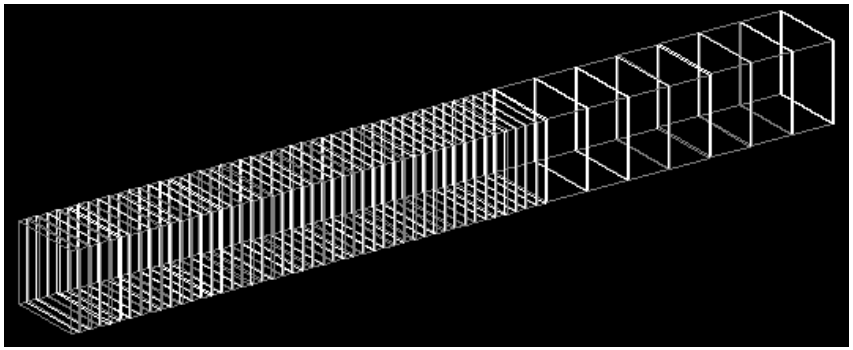
Pion energy resolution using a global software compensation method

A.-M. Magnan
Imperial College London

29/07/2014, HGCAL-Software weekly meeting

Setup

- Standalone simulation "version 23": Calice-like HCAL only, and "version 21": HGCAL-v4.
- Using QGSP_BERT physics list.
- Following method described in JINST 7(2012) P09017.
- Same energy range: pions from 10 to 80 GeV.
- Same digitisation procedure.



- Thanks to Marina Chadeeva (ITEP) and Frank Simon (MPI) for lots of help.

Digitisation procedure

Timing cut of 150 ns

10

MC digitization procedure

Conversion of MC signal from MeV to MIP $\rightarrow e_{MIP}$

$\Downarrow$

Generation of random number of photoelectrons n_{pe}
from Poisson distribution with $\langle n_{pe} \rangle = e_{MIP} \cdot N_{pe}$ $(N_{pe} = 11)$

$\Downarrow$

Calculation of number of pixels n_{pixel} taking in account saturation and cross talk C:
$n_{pixel} = N_{total} \cdot \frac{(1-x)}{(1-C \cdot x)}$ where $x = \exp(-n_{pe}/N_{total})$, $C = 0.25$, $N_{total} = 1156$

$\Downarrow$

Generation of random signal n_{pixel}^{new} from Gaussian distribution $Gaus(\langle n_{pixel} \rangle, \sigma_G)$
where $\sigma_G = 3$ pixels (≈ 0.2 MIP)

$\Downarrow$

Recalculation from pixel to MIP: $e_{MIP}^{new} = \frac{N_{total}}{N_{pe}} \cdot \ln(\frac{N_{total} - C \cdot n_{pixel}^{new}}{N_{total} - n_{pixel}^{new}})$

$\Downarrow$

~~Adding noise hits from random event of pedestal run cell to cell~~

Adding 0.12 MIP gaussian noise everywhere

Adding 2.5% cross-talk / 3-cm edge

Implementing exact granularity

choose 2% dead cells randomly in 1st event

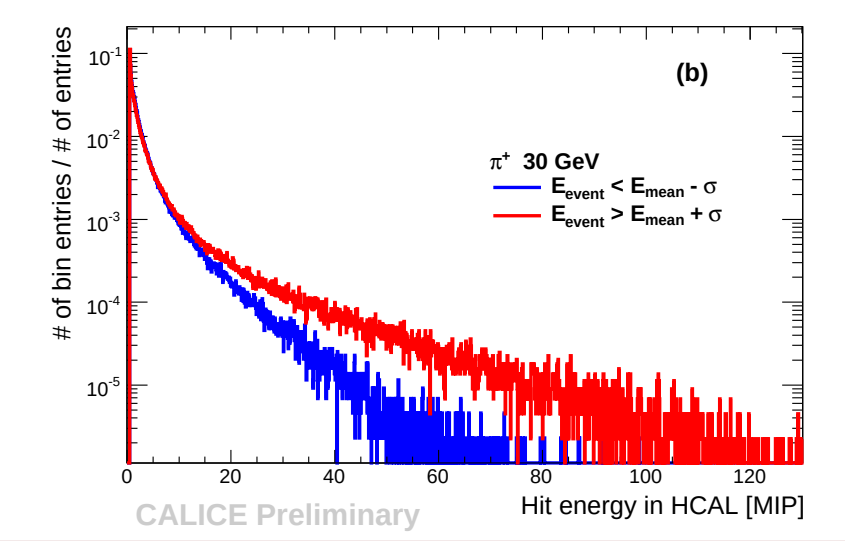
Marina Chadeeva, ITEP

ek at IPN Lyon, September 2009

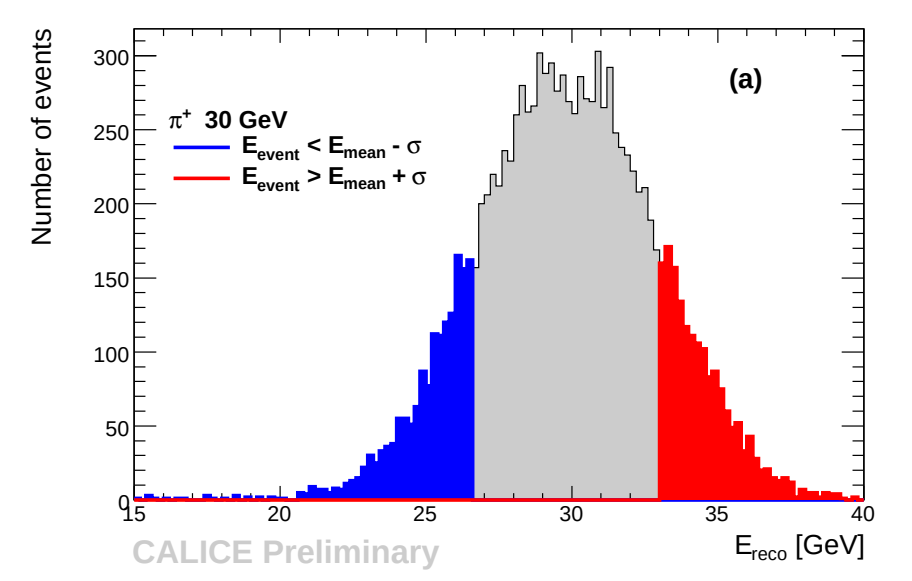
Global software compensation method (I)

- See Calice CAN-028: `https://twiki.cern.ch/twiki/bin/view/CALICE/CaliceAnalysisNotes`
- Main idea: identify event-by-event the electromagnetic content (higher energy density), and correct accordingly.
- Global method: based on different hit spectra for different EM content.

Hit spectra



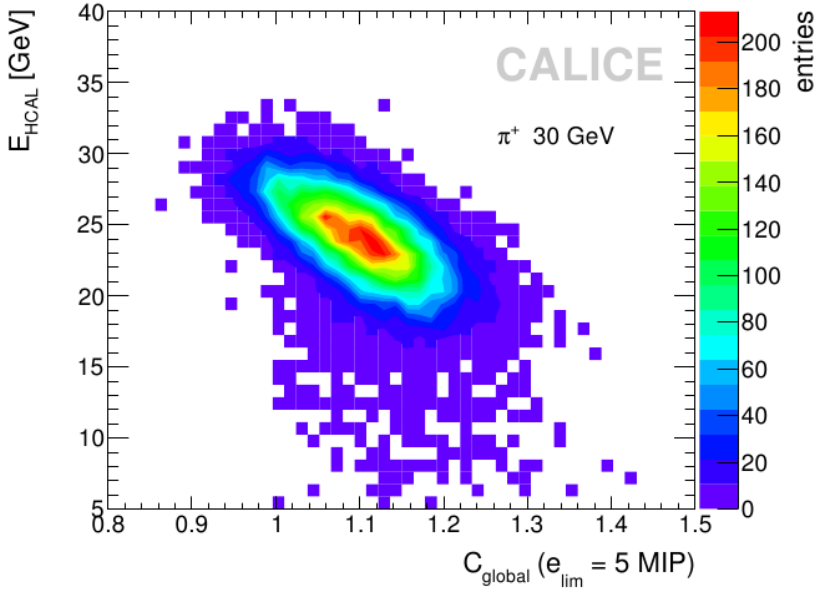
Total energy

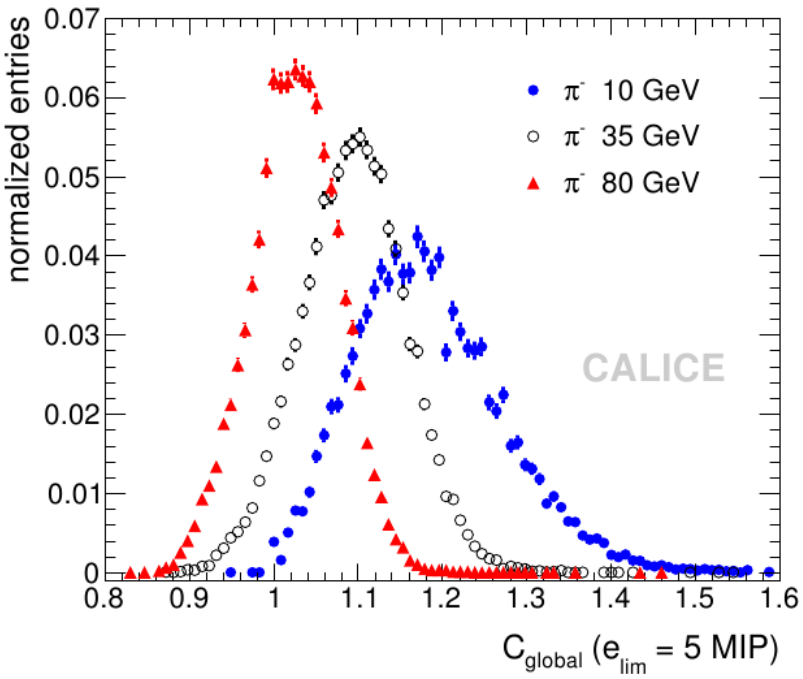


- Define $C_{global} = \frac{N_i(e \leq e_{lim})}{N_i(e \leq e_{av})}$
- N_i: number of hits in a given event
- $e_{lim} = 5MIPs$ free parameter tuned to work for all energies.
- e_{av} mean energy of the event hit spectrum in MIPs.

Global software compensation method (II)

- Use anti-correlation between total energy and C_{global} to correct on an event-by-event basis.
- $E_{shower} = E_{uncor} \times C_{global}$
- Ad'hoc correction to recover the linearity using second-order polynom.





- As expected: EM fraction increases with particle energy $\Rightarrow$ C_{global} decreases.

Calice Results

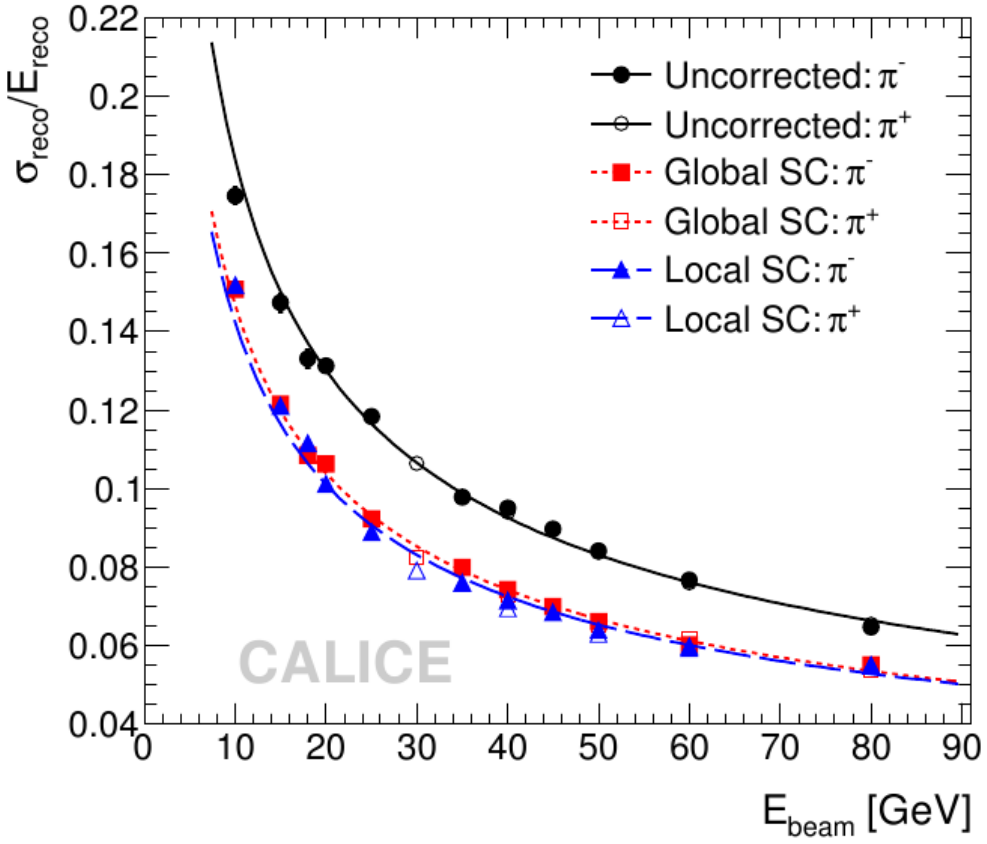
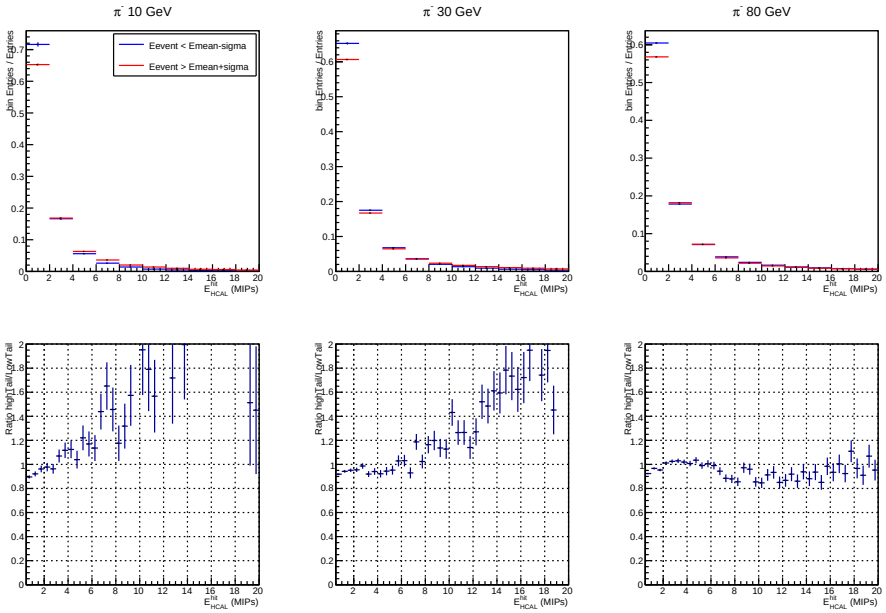


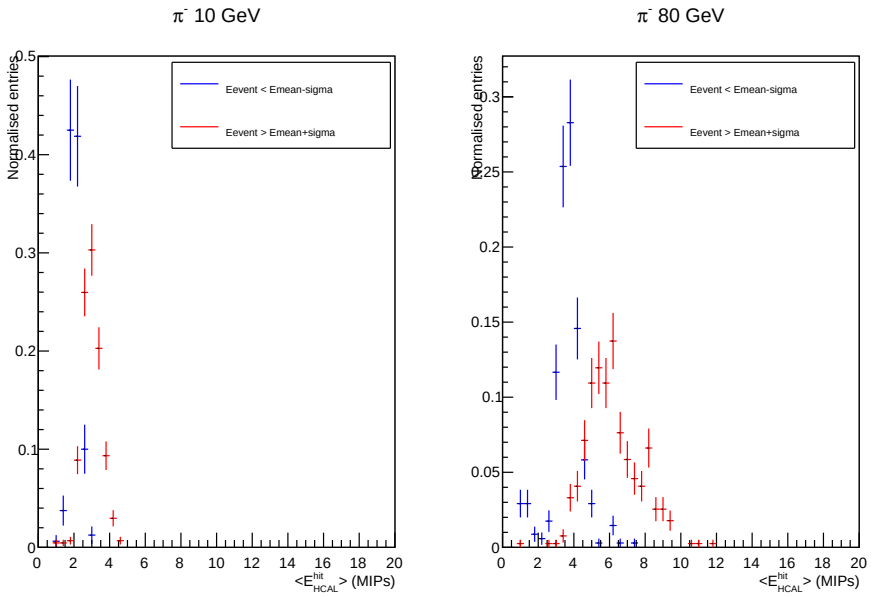
Figure 4. Energy resolution versus beam energy without compensation and after local and global software compensation. The curves show fits using Equation 2.2, with the black solid line showing the fit to the uncorrected resolution, the red dotted line to the global software compensation and the blue dashed line to the local software compensation. The stochastic term is $(57.6 \pm 0.4)\%$, $(45.8 \pm 0.3)\%$ and $(44.3 \pm 0.3)\%$, with constant terms of $(1.6 \pm 0.3)\%$, $(1.6 \pm 0.2)\%$ and $(1.8 \pm 0.3)\%$ for the uncorrected resolution, global software compensation and local software compensation, respectively.

Application: hit spectra

Hit spectra

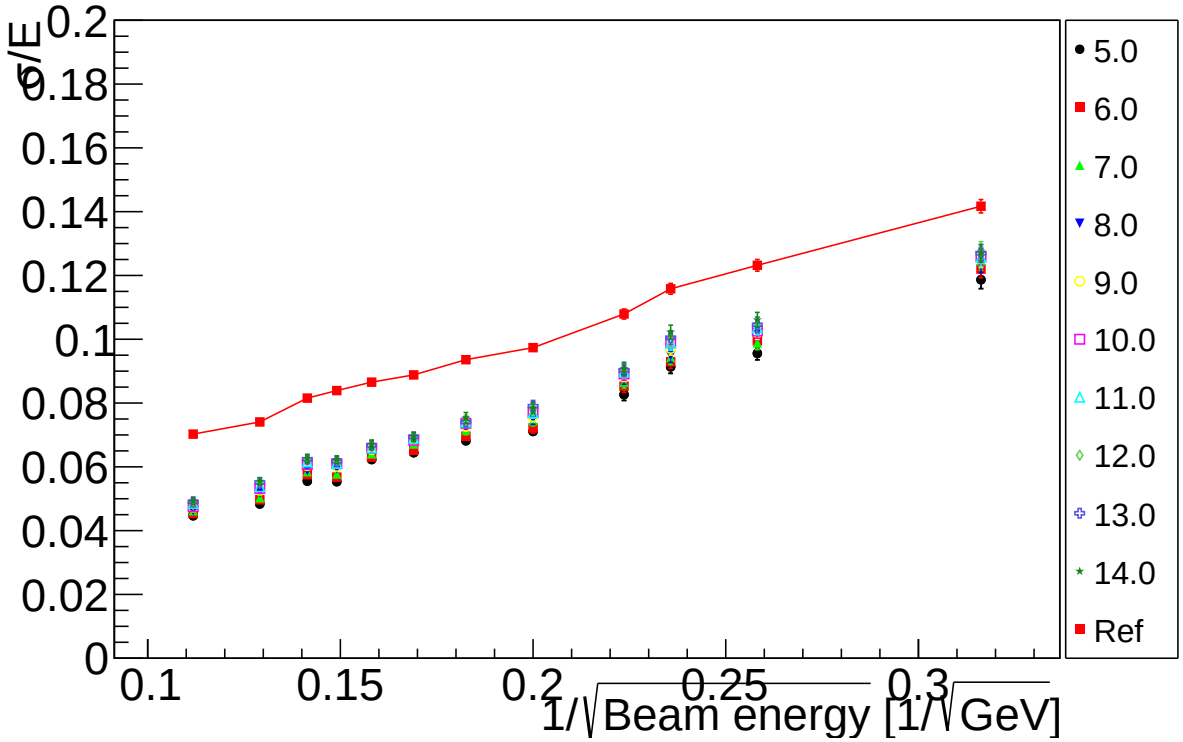


Mean/event of hit spectra



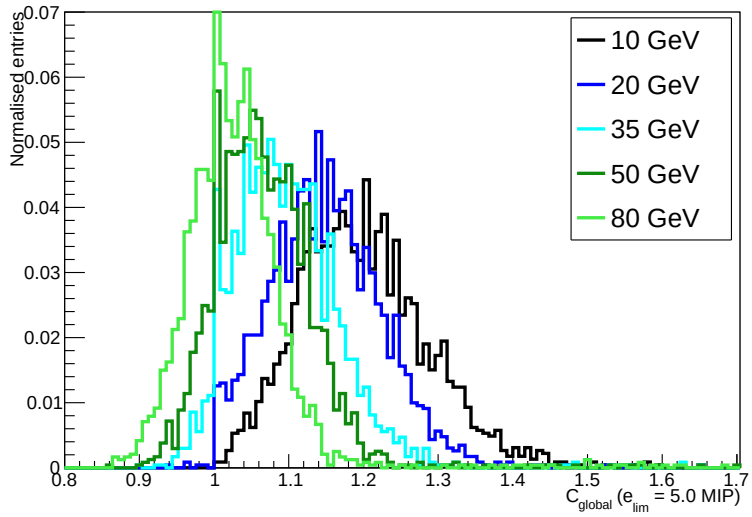
- Mean of hit spectra compatible with Calice case.
- Little stat available....

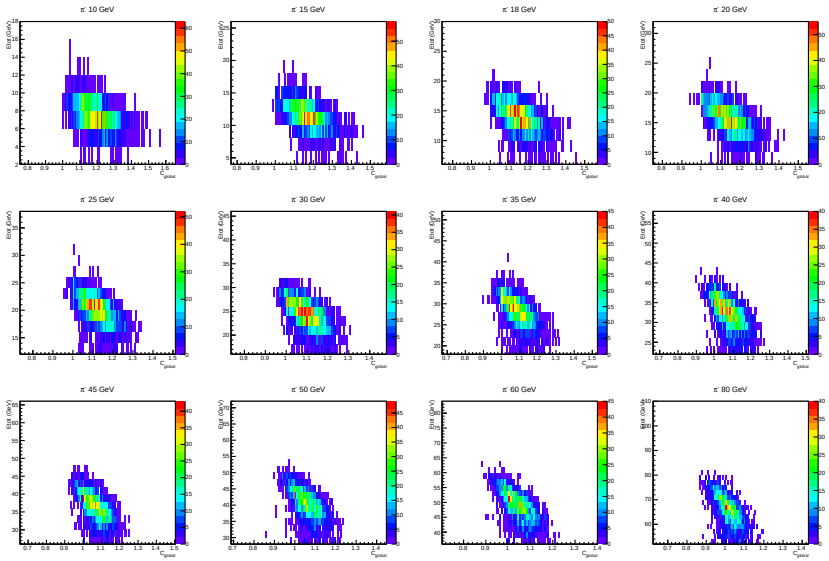
Tuning of e_{lim}

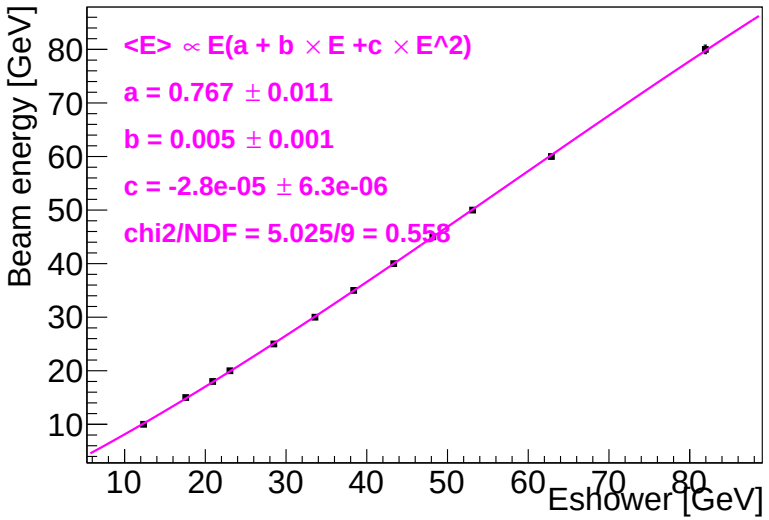


- $e_{lim} = 5$ MIPs looks good across all energy range.

Correction factor and linearity



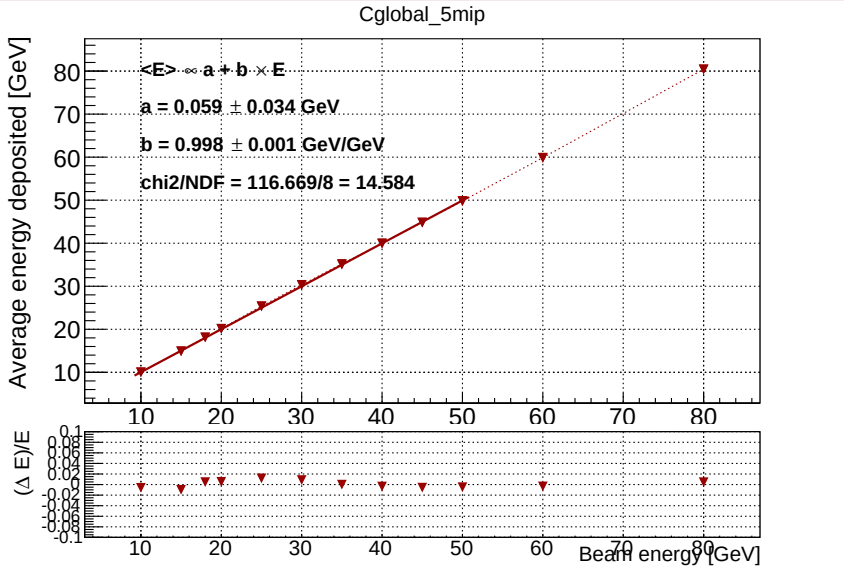




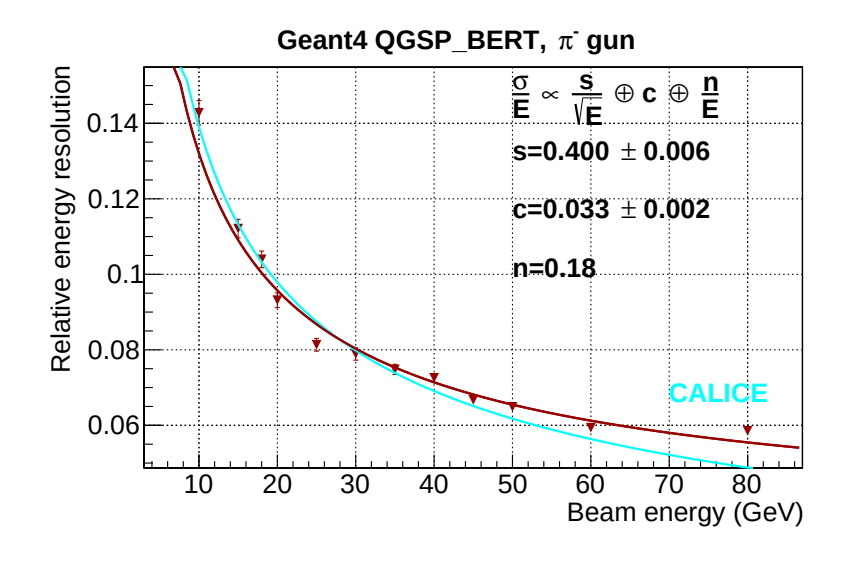
- Everything looks consistent with Calice results.

Results

Linearity

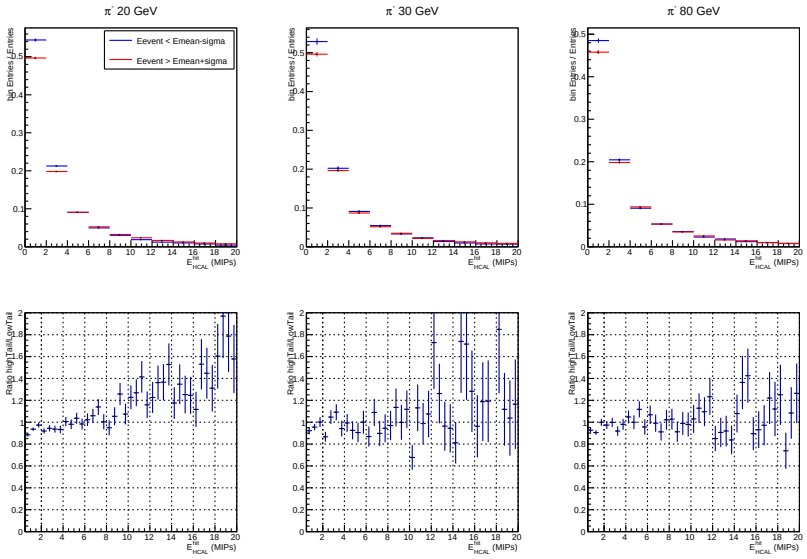


Resolution

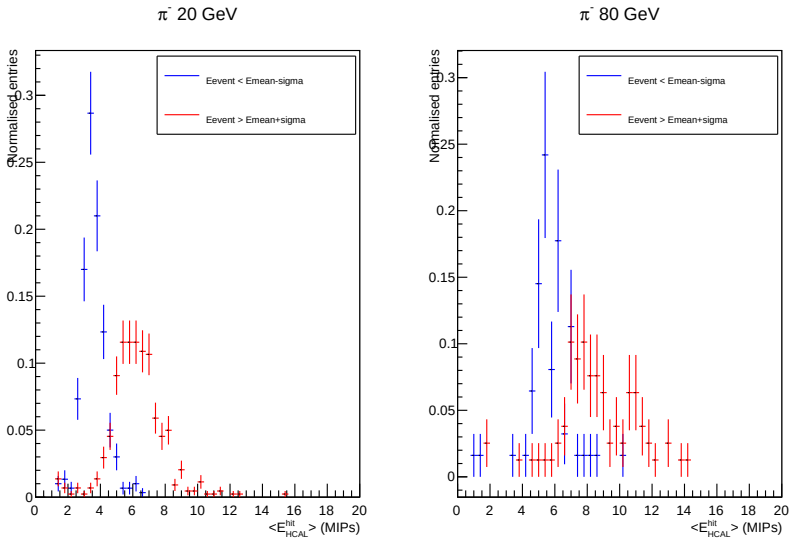


HGCAL Application: hit spectra

Hit spectra

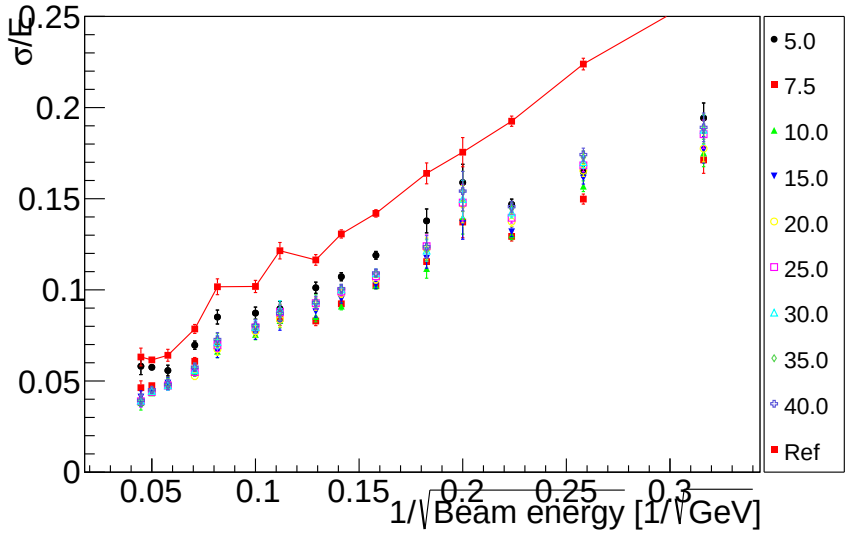


Mean/event of hit spectra



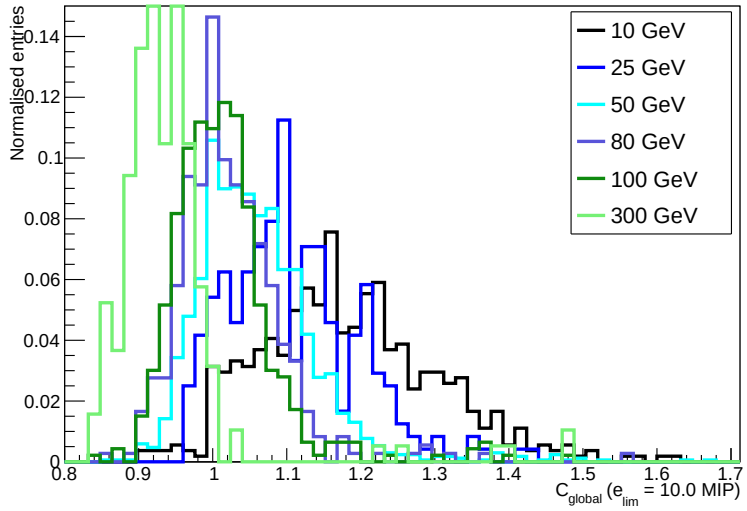
- Not enough stat to see clearly a working point :/

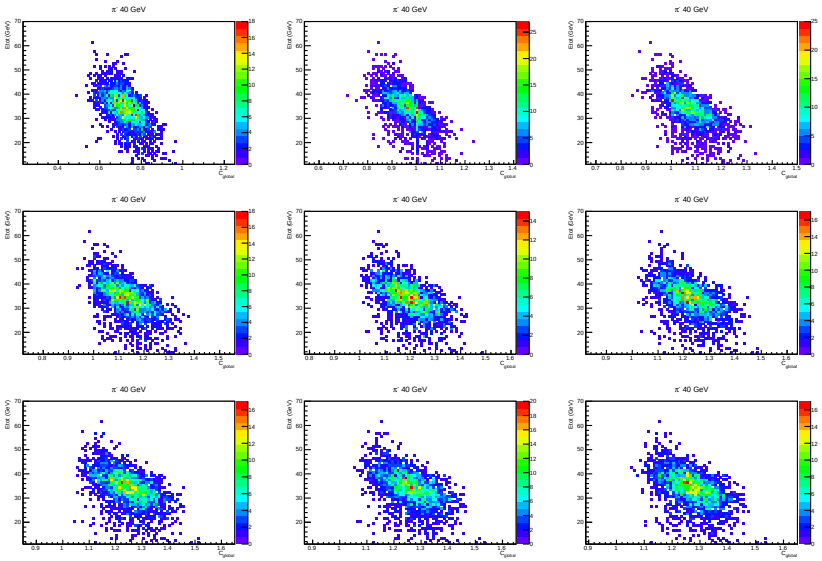
HGCAL: Tuning of e_{lim}

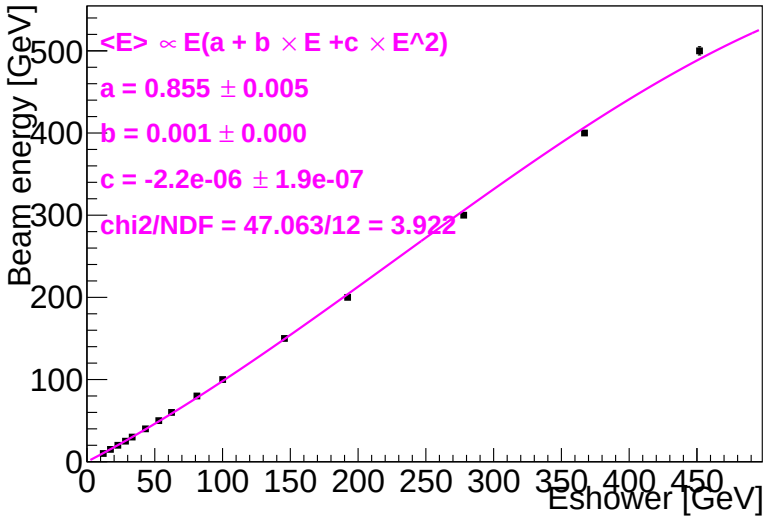


- showing here up to 500 GeV: compromise between low and high energies...
- Use $e_{lim} = 10$ MIPs in the following.

HGCAL: Correction factor and linearity

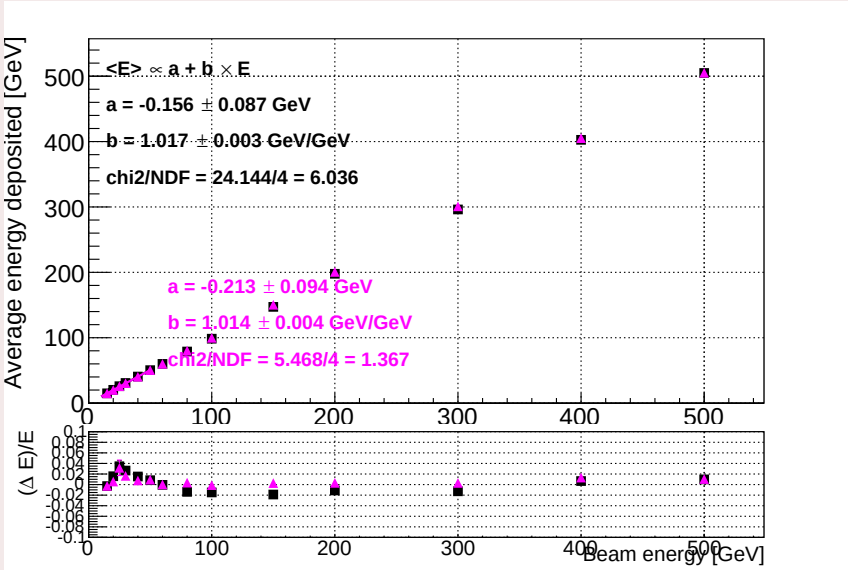




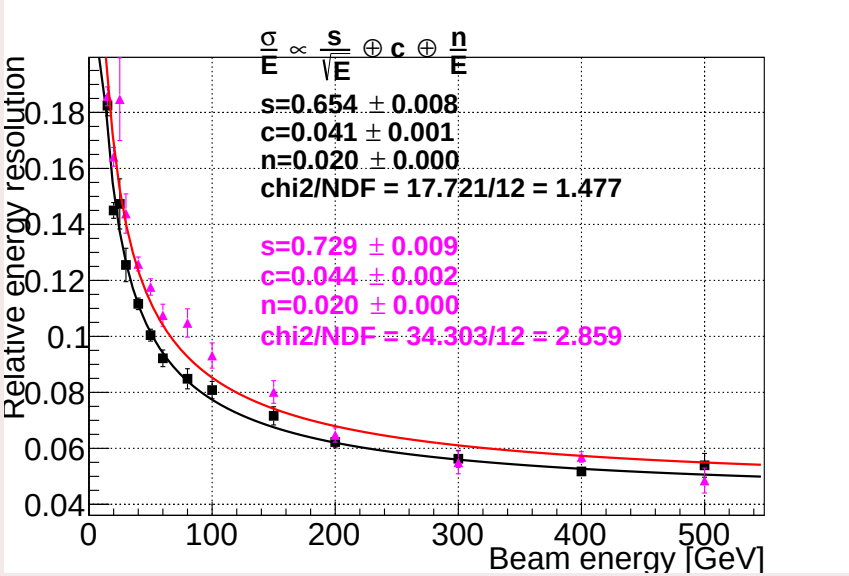


HGCAL: Results

Linearity



Resolution



- Larger gain at lower energies.
- Not as much gain as in Calice case, but no real optimisation done, and much larger energy range covered...

Conclusion

- Implemented Calice setup and digitisation in standalone simulation.
- Calice benchmark reproduced for both before and after global software compensation.
- Preliminary results with HGCAL: need more studies/tuning for our setup.
- Method can be used on PU-corrected clusters (after PUPPI-like algos).
- Need also local software compensation method: expected to perform better by itself in PU.