HCAL Reconstruction:
MC Correction Functions Update

E. Berry[1]

[1]Brown University

Friday, May 2, 2014

BROWN

Introduction

- Have derived (improved) MC correction functions for OOT PU
- Same derivation method as used for data
- Procedure:
    - Run Alexandre's ratio method on zero PU MC
    - Derive correction functions based on the pulse shape
    - Use the same definitions, fits, and methods as in data
    - Validate results on MC with OOT PU

Method

- Process a high-p_T QCD sample in two ways:
    - No pileup: for MC truth comparison (DONE)
    - With pileup: for validation (Processing)
- Compare results event-by-event, channel-by-channel:
    - No pileup
    - vs. with pileup and no corrections
    - vs. with pileup and corrections

Datasets

- Consider two `GEN-SIM` datasets (no PU) at `T1_US_FNAL`:

Dataset	Production release
/MinBias_TuneZ2star_13TeV-pythia6/Summer13-START53_V7C-v1/GEN-SIM	CMSSW_5_3_10_patch2
/QCD_Pt-1800_TuneZ2star_13TeV_pythia6/Fall13-POSTLS162_V1-v1/GEN-SIM	CMSSW_6_2_0_patch1

- `QCD_Pt-1800` dataset:
  - DAS link
  - 93453 ($\sim$ 100k) events, 95 files
  - HcalNoiseAnalyzer ntuples on FNAL EOS:
   `/eos/uscms/store/user/eberry/QCD1800MC/`
- `MinBias` dataset:
  - DAS link
  - 9999424 ($\sim$ 10M) events, 946 files
  - HcalNoiseAnalyzer ntuples on FNAL EOS:
    `/eos/uscms/store/user/eberry/MinBiasMC/`

Processing pileup sample

- Need to overlay `QCD` with `MinBias`
- Use `MixingModule` in `CMSSW_6_2_8`
- Pileup scenario: `AVE_50_BX_25ns`
- Two stages:
    - 1) `DIGI, L1, DIGI2RAW, HLT`
    - 2) `RAW2DIGI L1Reco RECO`
- Stage 1 all done: cmsDriver and python cfg
- Stage 2 part done: cmsDriver and python cfg
- High PU is VERY CPU intensive: 2 minutes/event

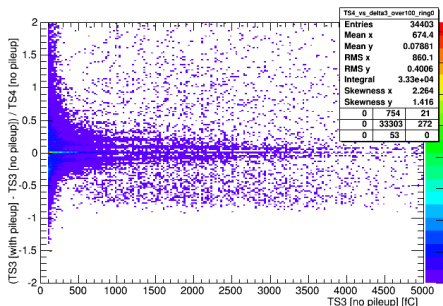# PU vs. No PU single digi comparison

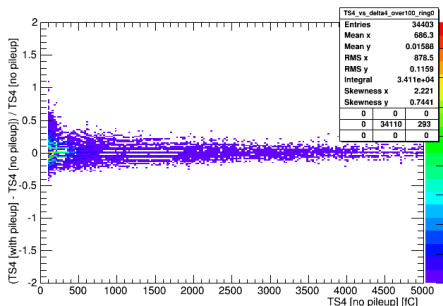single DIGI comparison: HB
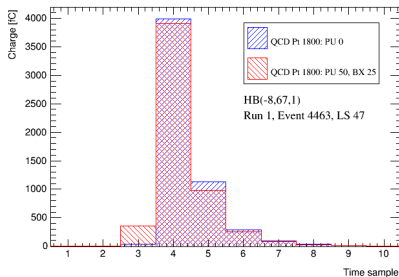
single DIGI comparison: HE
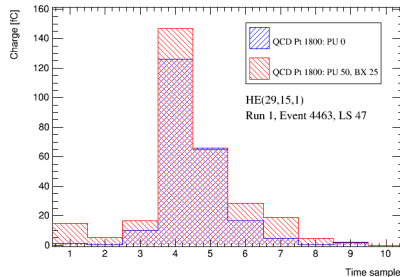
- HE as expected.
- HB as expected in TS3. Strangeness in TS4 + TS5.
- Bug in `MixingModule`? Investigating with M. Hildreth.

PU vs. No PU broad comp.: TS3 in HB

- *x*-axis: TS3 no PU
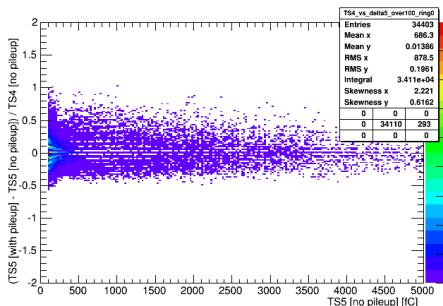- *y*-axis: (TS3 with PU - TS3 no PU) / TS3 no PU

PU vs. No PU broad comp.: TS4 in HB
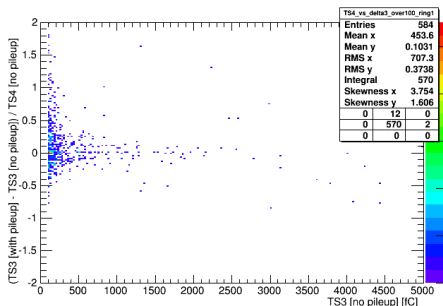

- x-axis: TS4 no PU
- y-axis: (TS4 with PU - TS4 no PU) / TS4 no PU
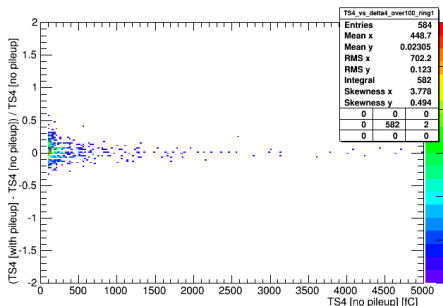
PU vs. No PU broad comp.: TS5 in HB

- x-axis: TS5 no PU
- y-axis: (TS5 with PU - TS5 no PU) / TS5 no PU

PU vs. No PU broad comp.: TS3 in HE 17:20

- *x*-axis: TS3 no PU
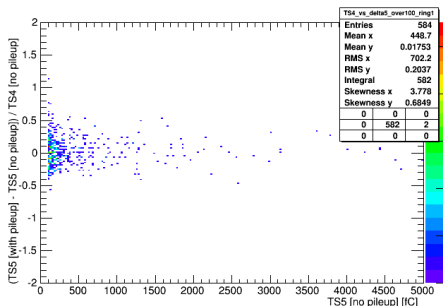- *y*-axis: (TS3 with PU - TS3 no PU) / TS3 no PU

PU vs. No PU broad comp.: TS4 in HE 17:20

- *x*-axis: TS4 no PU
- *y*-axis: (TS4 with PU - TS4 no PU) / TS4 no PU

PU vs. No PU broad comp.: TS5 in HE 17:20

- *x*-axis: TS5 no pileup
- *y*-axis: (TS5 with PU - TS5 no pileup) / TS5 no pileup
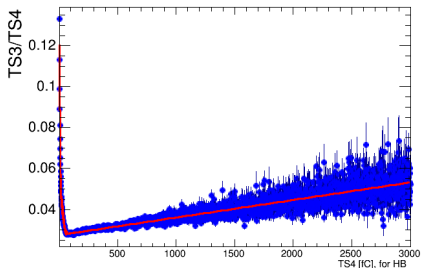
Pileup conclusion

- It is strange that the PU digis are sometimes lower than no-PU digis
- M. Hildreth has seen the single DIGI plots and code
- More investigation required
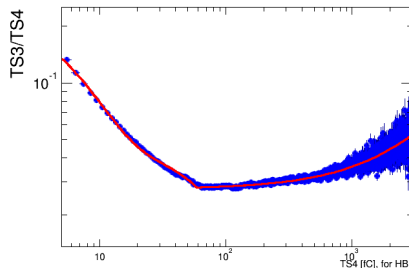- Have some pointers on where to look from M. Hildreth

Fits

- Fits have been improved! Better agreement now.
- Parameters available on GitHub
- Same functions as Alexandre for a1, a2, a3
    - 6 polynomials: 1 for each of 6 regions
- For a_1, this function works better on MC:

  if $x <$ [6]: $f(x) = [0] \cdot \mathrm{Exp}([1] + [2] \cdot x) + [3] + [4] \cdot x$

  if $x >$ [6]: $f(x) = [5] \cdot (x - [6]) + c$

    - c is chosen to ensure continuity of $f(x)$ at [6]

Introduction   Validation procedure   **Validation results**   Conclusion
○   ○○○   ○○○○○○○○○○●○○○○○○○○○○   ○
Fitting and sanity-checking function on low-PU sample

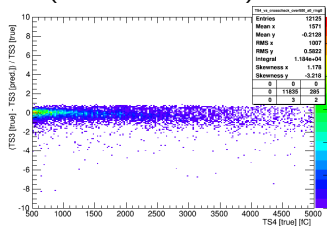# Function fitting on zero pileup sample: a_1

Fit of a_1 in HB (lin)

Fit of a_1 in HB (log)

- Fit done on zero pileup sample
- Fits now extend to TS4 = 3000 fC
- Parameters available on GitHub

Function validation on zero pileup sample: a_1

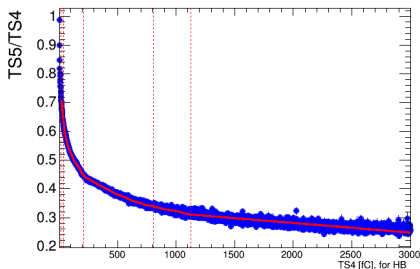Validation of a_1 in HB
(TS4 > 0 fC)

Validation of a_1 in HB
(TS4 > 500 fC)

- Done on zero pileup sample
- *y*-axis: (TS3 true - TS3 pred.) / TS3 true
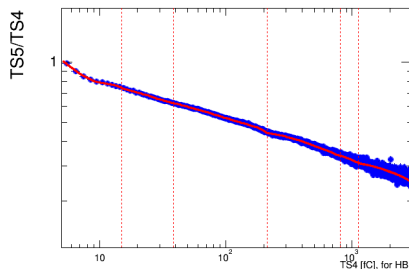- *x*-axis: TS4 true
- Spread all at low energy

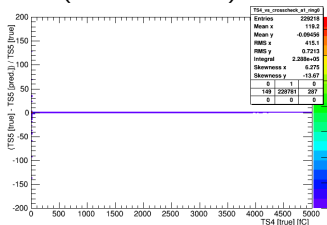# Function fitting on zero pileup sample: a1

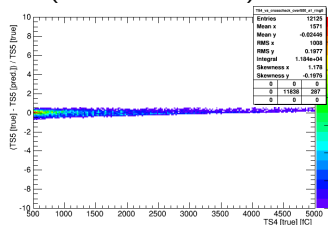Fit of a1 in HB (lin)   Fit of a1 in HB (log)

- Fit done on zero pileup sample
- Red lines correspond to fit ranges (Alexandre's functions)
- Parameters available on GitHub

Function validation on zero pileup sample: a1
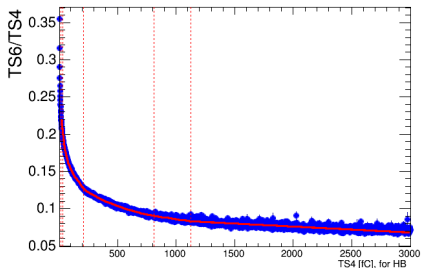
Validation of a1 in HB
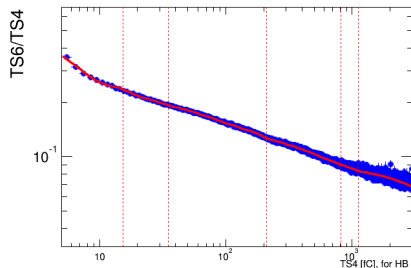(TS4 > 0 fC)

Validation of a1 in HB
(TS4 > 500 fC)

- Done on zero pileup sample
- *y*-axis: (TS5 true - TS5 pred.) / TS5 true
- *x*-axis: TS4 true
- Better performance than a_1

Function fitting on zero pileup sample: a2

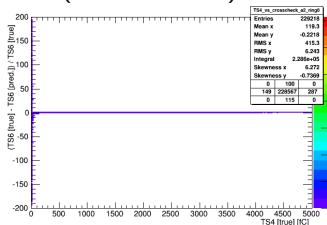Fit of a2 in HB (lin)   Fit of a2 in HB (log)

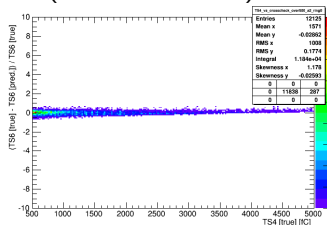
- Fit done on zero pileup sample
- Red lines correspond to fit ranges (Alexandre's functions)
- Parameters available on GitHub

Function validation on zero pileup sample: a2
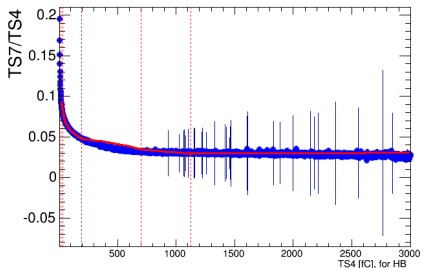
Validation of a2 in HB
(TS4 > 0 fC)

Validation of a2 in HB
(TS4 > 500 fC)

- Done on zero pileup sample
- y-axis: (TS6 true - TS6 pred.) / TS6 true
- x-axis: TS4 true
- Better performance than a_1

Function fitting on zero pileup sample: a3

Fit of a3 in HB (lin)

Fit of a3 in HB (log)

- Fit done on zero pileup sample
- Red lines correspond to fit ranges (Alexandre's functions)
- Parameters available on GitHub

Function validation on zero pileup sample: a3

Validation of a3 in HB
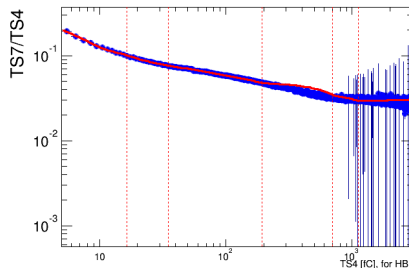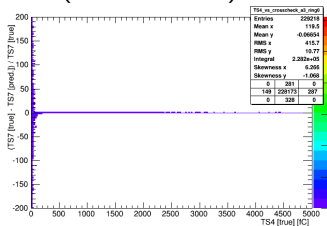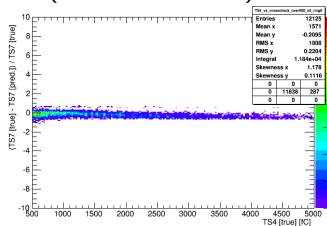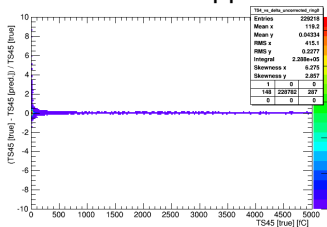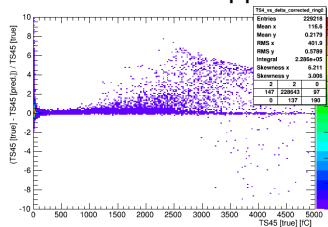(TS4 > 0 fC)

Validation of a3 in HB
(TS4 > 500 fC)

- Done on zero pileup sample
- y-axis: (TS7 true - TS7 pred.) / TS7 true
- x-axis: TS4 true
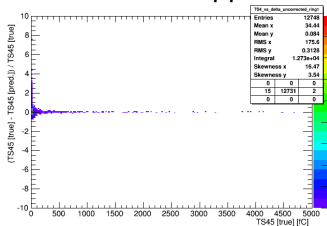- Better performance than a_1

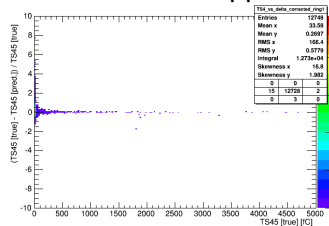# Results in HB

No correction applied

With correction applied

- Done on high pileup sample
- *y*-axis: (TS45 true - TS45 pred.) / TS45 true
- *x*-axis: TS45 true
- Large disc. for TS45 > 1000 fC (where fits are good)

Introduction   Validation procedure   **Validation results**   Conclusion
○   ○○○   ○○○○○○○○○○○○○○○○○○●   ○
Validating on high-PU sample

Results in HE: 17:20

No correction applied

With correction applied

- Done on high pileup sample
- *y*-axis: (TS45 true - TS45 pred.) / TS45 true
- *x*-axis: TS45 true
- Need more statistics to see if problems from HB persist

Conclusion

- Processed zero-PU samples: OK for shape studies
- Processed high-PU samples: OK for validation
- Fit functions ready to go using Alexandre's method:
  - Improved over fit functions from earlier talks
  - Fit functions model the zero-PU pulse shapes well
  - Fit functions do NOT predict the high-PU pulses well
- But the method does not pass basic sanity checks
- Separate: strange features observed after MixingModule
  - MC with PU often has less charge than MC without PU
  - Investigating with pointers from M. Hildreth
- Suggestion: look at MC similar to Alexandre's data
  - Working on this