

# HMS Calorimeter Calibration

V.Tadevosyan

**Hall C 12 GeV Software Meeting, 6/23/2014**

# Algorithm

## Definitions:

$q = \begin{pmatrix} q_1 \\ \vdots \\ q_N \end{pmatrix}$  -- QDC signals,  $\alpha = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{pmatrix}$  -- calibration constants,

$e_0 = E(e)$  -- mean energy,  $q_0 = E(q)$  -- mean signals,

$e_R = \alpha^T q$  -- reconstructed energy.

## Constrained minimization:

Minimize  $E(e_R - e)^2$  **subject to**  $\alpha^T q_0 = e_0$

## Solution:

$$\alpha_C = \frac{e_0 - \alpha_U^T q_0}{q_0^T Q^{-1} q_0} Q^{-1} q_0 + \alpha_U$$

$$Q = E(qq^T), \alpha_U = Q^{-1} q_e, q_e = E(eq)$$

**Ts.Amatuni, On the calibration of segmented full absorption calorimeters. Unpublished.**

# Scripts

A two-step process under Root/hcana:

- 1) `hcal_replay.cpp`(<RunNumber>, <NumberOfEvents>) – replays a run, generates ntuples with raw data.
- 2) `hcal_calib.cpp`+(<RunNumber>) – calibrates the replayed run.

- Works on single track electron events.
- Selects electrons in gas Cherenkov.
- Also makes use of  $\beta_{TOF}$  for  $e^-$  selection.

# hcana Shower code

In THcShower.cxx, add ntuples related to the 1-st track for the calibration of the HMS calorimeter.

In **THcShower:DefineVariables**:

```
// Additional quantities for calibration. Revise later on.
```

```
RVarDef cvars[] = {  
  { "trdelta", "Track momentum deviation, %", "fTRDeltaP" },  
  { "trbeta", "Track beta from scint's", "fTRBeta" },  
  { "trp", "Track momentum", "fTRP" },  
  { "trxp", "Track x-slope", "fTRXp" },  
  { "tryp", "Track y-slope", "fTRYp" },  
  { 0 }  
};  
return DefineVarsFromList( cvars, mode );
```

*Will be removed when the list of the HMS ntuples settled.*

# hcana Shower code

In `THcShower::FineProcess`:

```
// Additional quantities for calibration, taken from the 1-st track currently.  
  
Int_t Ntracks = tracks.GetLast()+1; // Number of reconstructed tracks  
  
if(Ntracks > 0) {  
    THaTrack* theTrack = static_cast<THaTrack*>( tracks[0] );  
    fTRDeltaP = theTrack->GetDp();  
    fTRBeta   = theTrack->GetBeta();  
    fTRP      = theTrack->GetP();  
    fTRXp     = theTrack->GetTheta();  
    fTRYp     = theTrack->GetPhi();  
};
```

# Replay script

`hcal_replay.cpp`, a replica of `hodtest_mkj.C`, albeit with minor modifications.

Event selection done in `hcal_replay_cuts.def`:

Block: Reconstruct

```
one_track H.dc.ntrack==1
#one_clust H.cal.nclust==1
one_sh_track H.cal.ntracks==1
in_delta H.cal.trdelta>-10.&&H.cal.trdelta<10.
good_cer H.cer.npesum>3.
good_beta H.cal.trbeta>0.9&&H.cal.trbeta<1.1
in_calx H.cal.trx>-60.&&H.cal.trx<60.
in_caly H.cal.try>-30.&&H.cal.try<30.
in_cal in_calx&&in_caly
```

# This version matches engine

```
Reconstruct_master one_track && one_sh_track && in_delta && good_cer &&
good_beta && in_cal
```

*Good\_beta is omitted for now.*

*The names of variables in the cut definitions will be changed when list of the HMS ntuples is finalized.*

# Replay script

Ntuple selection done in `output_hcal_replay.def`:

```
#  
# Output definition for the HMS calorimeter calibration.  
#  
  
block H.cal.*           ← comprises pedestal subtracted ADC signals  
variable H.cer.npesum  
variable H.dc.ntrack
```

# Calibration script

In `hcal_calib.cpp`:

```
...  
  
#include "THcShowerCalib.h"           ← the Shower calibration class  
  
...  
  
THcShowerCalib theShowerCalib(RunNumber);  
  
theShowerCalib.Init();                // Initialize constants and variables  
theShowerCalib.CalcThresholds();      // Thresholds on the uncalibrated Edep/P  
theShowerCalib.ComposeVMs();          // Compute vectors and matrices  
theShowerCalib.SolveAlphas();         // Solve for the calibration constants  
theShowerCalib.SaveAlphas();          // Save the constants  
// theShowerCalib.SaveRawData();      // Save raw data into file for debug purposes  
theShowerCalib.FillHEcal();           // Fill histograms
```



# Calibration output

Outputs calibration constants in `hcal.param.<RunNumber>`, in format compatible with engine:

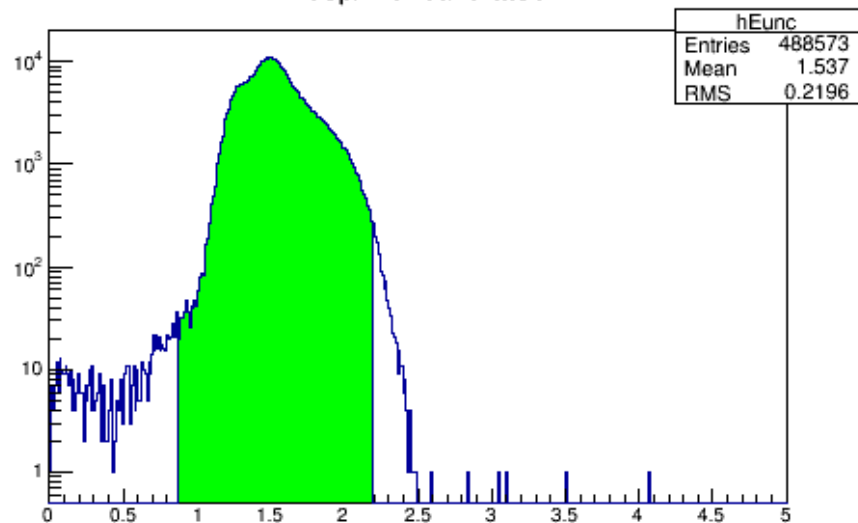
```
; Calibration constants for run 52949, 486190 events processed
```

```
hcal_pos_gain_cor=-0.035, 0.330, 0.417, 0.385, 0.247, 0.507, 0.555, 0.403, 0.521, 0.305, 0.420, 0.392, 0.936,  
-0.102, 0.456, 0.347, 0.388, 0.449, 0.484, 0.228, 0.443, 0.313, 0.290, 0.395, 0.675, 0.590,  
0.611, 0.624, 0.601, 0.850, 0.627, 0.655, 0.463, 0.544, 0.719, 0.718, 0.525, 0.628, 0.808,  
-0.126, 0.610, 0.579, 0.727, 0.728, 0.822, 0.847, 0.737, 0.790, 0.627, 0.692, 0.692, 0.393,  
hcal_neg_gain_cor= 0.772, 0.289, 0.286, 0.365, 0.352, 0.052, 0.178, 0.416, 0.214, 0.388, 0.145, 0.175,-0.546,  
0.855, 0.323, 0.378, 0.506, 0.328, 0.334, 0.449, 0.305, 0.457, 0.350, 0.272, 0.349,-0.351,  
0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000,  
0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000,
```

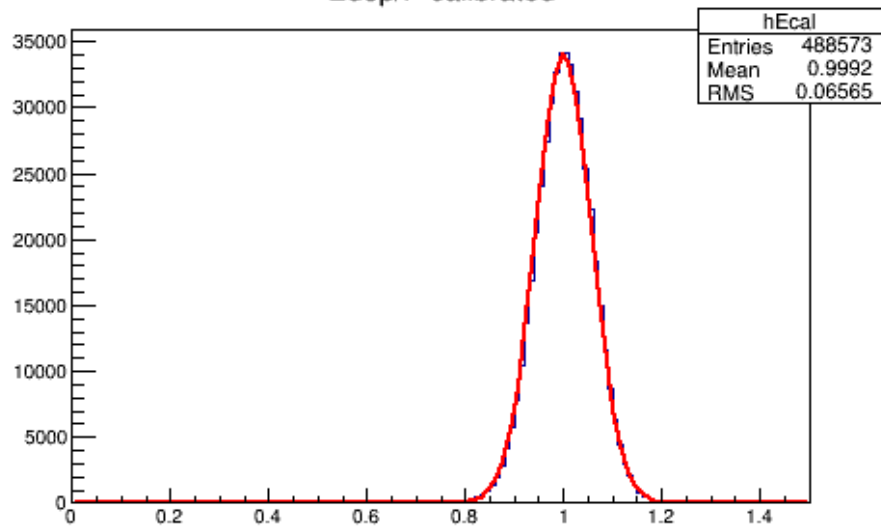
*Hcal\_calib.cpp was checked against engine, produces same constants on the same track set!*

# Graphics Output

Edep/P uncalibrated



Edep/P calibrated



$\Delta P$  versus Edep/P

