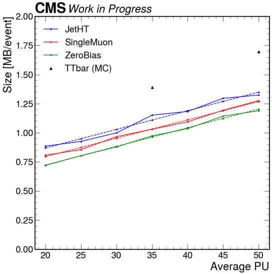
RAW' for pp collisions in CMS Run 3

Alessandro Brusamolino
Università degli Studi di Milano-Bicocca

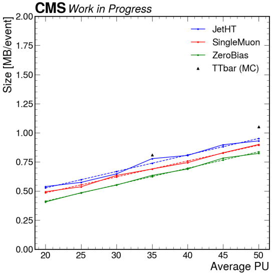
SIF 2022 - September 13th, 2022

Introduction

- **RAW data format**: full event information, contains raw detectors information;
- the trend of RAW size vs $\langle PU \rangle$[1] in 3 different datasets, which have different particle multiplicity ($t\bar{t}$ shown for reference)



uncompressed RAW data



compressed RAW data

		RAW data size [MB/event]	
	PU	uncompressed	compressed
Run 2	35	0.8-1.0	0.5-0.7
Run 3	50	1.2-1.3	0.8-0.9
	70	1.5-1.6	1.1-1.2

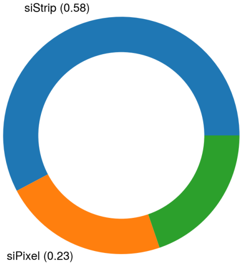
- possible strategy to face higher pileup conditions during pp collisions in Run 3 $\Longrightarrow$ data size reduction in order to save tape and disk space and increase HLT[2] rate

[1] Number of simlutaneous collisions. The PU measurement was made using 72 mb as cross section

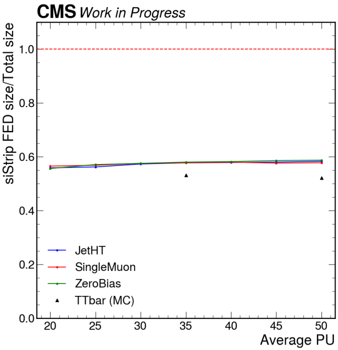
[2] High Level Trigger; it is the second level of the CMS trigger system

RAW format - current status

- SiStrips readout system comprises 10^7 channels, 437 FEDs included in the DAQ
- SiStrips FEDs represent $\sim$ 60% ($\sim$ 50%) of compressed (uncompressed) RAW data size; the second contributor is the SiPixel detector (with $\sim$25%)
- this contribution is almost stable vs $\langle PU \rangle$ and across the different datasets



Composition of a compressed RAW file in a mulit-jet event (JetHT dataset) event at PU=35



RAW format - redefinition of the RAW data format

- compressed (uncompressed) RAW size is 0.5-0.7 (0.8-1.0) MB/event
- HLT average rate $\sim 1 - 1.5$ kHz

$\Longrightarrow$ $\sim$ 0.5-0.7 (0.8-1.0) GB/s (@35 PU) stored @ T0 and T1s for pp collisions

$\Longrightarrow$ we can save tape and disk space by redefining its content

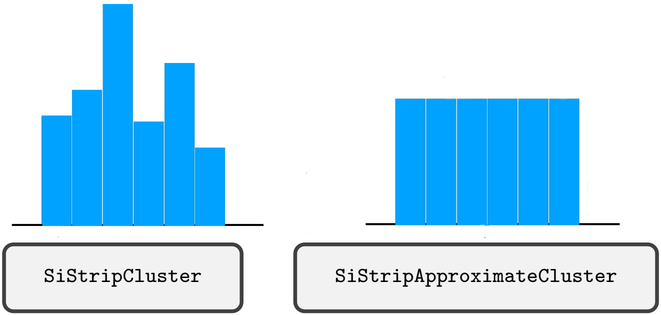
- use pre-processed information (exploiting objects reconstructed at the HLT)
- focus on biggest contributor $\rightarrow$ SiStrips digis
  $\Longrightarrow$ use approximated `siStripClusters` and store them in RAW' data format (but need to check physics performances)
  $\Longrightarrow$ transfer more data keeping the same bandwidth (loss in physics performance could be compensated by statistics)

Reduced RAW size also allows us to increase HLT rate

- this new approach will be used during the heavy ions collisions operation by the CMS experiment in Run 3

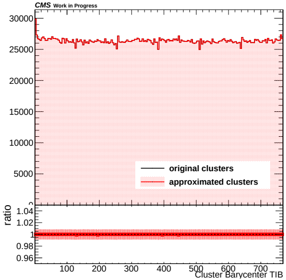
Approximated `SiStripClusters` - approach

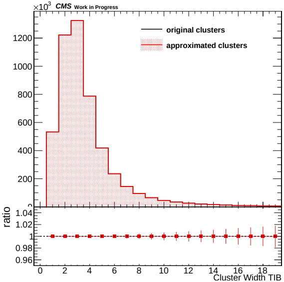
- **now**: SiStripClusters are produced by the strips local reconstruction and represent a group of neighbouring strips passing certain criteria about their amplitude in ADC counts
  - not stored in RAW files
- **new SiStripClusters**: shape approximated as rectangles with the same width as the original ones and charge defined as the average charge
- stored cluster quantities:
  - barycenter (`float`): cluster barycenter
  - width (`uint8`) : cluster size
  - average charge (`float`): total cluster charge divided by the number of strips



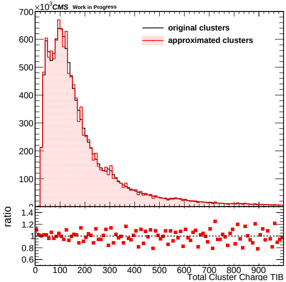
Approximated clusters - stored quantities distribution

The following distributions are referred to clusters in the TIB [3] (tracker inner barrel) sub-detector



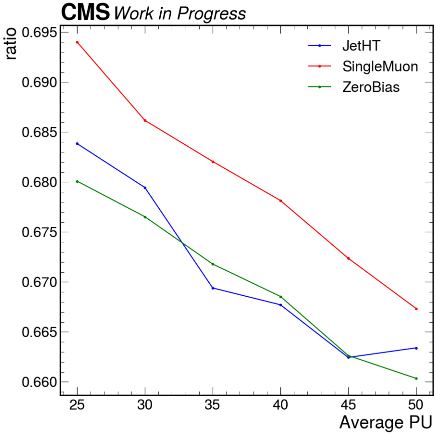


- cluster multiplicity unchanged
- barycenter and width distributions unchanged since we use the original quantities
- charge distribution is the only one affected by the approximation, but it is not significantly different (it is the mean of the original values * cluster size)



[3] 6 APVs per module; each APV has 128 strips

Approximated clusters - sizes

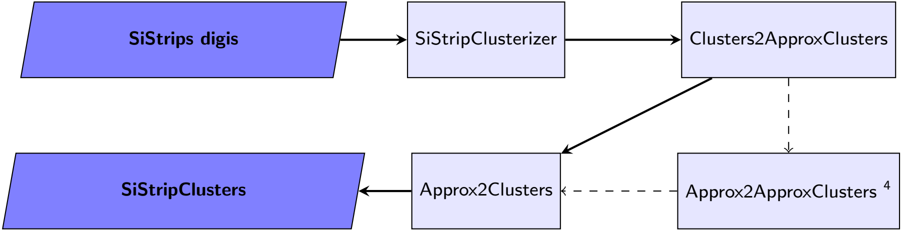


Ratio between approximated cluster size and SiStrip FEDs size

- we made a first measurement of the size reduction
  $\rightarrow$ **20-25% reduction** of the compressed RAW event in pp collisions at 35 PU
  $\implies$ + **400-500 Hz in HLT rate (25-30% increase)**
- studies performed on heavy ions collisions data show a reduction of $\sim 30 - 40\%$ (compressed event)

Approximated clusters - proposed workflow

- workflow for testing this new approach running the track reconstruction
- handle the local reco in steps
  - strip clusters → strip approx clusters (this should be done either at HLT step at P5 or in the packing step at T0)
  - strip approx clusters → strip clusters (this should be done at RECO step)



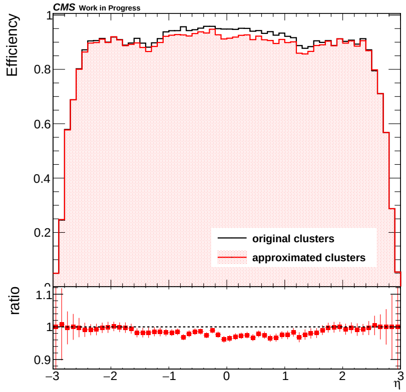
[4] different approximation versions that allow different resolutions of the stored quantities

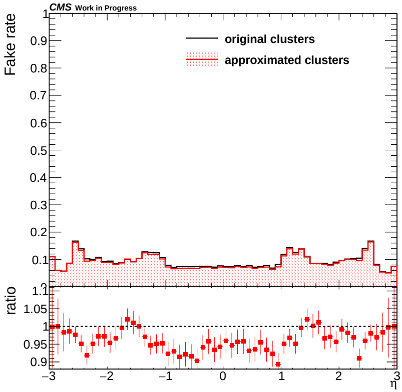
Approximated clusters - impact on tracking

- track reconstruction performances [5] given in terms of efficiency and fake rate

$$\epsilon = \frac{N_{\text{associated}}}{N_{\text{simulated particles}}} \qquad \text{fake rate} = \frac{N_{\text{not associated}}}{N_{\text{reconstructed tracks}}}$$

$\Longrightarrow$ efficiency [6] and fake rate almost unchanged wrt CMS RAW format

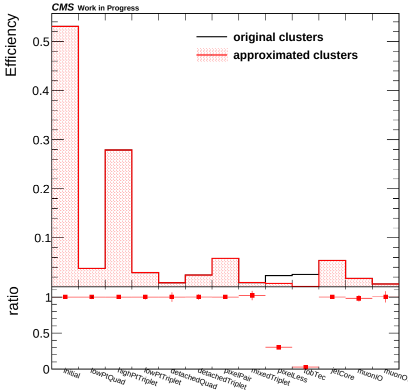


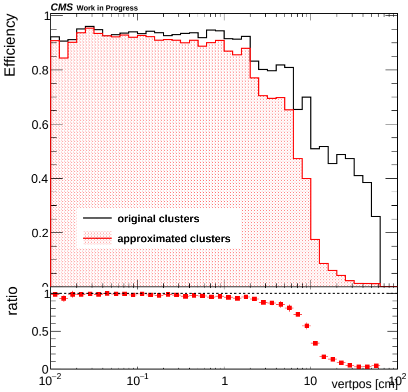


[5] Tests performed with a $t\bar{t}$ MC sample at 35 PU

[6] Here it is shown for simulated particles with $p_T > 0.9$ GeV and $|d_{xy}| < 2.5$ cm

Approximated clusters - outer tracking





- **vertpos** $\equiv$ track production point position wrt collision point
- inefficiencies arise in:
  - `pixelLess`
  - `tobTec`

  seeding step makes use of strips only clusters
- seeding makes use of a **cluster shape filter** in order to limit the fake rate raising from spurious hits and the large combinatorics

Conclusions

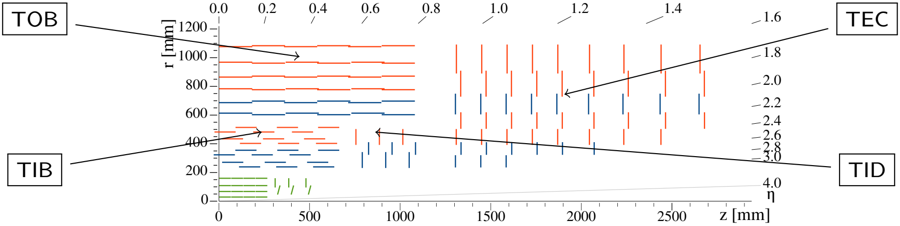
- with the increase of PU the RAW format increases
  $\rightarrow$ by reducing its size we can save both disk and tape space and increase the average HLT rate $\Longrightarrow$ more events for physics analyses!
- the most offending contributors are the digis of the SiStrip detector
  $\rightarrow$ simplifying the SiStrip cluster to a rectangular distribution we can
  - save up to 20-25% at 35 PU
  - keep the tracking performance almost unchanged

  $\rightarrow$ the cluster shape is used in the seeding of displaced tracks, and therefore the simple reduction algorithm used so far needs to be updated (particularly relevant for analyses looking at exotic signatures like in mono-poles searches)

BACKUP

CMS tracker

- pixels → 4 barrel layers + 3 endcap disks
  - hit position resolution: 10 μm ($r\phi$), 20-40 μm (z)
- strips → 10 barrel layers + 12 endcap disks
  - single and stereo modules
  - hit position resolution: 13-38 μm (TIB, TID), 18-47 μm (TOB, TEC)



since 2017

Tracking at CMS

- local reconstruction: detctor information, along with geometry and magnetic field informations, to create clusters → **hits**
- global reconstruction: uses hits to reconstruct **tracks**
  - seeding
    define starting trajectory parameters + uncertainties
  - pattern recognition (Kalman filter)
    addition of hits and update of the parameters + uncertainties
  - final fitting
    performed to obtain final track parameters at interaction point exploiting full trajectory information
  - quality flags

$\}$ Combinatorial Track Finder

Iteration	Step Name	Seeding	Target Track
0	Initial	pixel quadruplets	prompt, high p_T
1	LowPtQuad	pixel quadruplets	prompt, low p_T
2	HighPtTriplet	pixel triplets	prompt, high p_T recovery
3	LowPtTriplet	pixel triplets	prompt, low p_T
4	DetachedQuad	pixel quadruplets	from B hadron decay, $r \leq 5$ cm
5	DetachedTriplet	pixel triplets	from B hadron decay, $r \leq 10$ cm
6	MixedTriplet	pixel+strip triplets	displaced, $r \leq 7$ cm
7	PixelLess	inner strip pairs	displaced, $r \leq 25$ cm
8	TobTec	outer strip pairs	displaced, $r \leq 60$ cm
9	JetCore	pixel pairs in jets	high-p_T jets
10	Muon inside-out	muon-tagged tracks	muons
11	Muon outside-in	standalone muons	muons

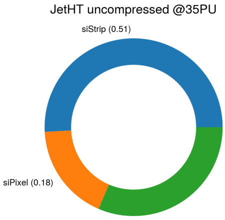
RAW format - strip amplitudes & local reconstruction

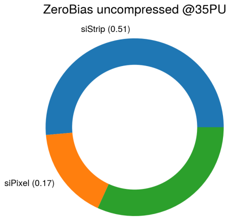
- SiStrip FEDs contain strip amplitudes passing zero suppression

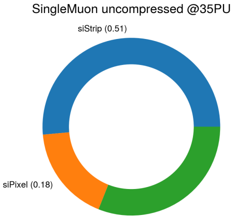
strip is accepted if
- → its charge is at least 5x channel noise
- → strip + neighbour strip have charge 2x channel noise

- amplitude for each strip is a `int` value
- clusters are seeded by channels passing zero-suppression with a charge $\geq$3x channel noise
- strips are added if if their charge is $\geq$2x channel noise
- clusters kept if their charge is at least 5x cluster noise

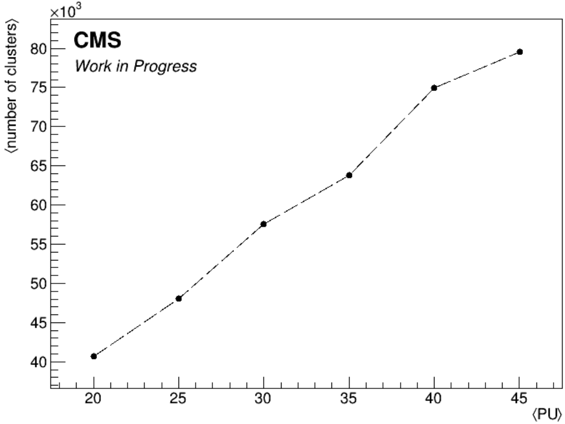
RAW format - current status







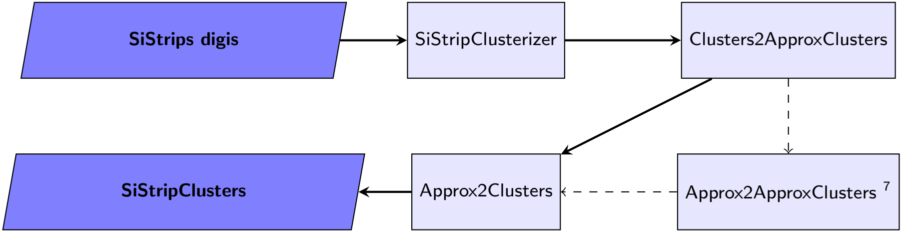
Clustering



×10^3
CMS
Work in Progress
⟨number of clusters⟩
80
75
70
65
60
55
50
45
40
20
25
30
35
40
45
⟨PU⟩

Approximated clusters - proposed workflow

- workflow for testing this new approach running the track reconstruction
- handle the local reco in steps
  - strip clusters → strip approx clusters (this should be done either at HLT step at P5 or in the packing step at T0)
  - strip approx clusters → strip clusters (this should be done at RECO step)



As for now, we only used `ORIGINAL` version

- `barycenter = std::round(barycenter);`
- `width = (width > 63) ? 63 : width;`
- `avgCharge = std::round(avgCharge);`

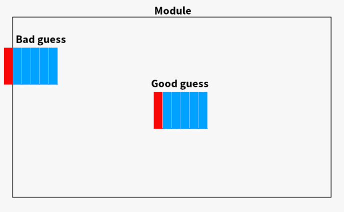
[7] there are different approxVersions that allow different resolutions of the stored quantities

Approximation versions

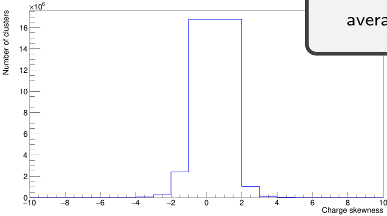
- ORIGINAL
  - `barycenter = std::round(barycenter);`
  - `width = (width > 63) ? 63 : width;`
  - `avgCharge = std::round(avgCharge);`
- FULL_WIDTH
  - `barycenter = std::round(barycenter);`
  - `width = width;`
  - `avgCharge = std::round(avgCharge);`
- BARY_RES_0.1
  - `barycenter = std::round(barycenter*10)/10;`
  - `width = (width > 63) ? 63 : width;`
  - `avgCharge = std::round(avgCharge);`
- BARY_CHARGE_RES_0.1
  - `barycenter = std::round(barycenter*10)/10;`
  - `width = (width > 63) ? 63 : width;`
  - `avgCharge = std::round(avgCharge*10)/10;`

Approximated clusters - bug in SiStripClusters

- a bug was found while looking into cluster shape filters
- variable firstStrip_ was badly initialized with a 0 default value → clusters were wrongly considered to be outside of the modules



- in order not to store additional variables:
  `firstStrip_ = cluster.barycenter() - cluster.width()/2`
  → this turns out to be a good approximation!



average skewness: 0.20

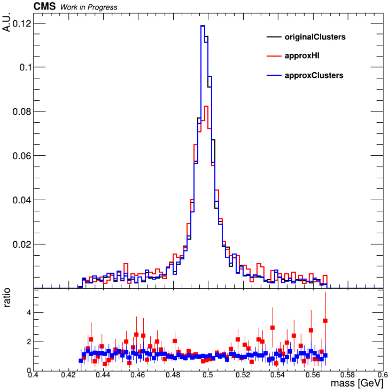
Approximated clusters - cluster shape filters

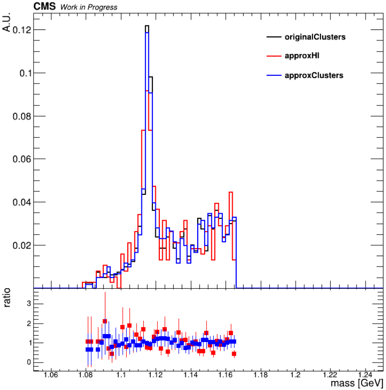
- Main selections are:
  - Saturated strips: if we have three consecutive saturated strips in a cluster (ADC count $>$ 254, link), this is discarded
    - one strip can have charge so high that the average cluster charge is above threshold
    - the three saturated strips can have a charge slightly higher than the threshold and the average cluster charge is below the threshold
  - Trimming: tails in the charge distribution of the cluster are removed if strip have less than 15-25% of the neighbouring strips charge . Having rectangular clusters, trimming may not work properly
    - dedicated studies are on-going:
- we are in contact with groups interested in HSCP, disappearing Tracks and monoPole that use dEdx

Current studies aimed at implementing some selection algorithms directly in the `SiStripApproximateCluster` constructor

Approximated clusters - V0 monitoring

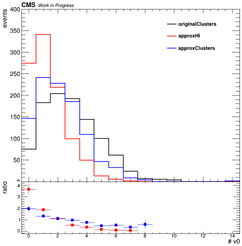
- check impact on K-short and Λ resonances

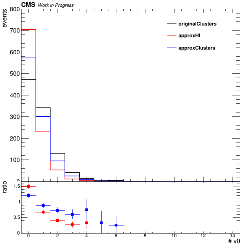


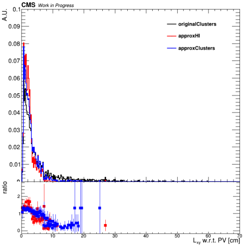


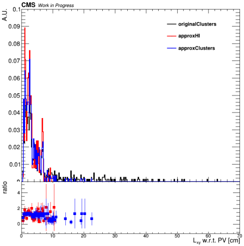
→ despite having less candidates, performances on V0 resonances look almost the same as the ones with standard clusters

Approximated clusters - V0 monitoring









K-shorts

Λ