2nd order correction is not applied, @40% occupancy: mean~-1.5 ADC
  - Complete restoration of baseline bias without increase in RMS with 2nd order correction
  - Slightly improved performance of 2nd order mean IT correction, which can also be achieved with mean CM by adjusting the factorMeanCM

Conclusions

- **Effects well-understood and under control**

- **CM correction method to be decided**
  - Methods “neighbors” and “histogram LTM” work well if parameters are tuned
  - Simple histogram method has increased bias
  - Histogram LTM requires only one tune parameter (fractionLTM)

- **IT correction method is finalized**
  - IT parameters used to be decided based on success of Kr. calibration

- **2nd order correction method to be decided**
  - Both should be straightforward to implement
  - Both can lead to full restoration of baseline, by selecting the tune factor