ECAL time reconstruction using weights

29 July 2013

Kostas Theofilatos

Time reco with weights

- Purpose of this study: provide alternative time reco using weights to aid the discussion evolving in hn

- Weights algo to reconstruct amplitude *& time* see: **[Eur. Phys. J. C 46, s01, 23–35 (2006)]**
- Software generating the time weights was fixed few weeks ago *broken since (CMSSW_0XX)*
- Tested (so far) only for Gain 12 signals

- ***Disclaimer:*** *this is a first attempt trying to reconstruct ECAL time of LHC data using the weights Algo, what's reported here should be consider preliminary*

How-to use the weights

- $\mathbf{S_i}$ denote the non pedestal subtracted MGPA samples (10 integer ADC counts) e.g. S[] = {210, 209, 209, 218, 548, 650, 602, 510, 423, 355} with pedestal ~210
- $\mathbf{w_i}$ denote weights for amplitude reco (10 floating numbers)
- $\mathbf{t_i}$ denote weights for time reco (also 10 floating numbers)
- **A** denotes the reco amplitude **δt** denotes signal's jitter
- **A** and **δt** are evaluated from the inner products
  - $\mathbf{A = w_i * S_i}$
  - $\mathbf{A*\delta t = t_i * S_i}$ (thus we need to divide with **A** the $\mathbf{t_i * S_i}$ to get **δt**)
- **δt** has units of ADC clock [25 ns], in CMSSW **uncalibTime** = **-δt*25**
- **offline time = uncalibTime + timeIC** (offline time intercalibration)
- We have **timeIC** only for Ratio Algo, no **timeIC** is available for weights algo

Generated weights (1/2)

- **3+5 amp & time jitter compensating weights** have been generated for **Data/MC** and **EB/EE** using pulse shapes hardcoded in **CMSSW** (Data have 2ns phase)
- We know that **pulse shape in CMSSW is imperfect**
  - expect some amplitude dependence in reco time
  - we can apply EB weights to EE digis and MC weights to Data (+2ns offset) to get a feeling of the systematic uncertainties due to the imperfect knowledge of shapes

Generated weights (2/2)

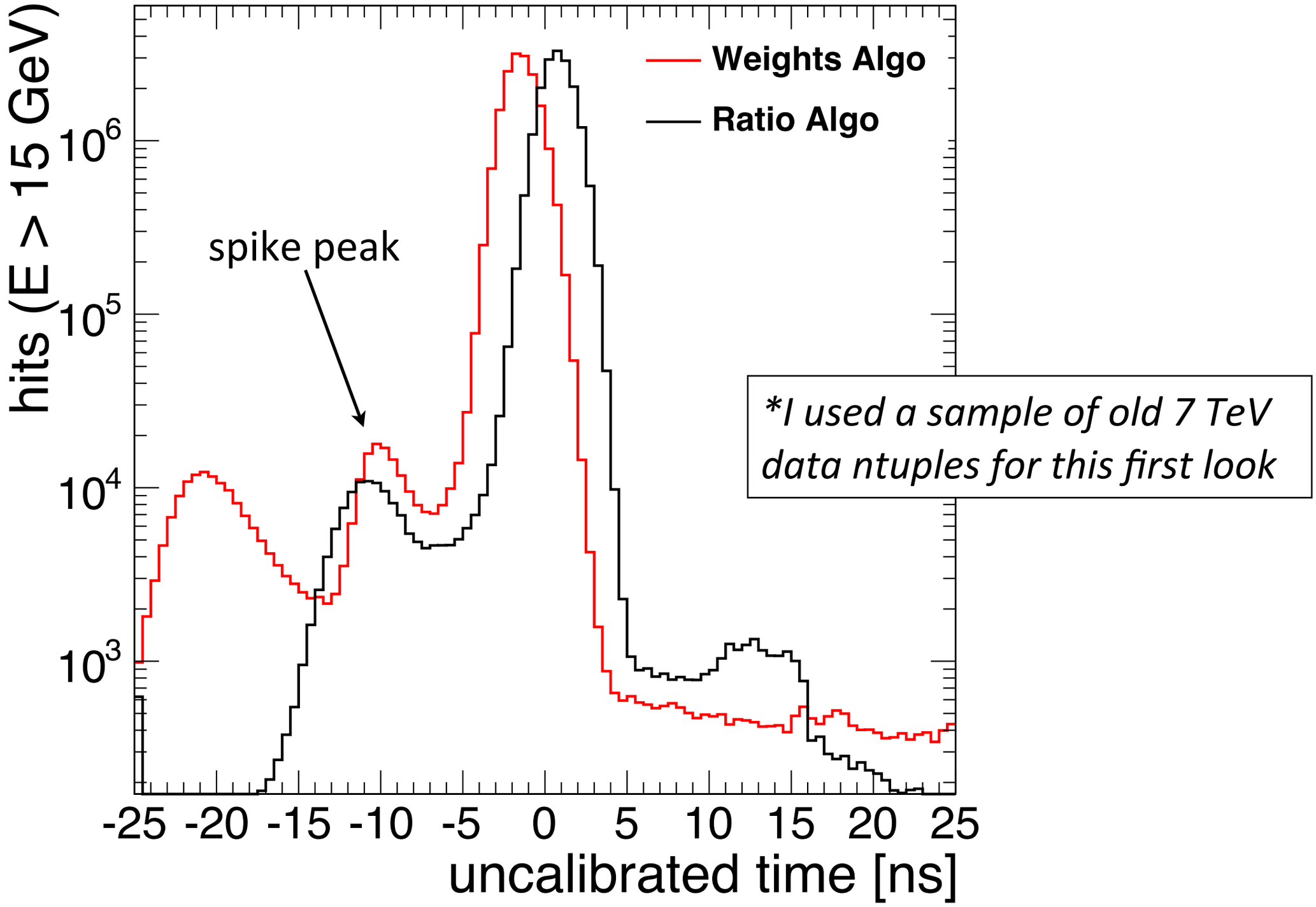
- float amp_weights_EB_MC[] = {-0.3828459, -0.3828459, -0.3828459, 0, 0.1846295, 0.424762, 0.3493013, 0.1768834, 0.01296132, 0};
- float time_weights_EB_MC[] = {0.04066309, 0.04066309, 0.04066309, 0, 1.325176, -0.04997078, -0.504338, -0.5024844, -0.3903718, 0};

- float amp_weights_EE_MC[] = {-0.3820672, -0.3820672, -0.3820672, 0, 0.2047881, 0.4153016, 0.3395061,0.1720173, 0.01458851, 0};
- float time_weights_EE_MC[] = {0.0579372, 0.0579372, 0.0579372, 0, 2.054817, 0.01276306, -0.7870374, -0.8303249, -0.6240293, 0};

- float amp_weights_EB_data[] = {-0.3837665, -0.3837665, -0.3837665, 0, 0.1549371, 0.4275022, 0.3594514, 0.1878533, 0.02155539, 0};
- float time_weights_EB_data[] = {0.002620844, 0.002620844, 0.002620844, 0, 1.21463, -0.03812831, -0.4031324, -0.4313319, -0.3499002, 0};

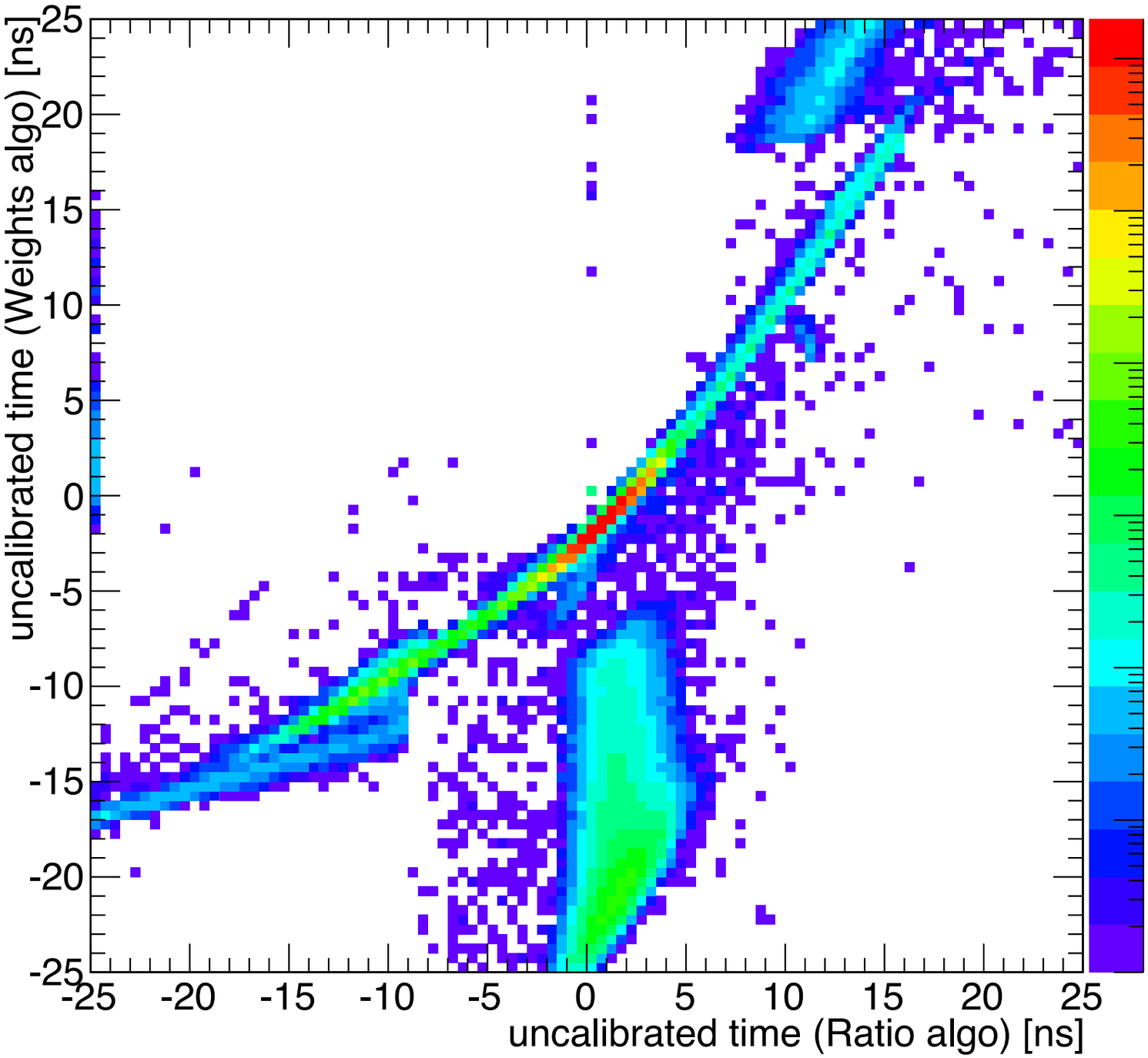
- float amp_weights_EE_data[] = {-0.3835888, -0.3835888, -0.3835888, 0, 0.1646038, 0.4132025, 0.354657, 0.1895535, 0.02874949, 0};
- float time_weights_EE_data[] = {0.0119698, 0.0119698, 0.0119698, 0, 1.883419, 0.06886431, -0.6895168, -0.7495467, -0.5491289, 0};

First look in CMS data*



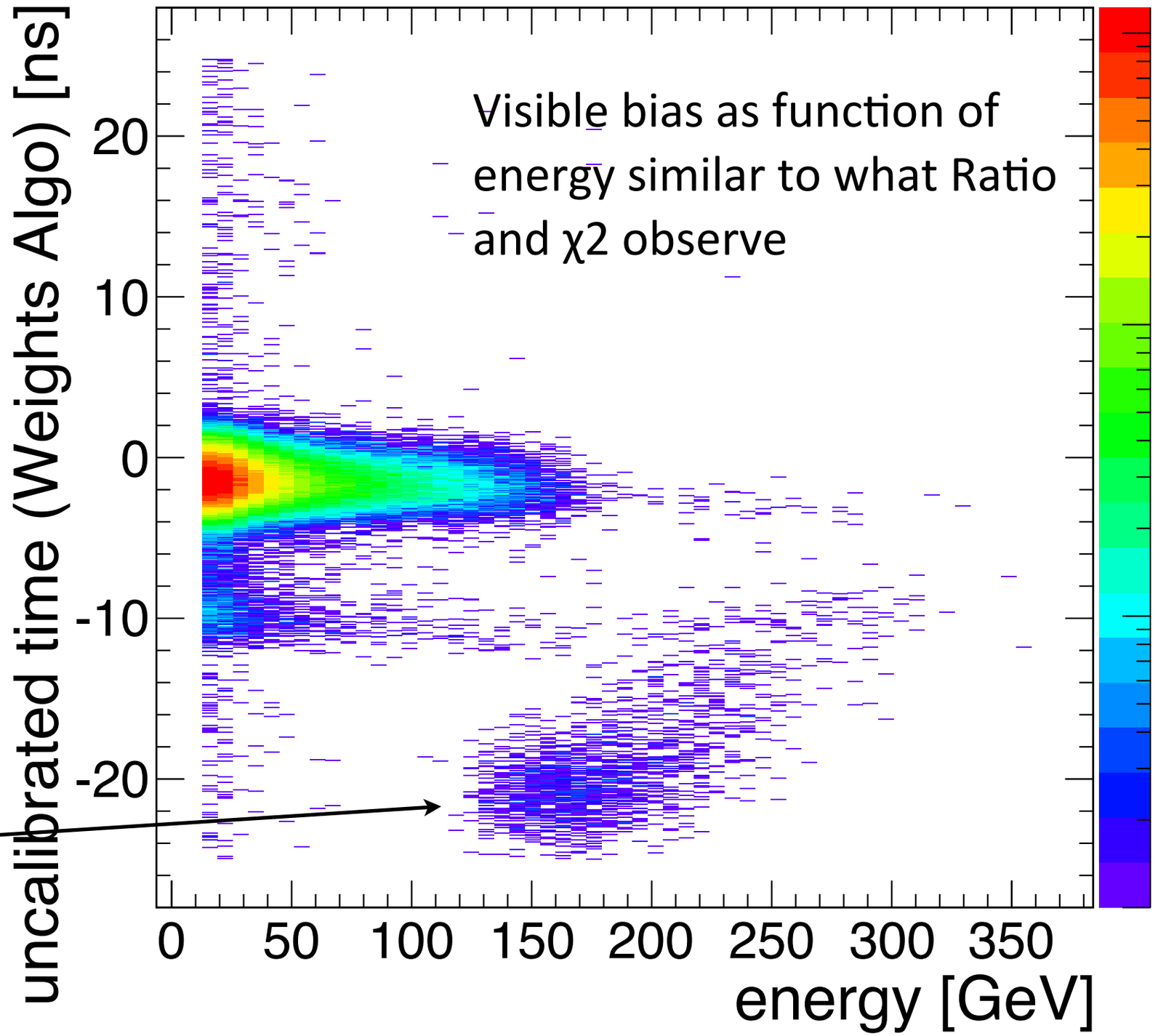
**I used a sample of old 7 TeV data ntuples for this first look*

Weights vs Ratio



Linear behavior for small δt

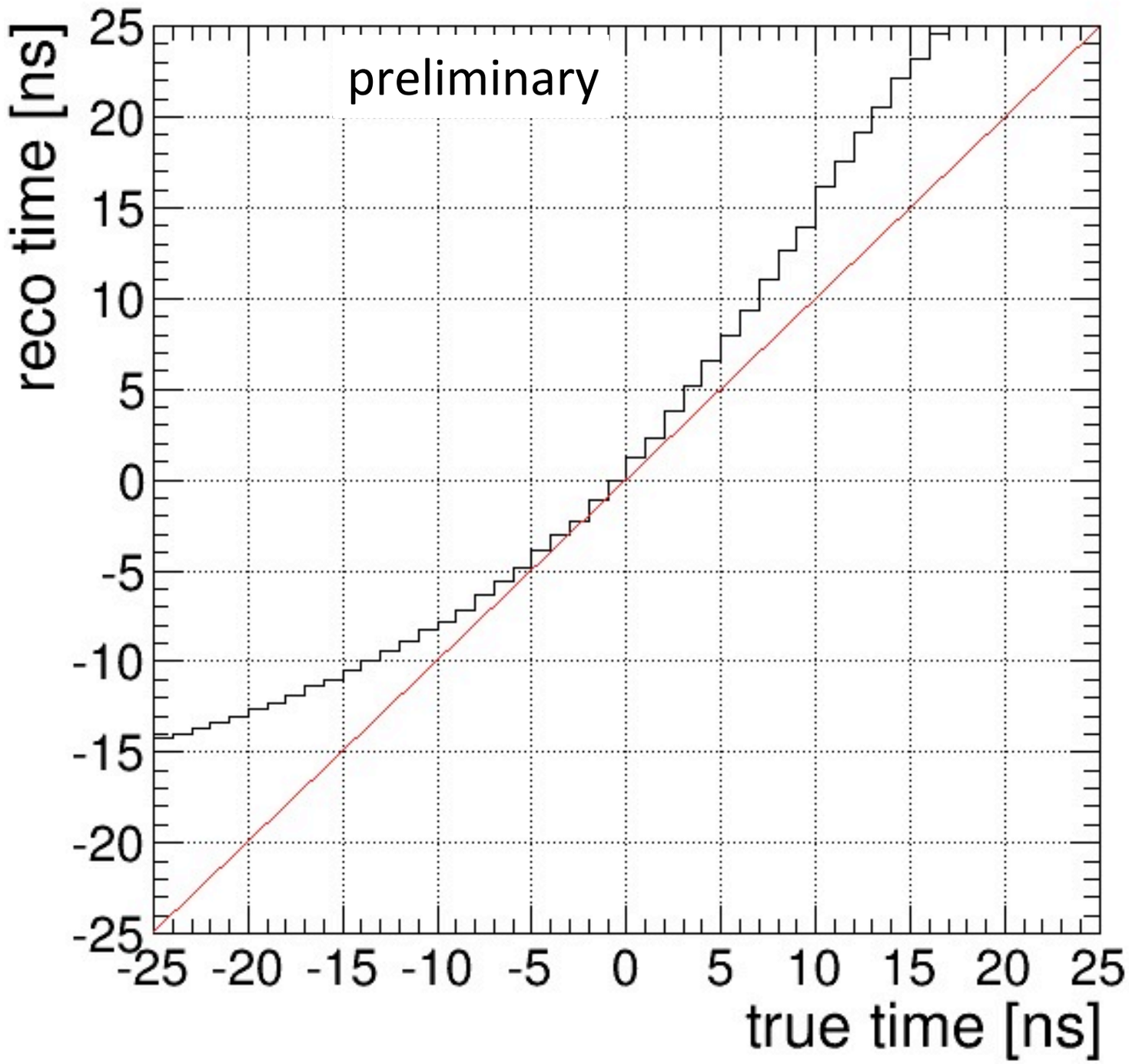
Time vs Energy



Blob should be due to a calibration run (probably laser) data are old 7 TeV and w/o any cleaning, but OK for the purpose of this report

Reco vs True time in MC

True time is the time offset injected in the CMSSW ECAL pulse that was reco'ed with the nominal weights (expecting δt=0)



Summary

- Reconstructing ECAL time using weights is **fast and simple** (just 1 line of code *once weights are known*)
  - **for (int i=0 ; i<10 ; i++) δt = -t[i]*S[i]/A;**

- A first attempt to reconstruct ECAL time with weights seems promising, performance is unknown though … should be studied