# Suppressing and Subtracting Pile-Up Noise

**Kyle Cranmer**
BNL

**May 4, 2006**
Hadronic Calibration Workshop, MPI

Outline:

- Quick Re-Cap on Local Noise Suppression and Relationship to TopoClusters
- Some Observations on Pile-Up Noise
- Some key points where we can improve
- Thoughts on Subtracting In-Time Pile-Up

## *Overview*

In 2004-2005 we learned a lot about noise suppression: TopoClusters and Local Noise Suppression (LNS)

Most of the work that was done was related to electronic noise: which is symmetric and has relatively little correlation from cell-to-cell

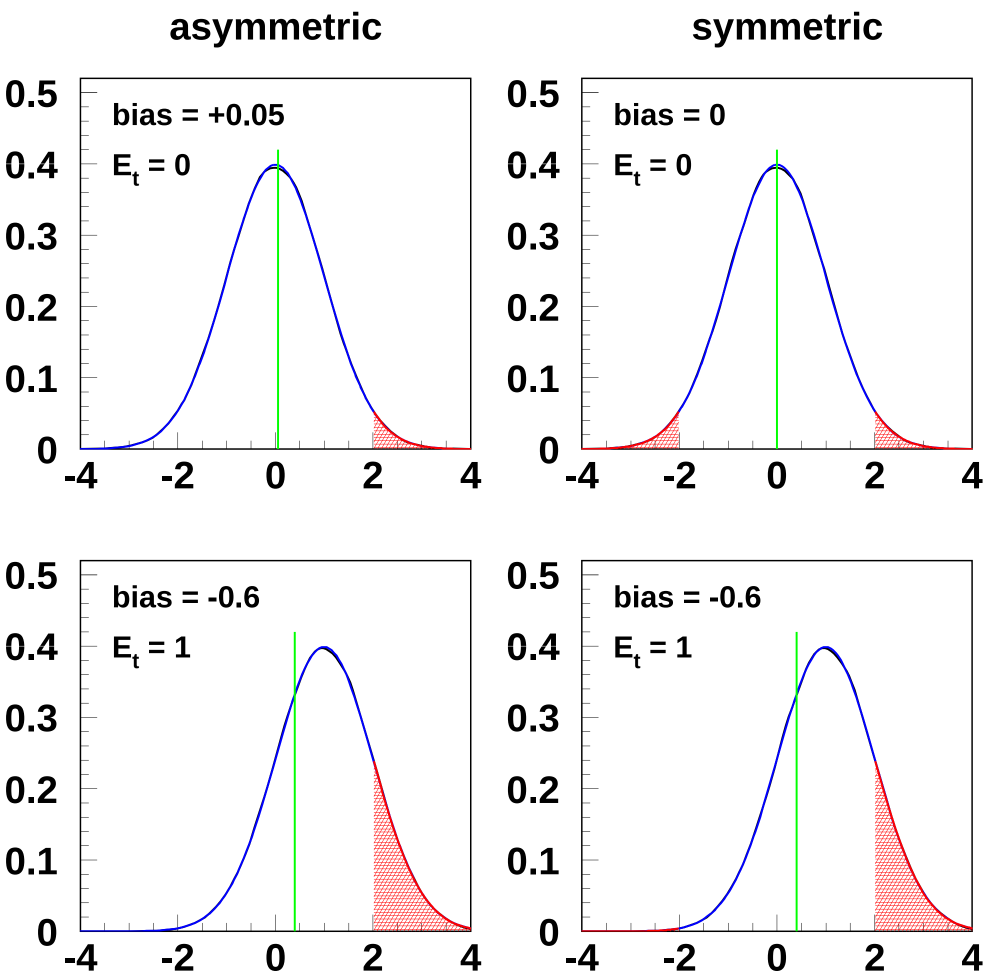
The LNS scheme starts with a formalism that makes it fairly easy to include correlations, non-Gaussian behavior etc.

The goal of this presentation is to present some observations on pile-up noise, and what we should include to try include explicitly in our attempt to suppress it.

At the end, I will mention an idea for subtracting it out of jets, etc.

# Preliminaries: Electronic Noise

## Symmetric vs. Asymmetric Cuts




When cell is empty:

- very small positive bias from asym. cut
- no bias from sym. cut

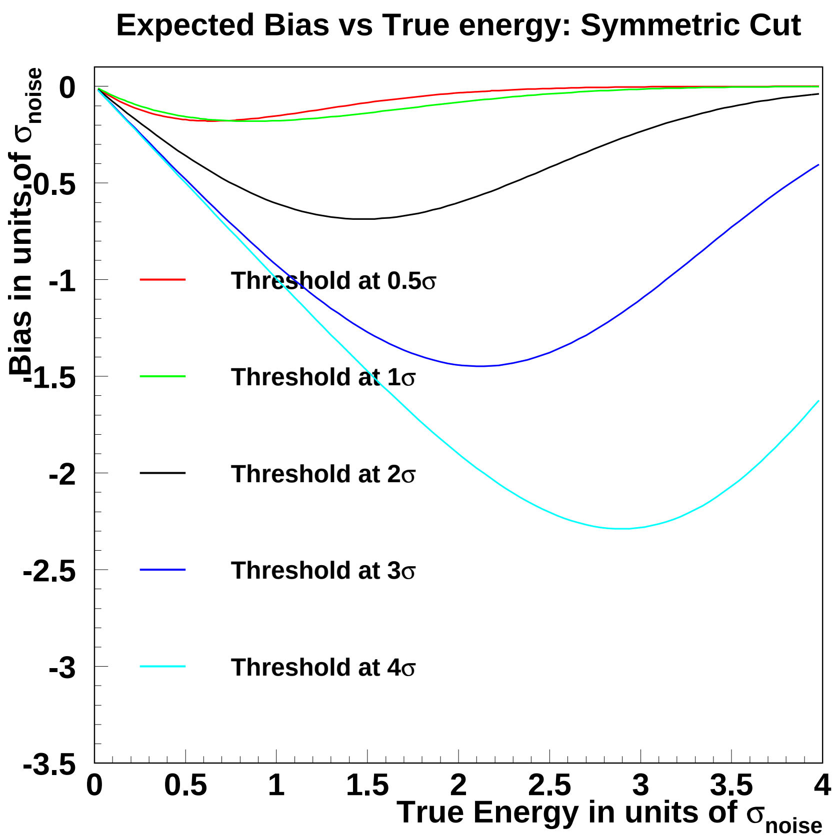
When $E_t \sim$ RMS noise:

- larger negative bias from both sym. or asym.

Green Line indicates mean reconstructed energy after noise cut

## Bias and Noise: Symmetric cut

Even Symmetric cuts cause bias at the cell level




$$\overline{\mathrm{bias}_{\mathrm{sym}}(\mathrm{E_t})} = [\bar{E}_A(E_t) - \bar{E}_A(-E_t)] - E_t$$

$$= \frac{\sigma_n^2}{\sqrt{2\pi\sigma_n^2}}\left[e^{-\frac{(N\sigma_n - E_t)^2}{2\sigma_n^2}} - e^{-\frac{(N\sigma_n + E_t)^2}{2\sigma_n^2}}\right]$$

$$-\frac{E_t}{2}\left[2 + \mathrm{erf}\left(\frac{N\sigma_n - E_t}{\sqrt{2\sigma_n^2}}\right) - \mathrm{erf}\left(\frac{N\sigma_n + E_t}{\sqrt{2\sigma_n^2}}\right)\right]$$

Conclusion: cutting out cells with real energy causes bias. TopoClusters and LNS try to avoid cutting out cells with real energy by looking at neighbors.

## Using Bayes Theorem

In my formalism, I've generalized the discontinuous noise threshold, to be a continuous *Noise Suppression*.

$$\text{Assume } P(E_{meas}|E_{true}) = \frac{1}{\sqrt{2\pi\sigma_n^2}} e^{-\frac{(E_{meas}-E_{true})^2}{2\sigma_n^2}}$$

Use Bayes theorem to invert $E_{true} \leftrightarrow E_{meas}$

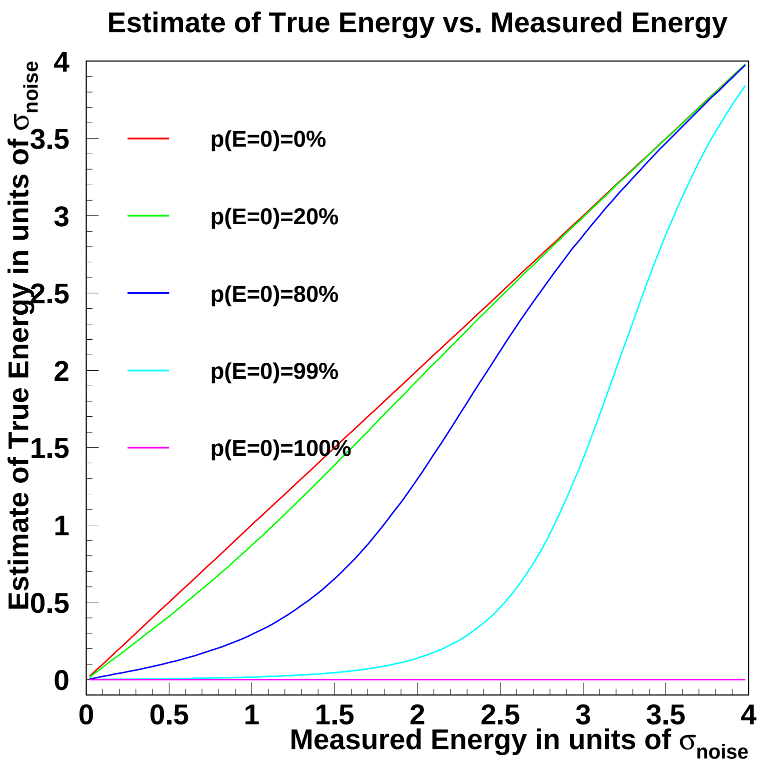
$$p(E_{true}|E_{meas}) = \frac{p(E_{meas}|E_{true})p(E_{true})}{p(E_{meas})}$$

$$\text{Assume } p(E_{true}) = a_0\delta(0) + \text{flat prior elsewhere}$$

The prior $a_0 = p(E = 0)$ is the important part. I will make it *local*

Note: Many places include effect of pile-up: $P(E_{meas}|E_{true})$, $\sigma_n$, & $p(E_{true})$

The plot shows an estimate of $E_{true}$ given the measurement & several values of $a_0$




$$\bar{E}(E_m) = \frac{0 \cdot a_0 e^{-E_m^2/2\sigma^2} + (1-a_0)E_m}{(1-a_0) + a_0 \cdot e^{-E_m^2/2\sigma^2}}$$

Notice that:

$\lim_{E_m \to \infty} \bar{E}(E_m) = E_m$

$\lim_{a_0 \to 0} \bar{E}(E_m) = E_m$

$\lim_{a_0 \to 1} \bar{E}(E_m) = 0$

This estimator acts like a softer nose cut.

The key is to adjust $a_0$ (the prior) given the surrounding cells.

## *Using Neighboring Cells to Estimate Prior*

1.) For each cell, get the neighboring Calo Cells (2d, 3d, or super-3d)

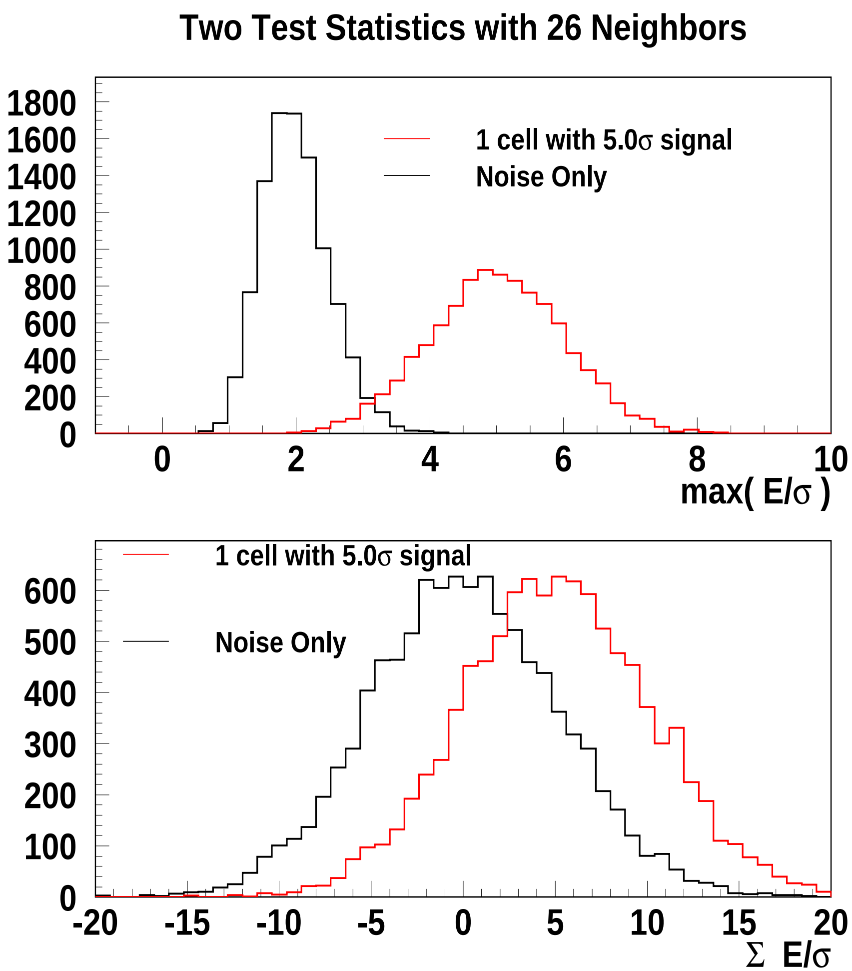
2.) Calculate $\mathcal{X} = \sum_{i=1}^{N_N} E_i/\sigma_i$

$\Rightarrow$ If all cells only have noise, the distribution of $\mathcal{X}$ should be a Gaussian with $\mu = 0$ and $\sigma = \sqrt{N_N}$.

3.) Estimate $a_0$ (prior that a cell is just noise) to be the probability that noisy neighbors produce an $\mathcal{X}$ greater than the measured $\mathcal{X}$

$\Rightarrow$ If $\mathcal{X}$ is small probably noise. if $\mathcal{X}$ is large, probably signal.

Note: If signal's energy is evenly distributed in neighboring cells, this is the most powerful variable to separate signal from noise




Originally I used $\mathcal{X} = \sum(E_i/\sigma_i)$ as a test statistic (i.e. the quantity to discriminate between signal and noise)

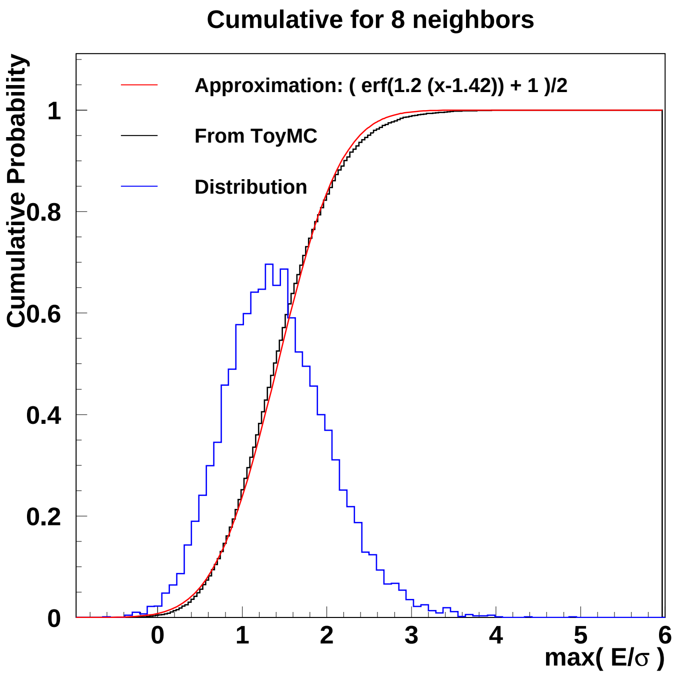
But $\mathcal{X}$ is not the most best variable near the outside of a cluster

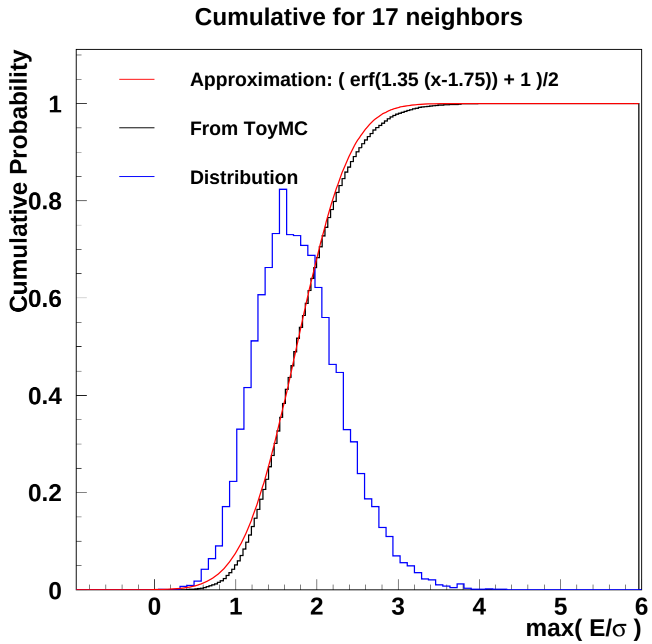
With 26 3-d neighbors, one cell with $\sim 5\sigma$ doesn't impact $\sum(E_i/\sigma_i)$ much (see bottom plot)

For only one neighbor cell with $E > 0$ $\mathcal{X}' = \max_i(E_i/\sigma_i)$ has better discrimination

Note: With pile-up noise, the noisy background will be correlated; however, this is the place to include that correlation explicitly

## *Dependence on number of neighboring cells*




Can write analytically the distribution of $\mathcal{X} = \sum(E_i/\sigma_i)$

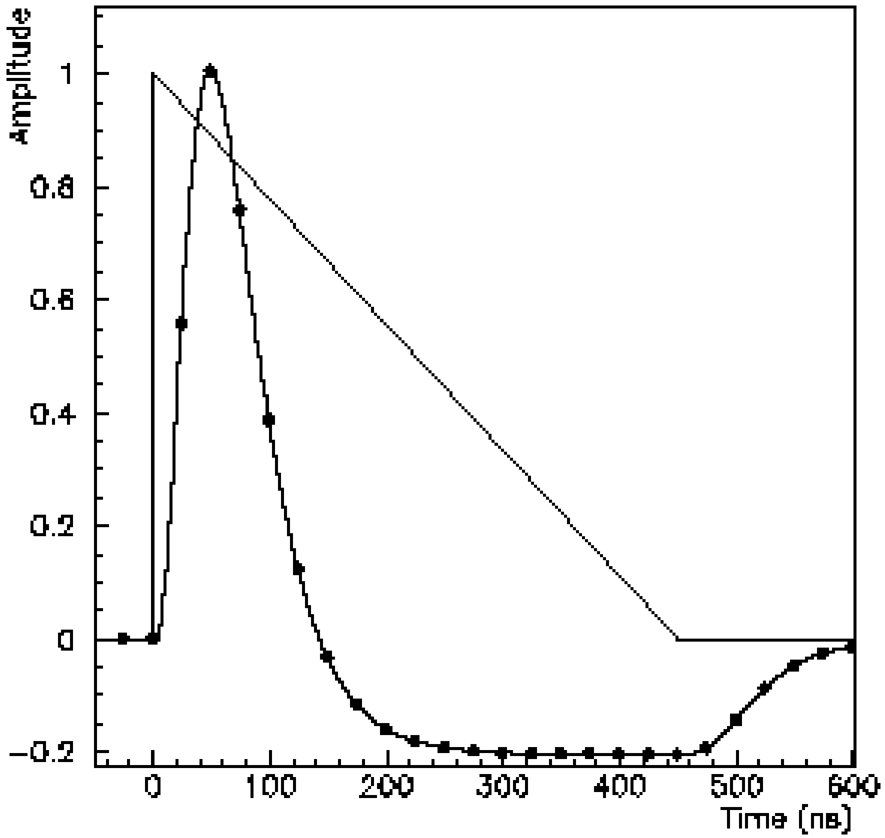
Hard to write analytically the distribution of $\mathcal{X}' = \max_i(E_i/\sigma_i)$

I have an approximation of the cumulative for various numbers of neighboring cells. This values of these fits are used by the LNS tool, configured in jobOptions

This is one place TopoClusters can be improved

# Pile-Up in the LAr

## *The LAr Pulse Shape*




The RMS of the the pulse amplitude is

$$I = \sqrt{\sum_i^{N_{samples}} g_i^2} \approx 1.5$$

The CaloNoiseTool includes the variance in the pile-up energy depositions ($\sigma_E$) and the number of min. bias interactions ($N_{MB}$) in the noise estimate and the autocorrelation function (see M. Lechowski's thesis [CERN-THESIS-2005-042] ):

$$\langle S_i S_j \rangle = N_{MB} \sigma_E^2 \sum_k^{N_{samples}} g(i-k) g(j-k)$$

Note: the distribution of $S_i$ only looks Gaussian when $N_{MB} \gg 1$

## Distribution of Energy in a LAr Cell

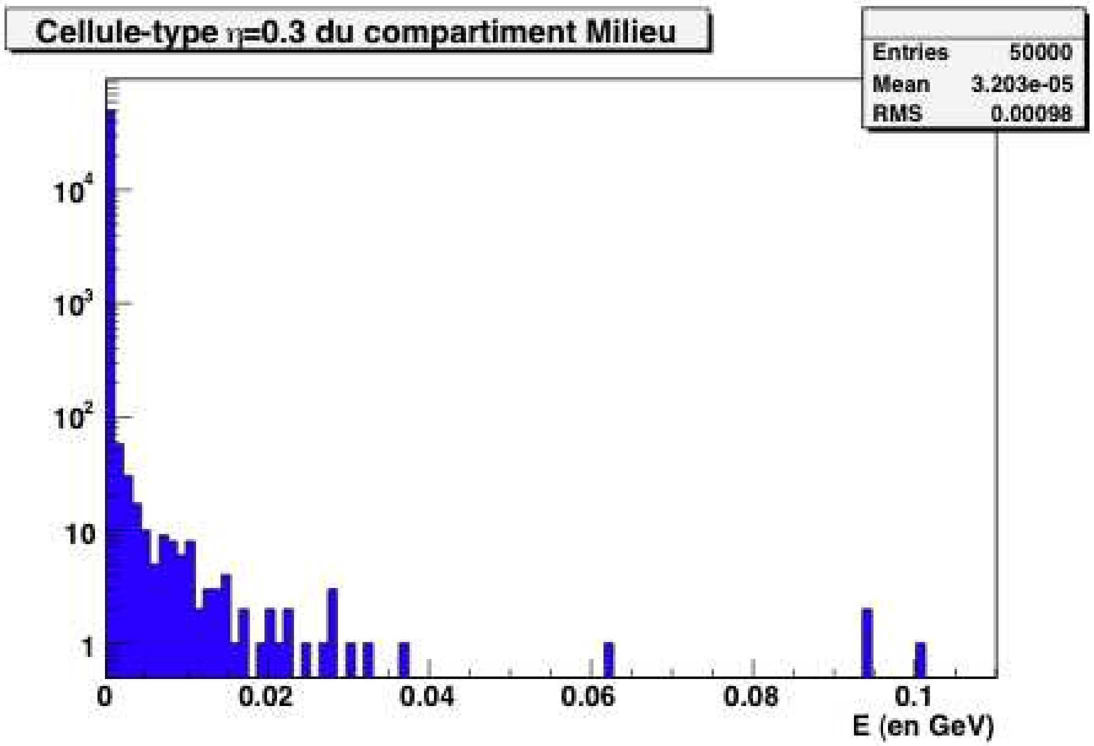



FIG. 3.19 – *Energie déposée dans l'Argon liquide par événement de biais minimum, pour une cellule ($\eta$ = 0.3) du compartiment Milieu dans le Tonneau du calorimètre électromagnétique*

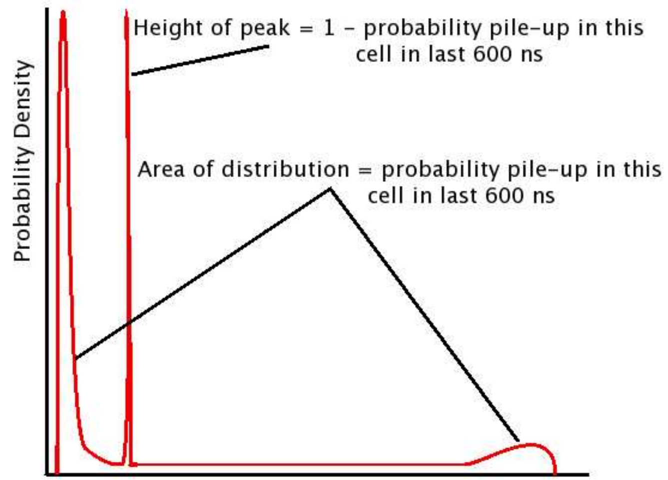
From the distribution, we can see the chance for a cell to have an energy deposit is $\approx 10^{-3}$

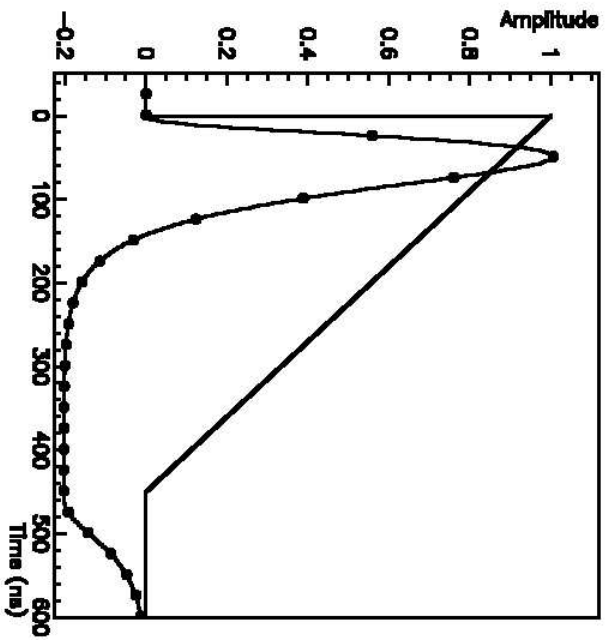
Assuming 16 particles per unit $\eta$ one predicts 0.16% chance to transvers a region of $0.025 \times 0.025$ in $\eta - \phi$ *in a single crossing*

For 2.3 interactions per crossing, expect 1.4 events in $0.1 \times 0.1$ region in a 600ns window

Conclusion: $N_{MB} \not\gg 1$, so don't expect anything to be Gaussian

## *Translating from Pulse Shape to Energy Distribution*




The left figure shows how the pulse shape translates to a very non-Gaussian energy distribution. (To simplify things, I'm not using OFCs, I'm just using the height of the nominal peak sample.)

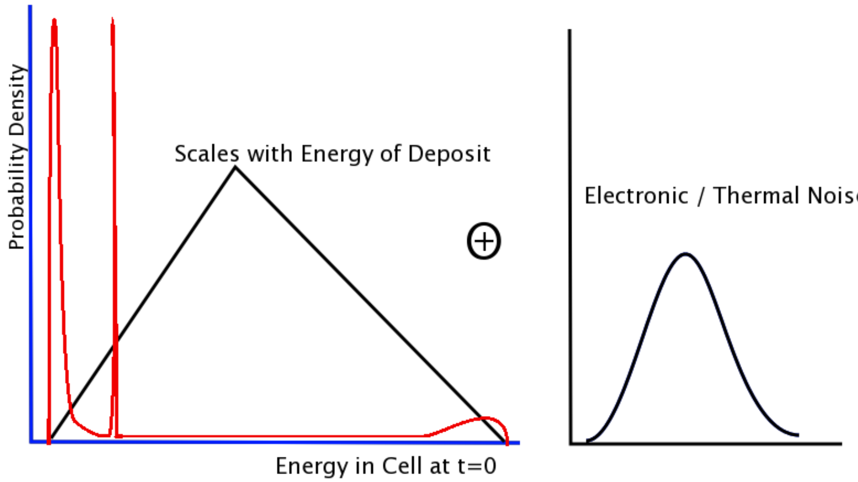
There will be a large spike at $E = 0$, corresponding to the cell not being hit in the 600ns window.

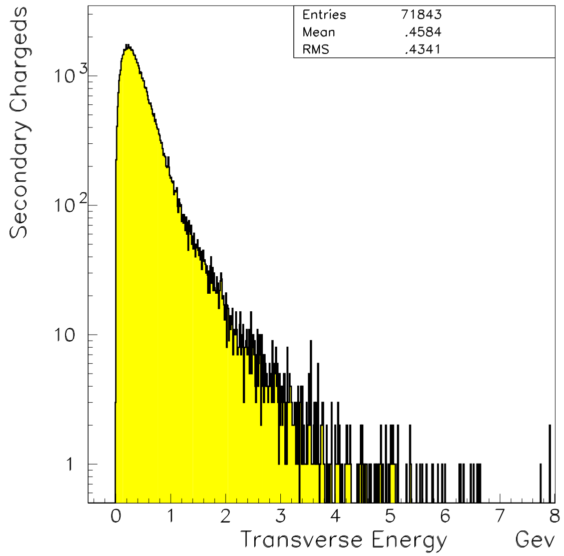
- For $L = 10^{31}$, the spike dominates.
- At $L = 10^{33}$ they are comparable.
- At $10^{34}$, the spike is negligible.

The continuous shape is also an idealization. Given discrete 25ns samplings, the distribution will be spiky (more realistic plots in a few slides)

The low energy cells probably won't have $t, \chi^2$, so we may only have energy to work with

Of course, this pile-up noise is convoluted with the electronic/thermal noise



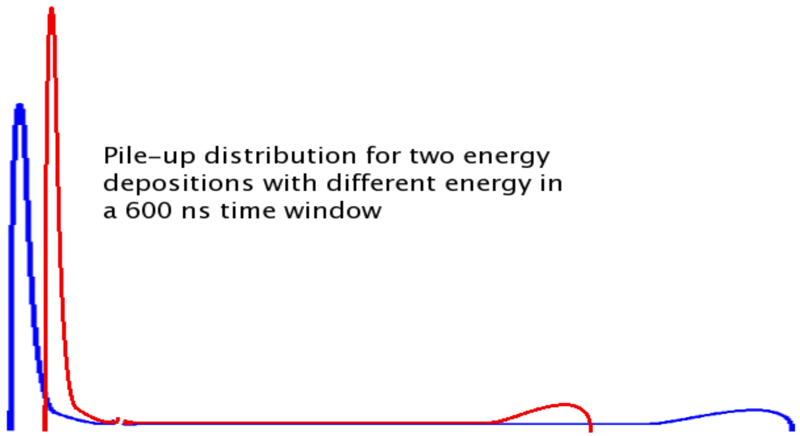

The magnitude of the pile-up noise will scale with the energy of the original deposit, while the electronic noise is constant.

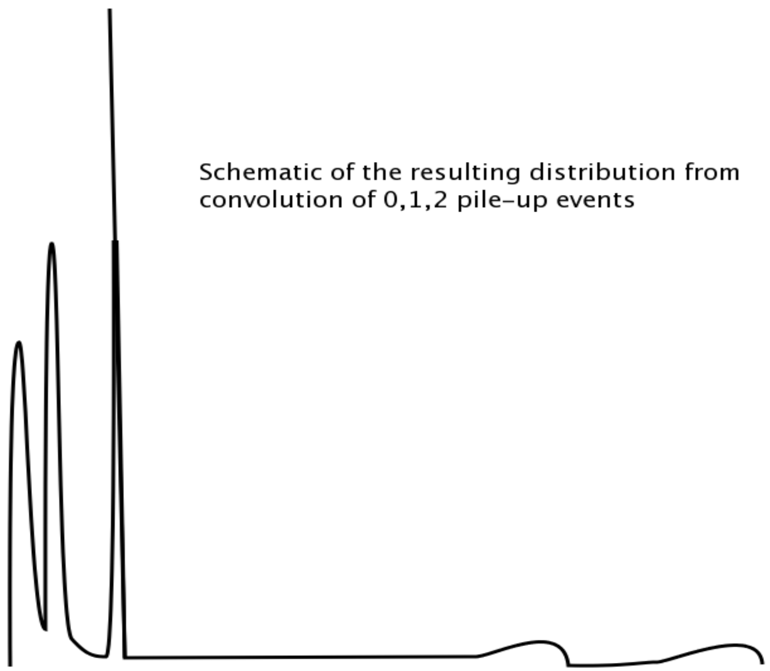
Folding the pile-up $p_T$ is not a convolution, and the spectrum is very non-Gaussian... so we need to use Monte Carlo to do that properly

## Poisson Fluctuations

And when the number of pile-up events hitting a cell in the 600ns window is $\mathcal{O}(1)$, we need to worry about Poisson fluctuations.

The bottom left figure shows the energy distribution for two pileup events with different energies, each randomly distributed in 600ns window. The bottom right shows (schematically) what the distributions would look like with Poisson fluctuations when the mean is near 1.

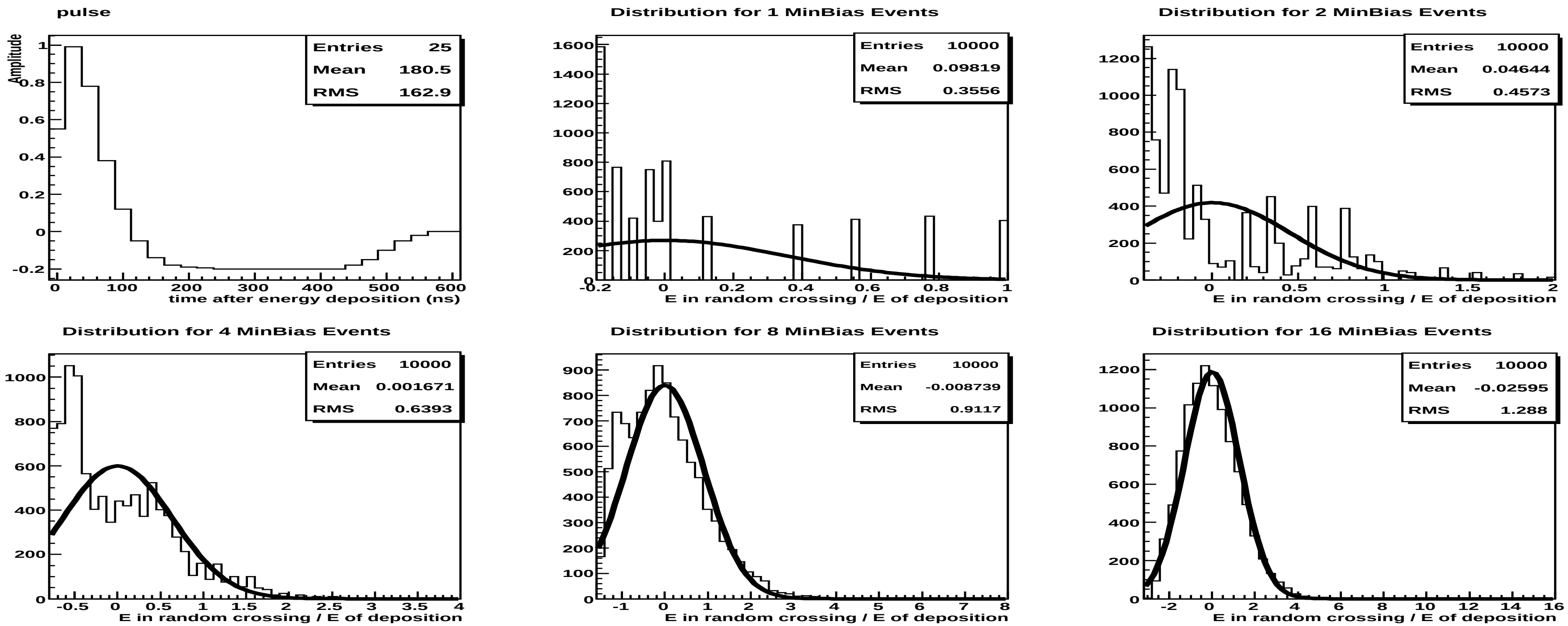


## Some Toy MC results (no electronic noise, trivial $p_T$ spectrum)

Top Left: Pulse Shape for 600ns sampled every 25ns

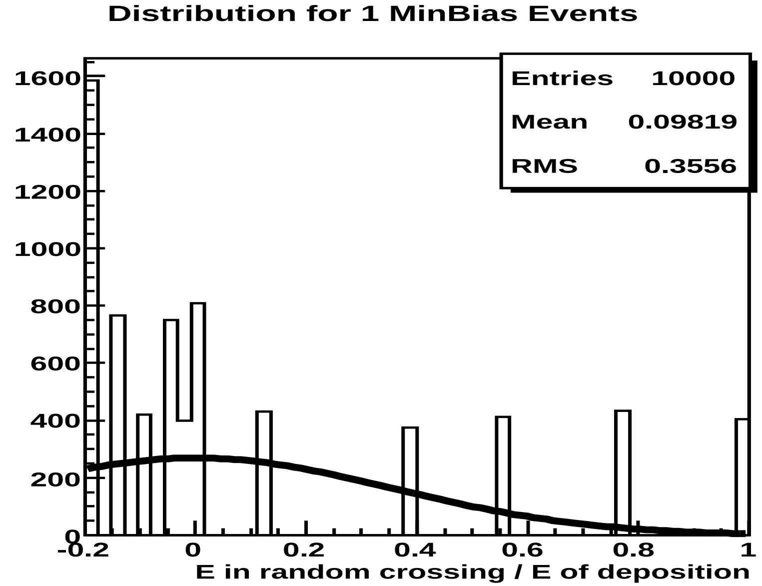
Other Plots: Energy distribution from randomly sampling pulse shape $N$ times

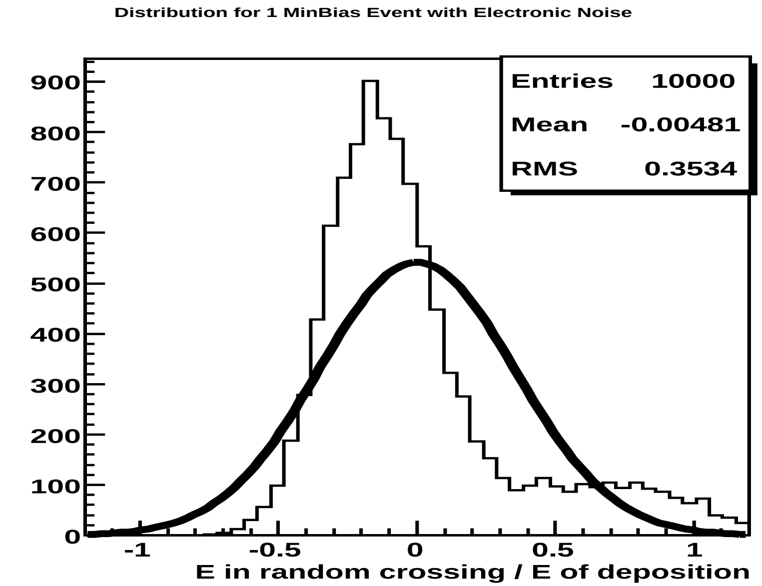



Conclusion It's not until we get to about 10 pileup events in 600ns window that things look Gaussian. For 1 event, the Gaussian approximation is horrible!!!

Note: Structure will be smeared out by $p_T$ spectrum of pile-up and electronic noise, but it won't be Gaussian

# *Measuring Pile-Up Noise*




When there is exactly one pile-up event in the 600ns window, the energy distribution (before a convolution with electronic noise) looks like the plot on the top left

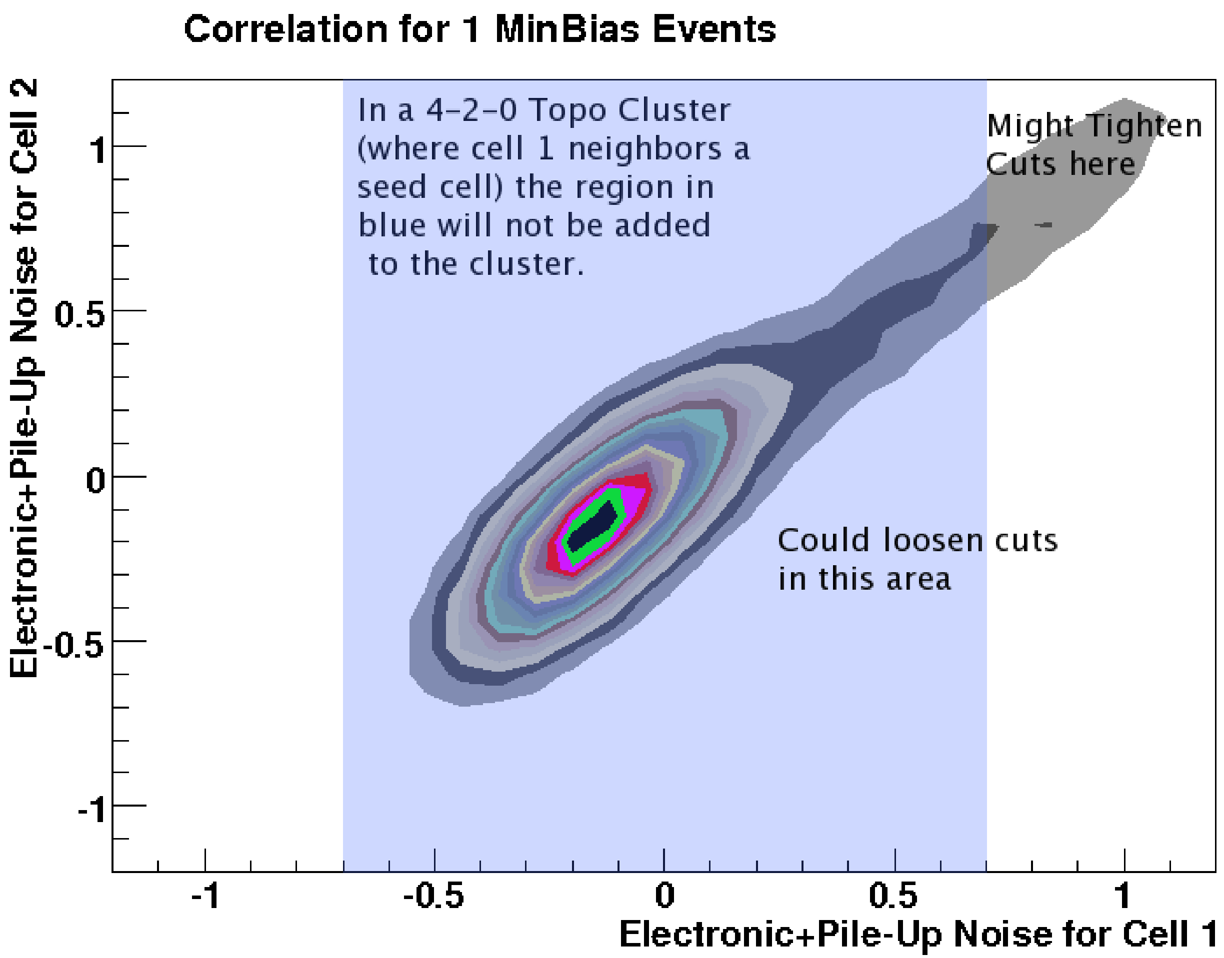
After a convolution with electronic noise (when original energy deposite $2 \times \sigma_{noise}$) it looks like the bottom left still non-Gaussian.

The curves are Gaussians with the right RMS and mean=0. **Isn't this what the CaloNoiseTool uses to describe the pile-up + electronic noise?**

Do we feel confident that we can predict this? We can probably measure the pile-up noise with random triggers.

# Ways to Improve Suppression of Pile-Up Noise

Assuming electronic noise is uncorrelated and pile-up noise is totally correlated (and twice as large as electronic noise for this example), one can see the areas that the TopoCluster or an LNS-inspired algorithm can improve:



# Scaling of Pile-Up Noise With Area

Interesting ATLAS note by J. C. Chollet, CAL-NO-75 (1995)

Simple model that predicts that RMS in pile-up noise doesn't scale like $\sqrt{area}$

$$= \sum_{N=0}^{\infty} P_N * \sum_{K=0}^{\infty} K*(K-1)*\frac{N!}{(N-K)!*K!}*a^K*(1-a)^{N-K} \quad -E(K)-E^2(K)$$

$$= \sum_{N=0}^{\infty} P_N*N*(N-1)*a^2 \underbrace{\sum_{K=2}^{\infty} \frac{(N-2)!}{(N-K)!*(K-2)!}*a^{K-2}*(1-a)^{N-K}}_{=1} \quad -E(K)-E^2(K)$$

$$= E(N^2)*a^2-E(N)*a^2+a*E(N)-a^2*E^2(N) = a^2*V(N)-E(N)*a^2+a*E(N)$$

$$V(K) = a^2*(V(N)-E(N))+a*E(N)$$

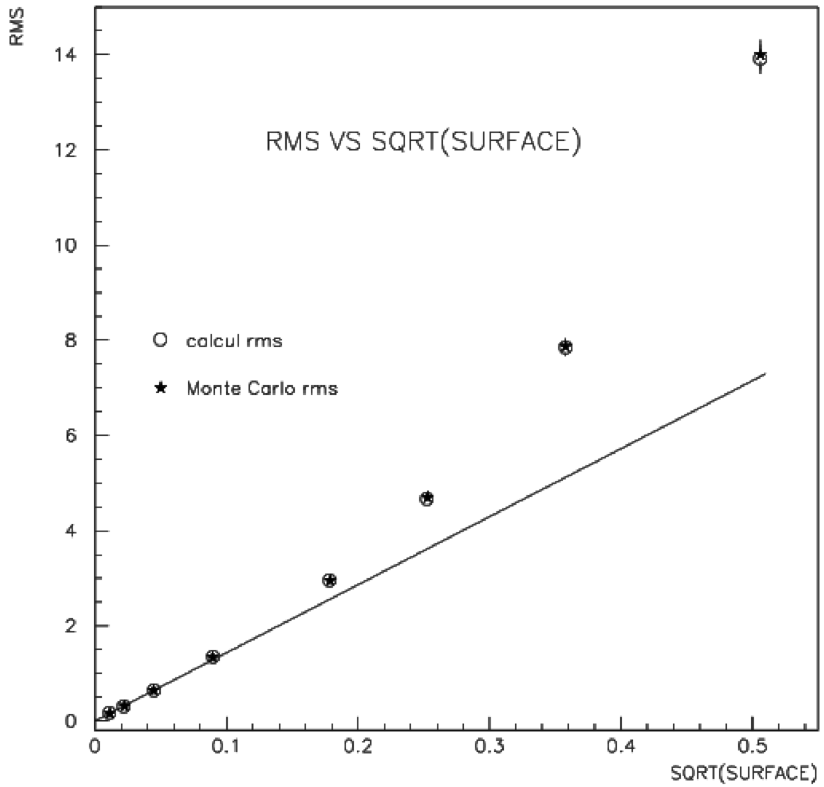
Already here the variance has two components.One proportionnal to a. The other to $a^2$.It is only when V(N)=E(N) (Poisson distribution for instance)that the particles number rms grows with the $\sqrt{surface}$
With the previous result

$$E(N) = mt*mb \qquad V(N) = (mt^2-mt)*mb$$

$$E(K) = a*mt*mb$$

$$V(K) = a*mt*mb*(1-a*mt)$$

The term $a^2*mt^2*mb$ comes from the distribution of minibias events. The other $a*mt*mb$ results from the fluctuation of particles hitting the surface per event

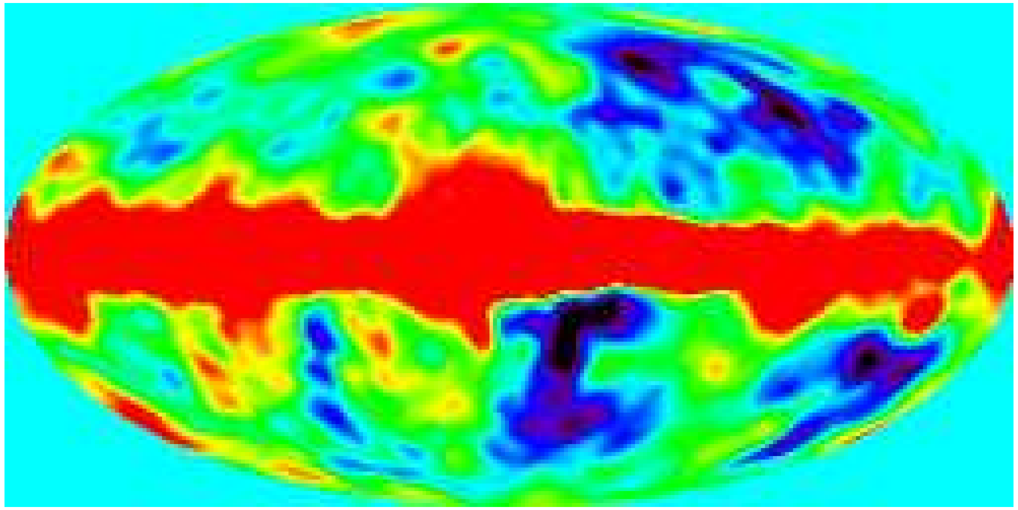


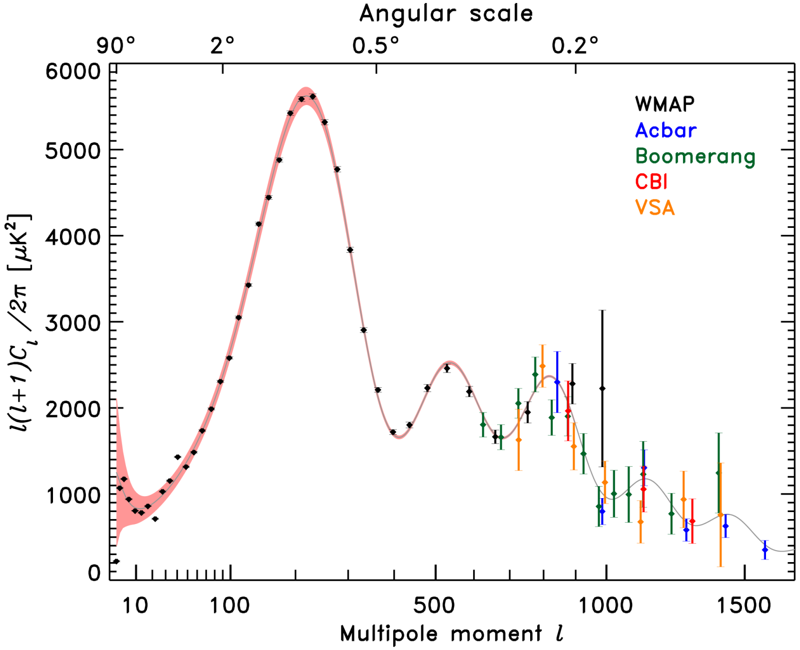

This was observed in Monte Carlo, but this model helps explain why

# Subtracting In-Time Pile-Up

## *Inspiration from Cosmology*

When cosmologists look at the CMB, they decompose the sky into a power-spectrum based on spherical harmonics (a linear, complete basis).





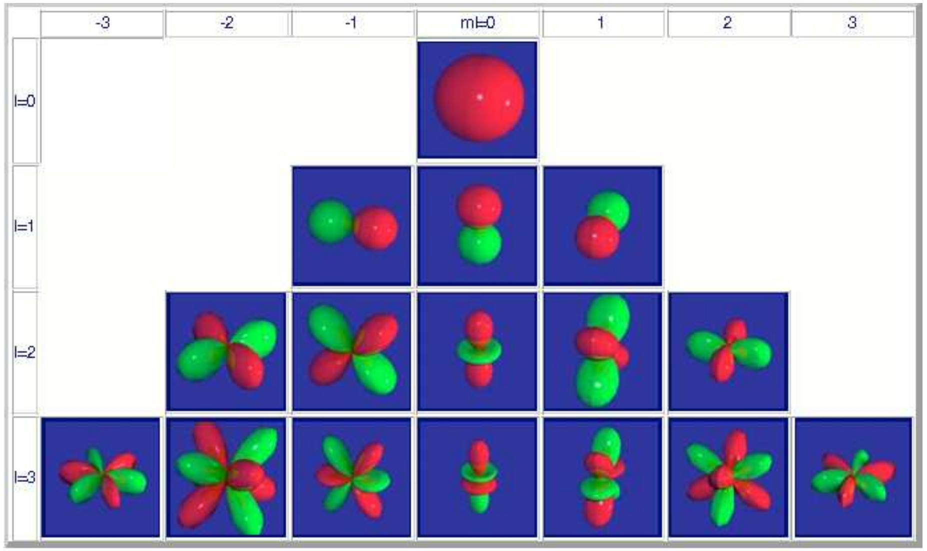

The energy distribution in the calorimeter from a minimum bias interaction can be treated the same way, but it would be nice to have a more QCD-inspired basis (with $\phi$ invariance).

We could then subtract it from "physics" (except for non-linear detector effects)

The big difference is that there's only one sky!

## Decomposing Minimum Bias

If we decompose many min. bias events into some suitable basis, we can begin to build *distributions* of the power-spectrum with correlations across the various "multipole moments"




At the least, we can use these descriptions to subtract min. bias from a jet in an average way.

We can also use the variation in the moments to estimate systematic uncertainty on the subtraction

Ultimately, we could look at the energy deposits known to come from pile-up (e.g. via tracking) to constrain some moments and subtract based on the conditional distribution. (e.g. if the pileup looks like it's forming mini-jets, then one would take that into account in the subtraction procedure).

## Conclusions

Local Noise Suppression has a nice statistical framework that makes it fairly clear where to include certain effects (correlations, non-Gaussian distributions, etc.) in an explicit way.

LNS and TopoClusters are quite similar in terms of noise suppression for pure-electronic noise, but LNS has smooth turn-on curves which depend on the number of neighboring cells

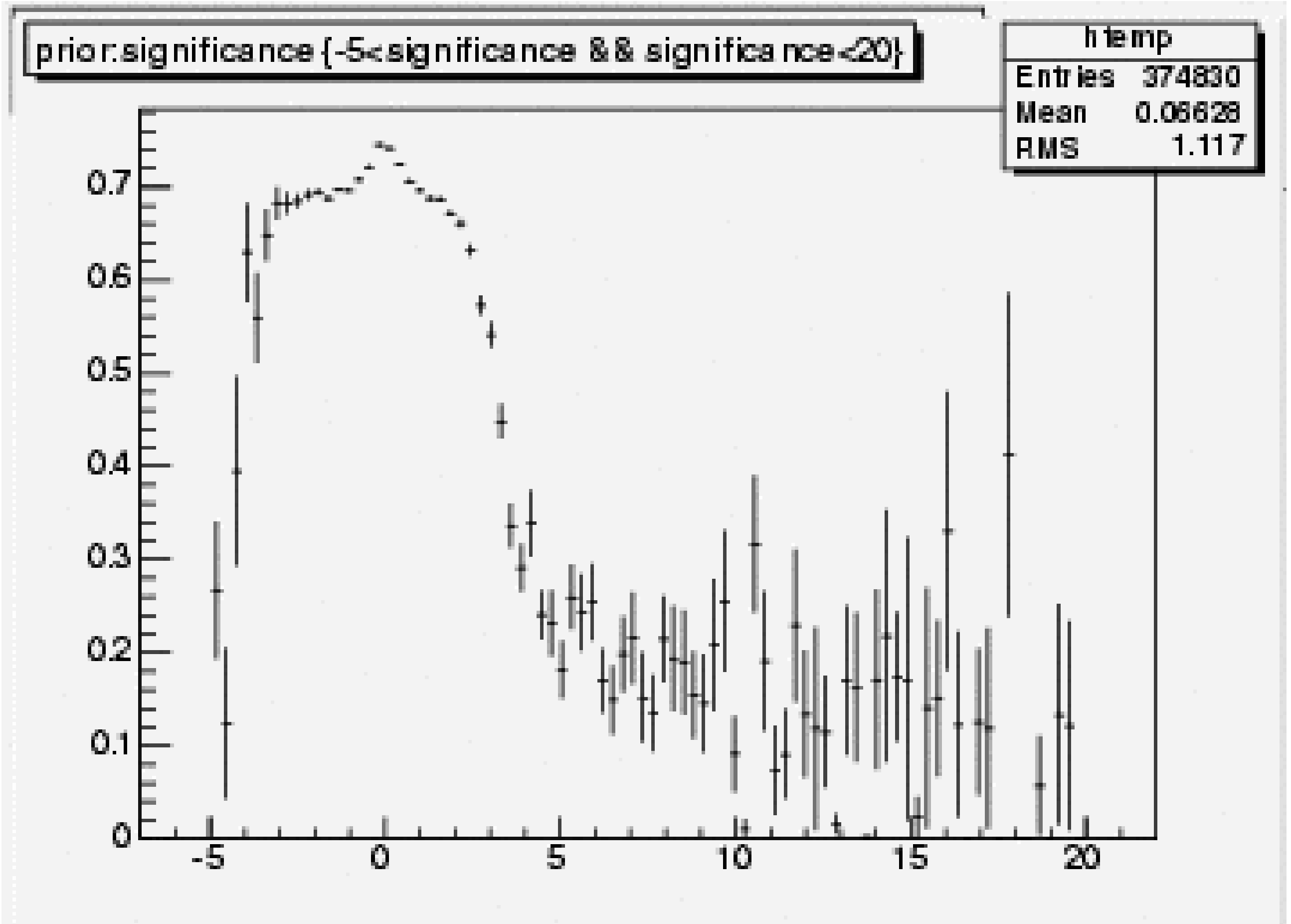
The pile-up noise is very non-Gaussian, and suffers from large Poisson fluctuations at low luminosity. There are some fairly clear places we can improve on what we are doing now.

Predicting the pile-up noise from the $p_T$ spectrum of min bias, LAr pulse shapes, shower shapes, and electronic noise sounds hard... we probably want to measure it. (We should start practicing now.)

Subtracting in-time pile-up is a different beast. I have presented some very speculative ideas about how we might utilize a linear basis and distributions of "multiplole moments"

# Backup and Appendix

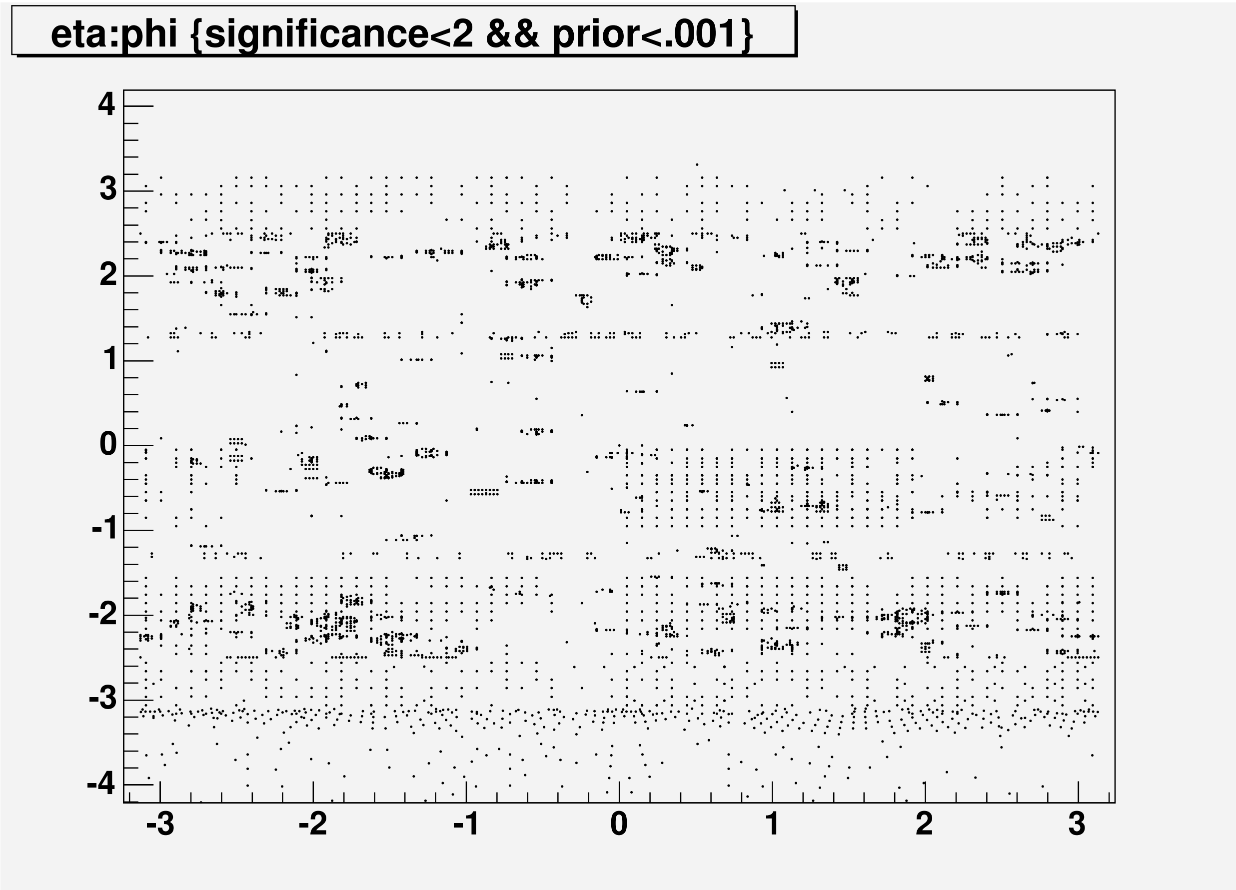
## *Prior vs. Significance*




Prior is $\sim 1$ for 75% of cells with significance $< 3$

Prior is $\sim 0$ for 80-90% of cells with significance $> 5$

## *Biased Cells*

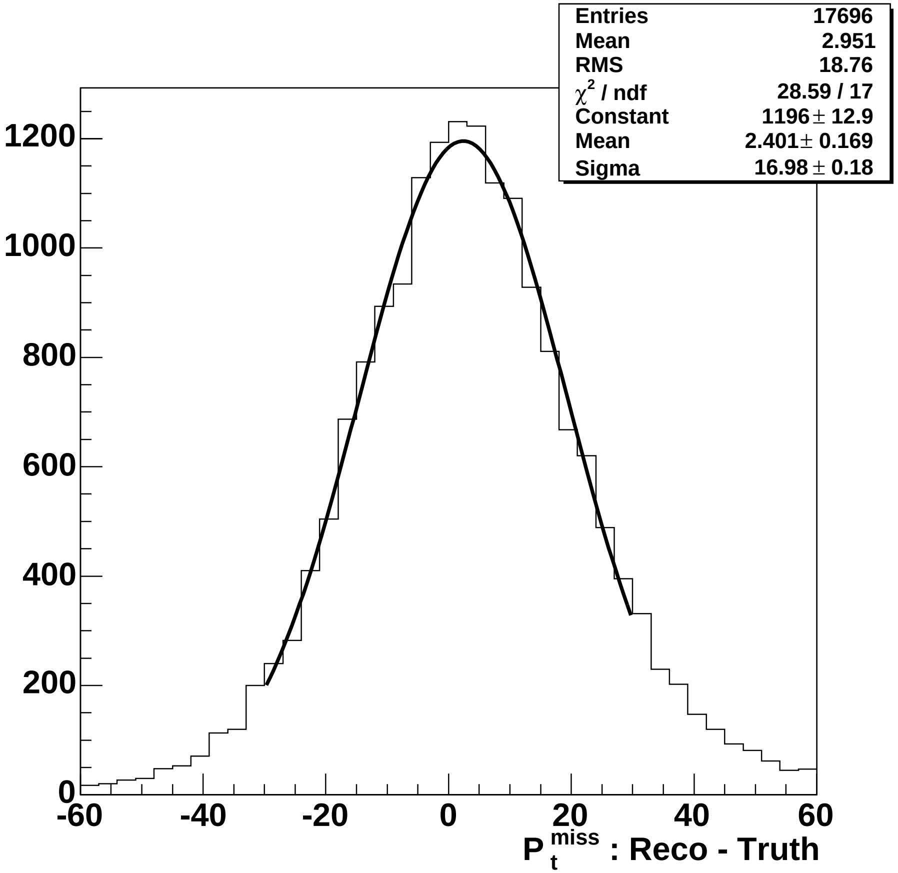


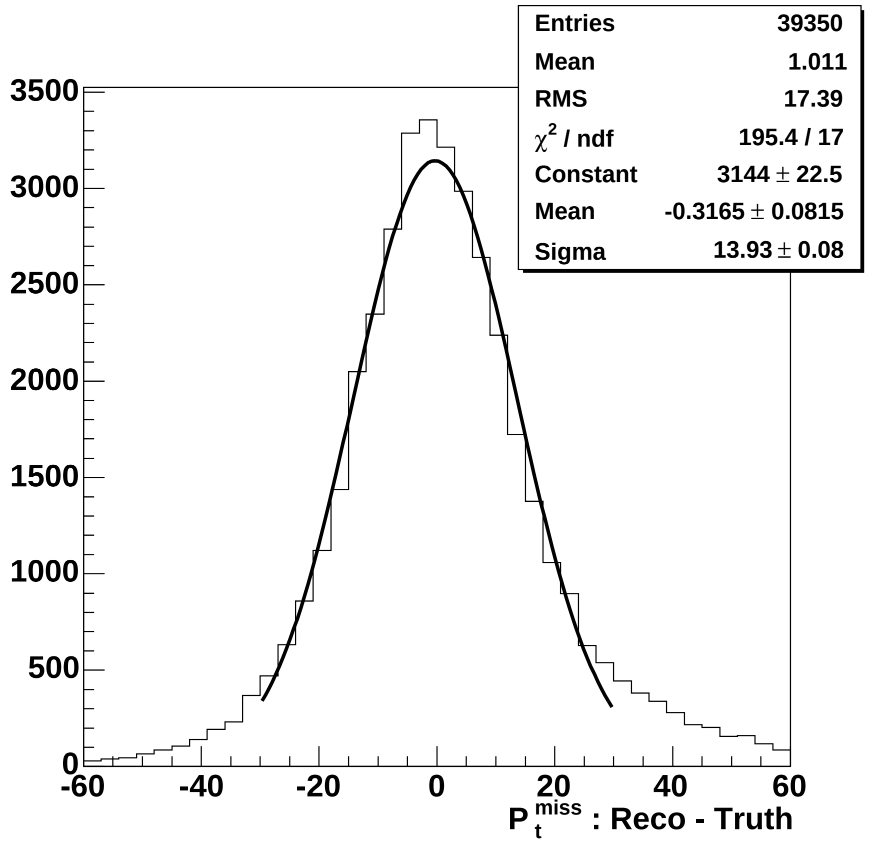

These are Cells that would be cut by a $2\sigma$ noise threshold, but that I keep.

I suspect these are the cells causing bias.

## Final $p_T^{miss}$ resolution

Using G4Beta2 Weights



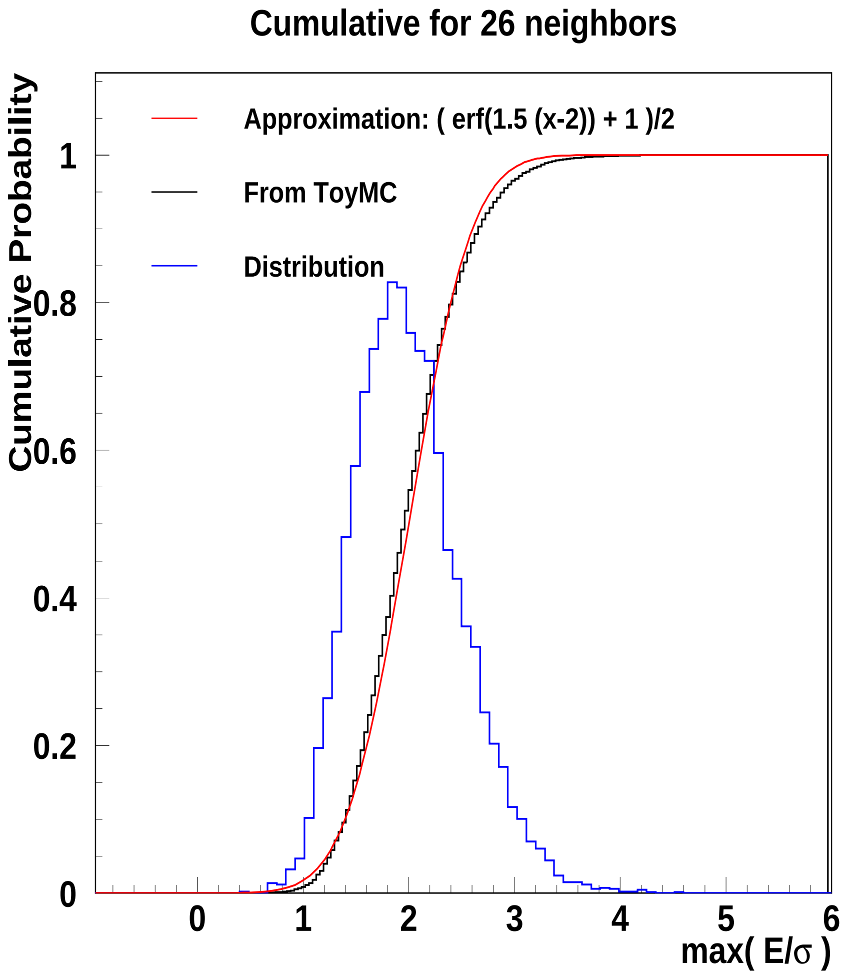




Global $2\sigma$ cut on LAr

Local Noise Suppression

Local Noise Suppression improves resolution, reduces bias

## *Improvements*




Easy to write analytically the distribution of $\mathcal{X} = \sum(E_i/\sigma_i)$ if all neighbors are noise only

Hard to write analytically the distribution of $\mathcal{X}' = \max_i(E_i/\sigma_i)$ if all neighbors are noise only

I have an approximation of the cumulative for various numbers of neighboring cells. This values of these fits are used by the tool, configured in jobOptions