Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

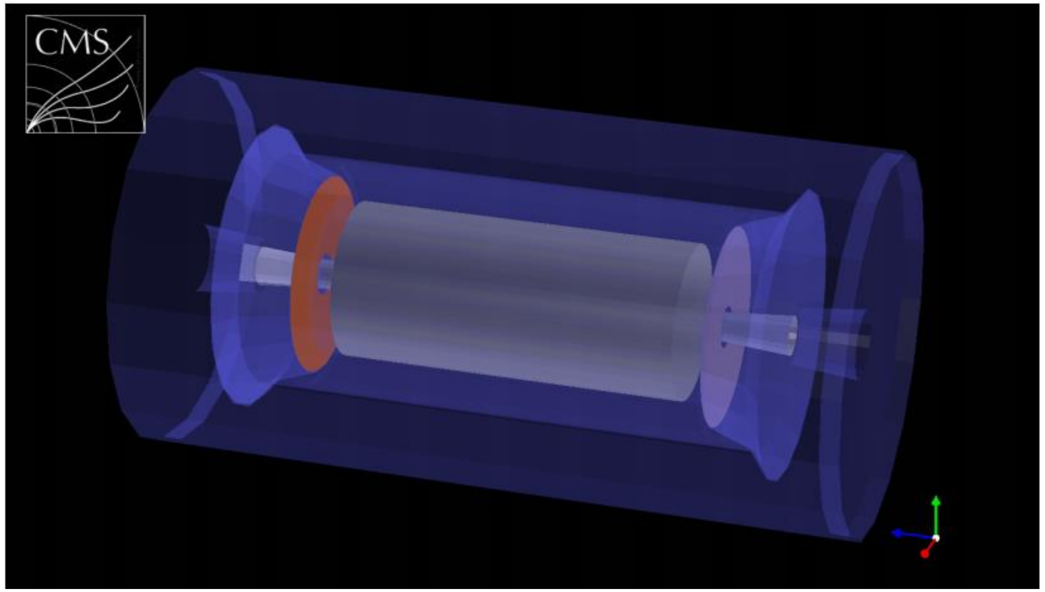
Track reconstruction with timing in CMS during HL-LHC

Ksenia de Leo, Fabio Cossutti (INFN Trieste) for the CMS Collaboration

Polarized Perspectives: Tagging and Learning in the SM –21/02/2025

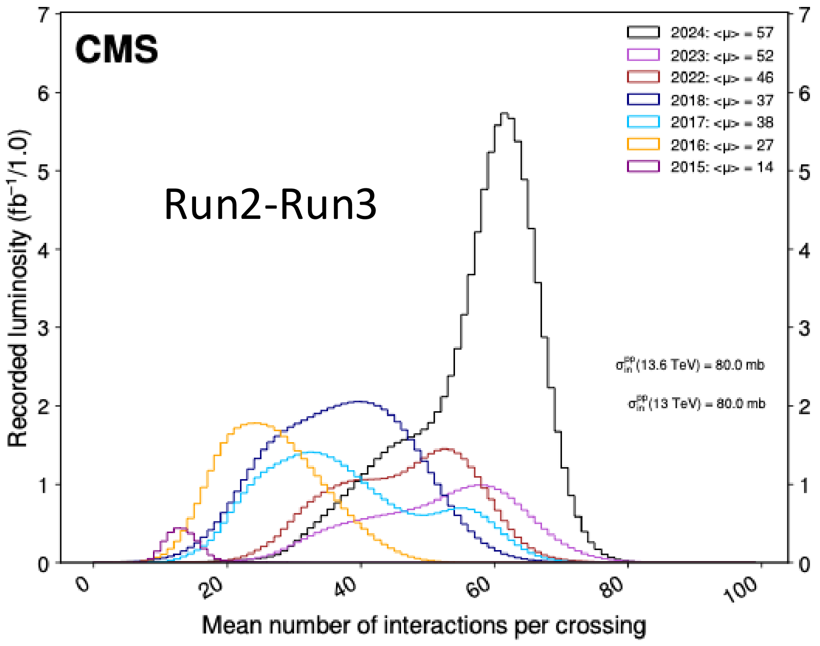
Outline

- HL-LHC
- Mip Timing Detector
- Tracking with timing
- 4D vertex reconstruction
- Update of the 4D algorithm
- Performace study of:
  - vertex time resolution
  - reconstruction efficiency
  - pileup rejection

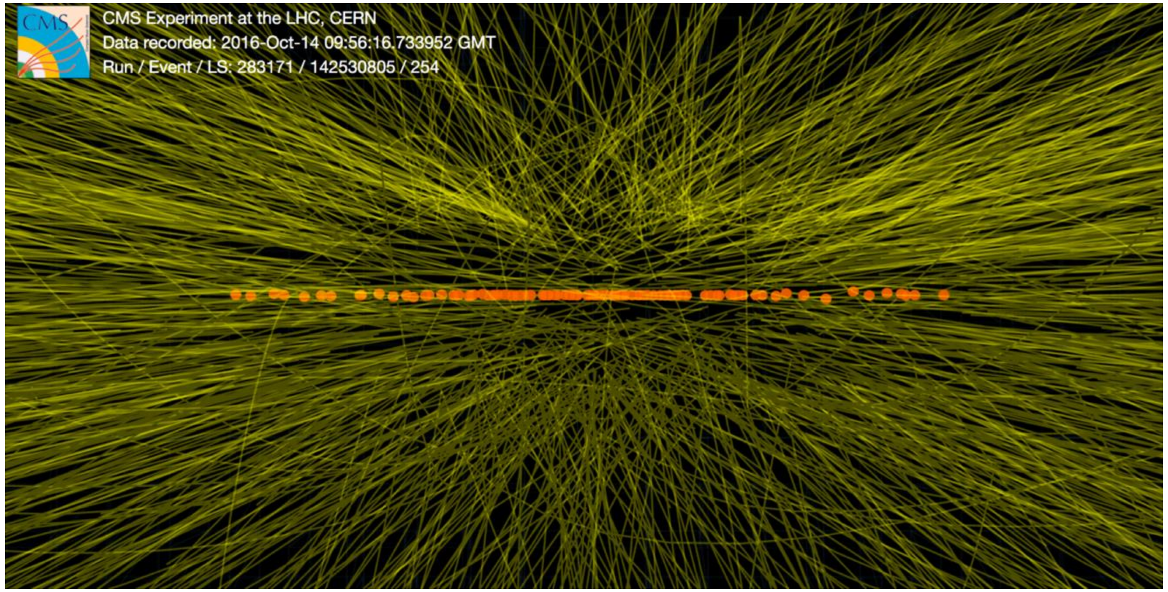


HL-LHC

- **High-luminosity** LHC era (HL-LHC) starting in ~2030 ➜ precise measurements of the Standard Model and searches for new physics
- Higher instantaneous luminosity ➜ **higher** number of **pileup** (PU) interactions <PU>=140-200
- Crucial to isolate **interaction of interest** and mitigate **effects of PU** on object reconstruction
- How? **Track-vertex association**



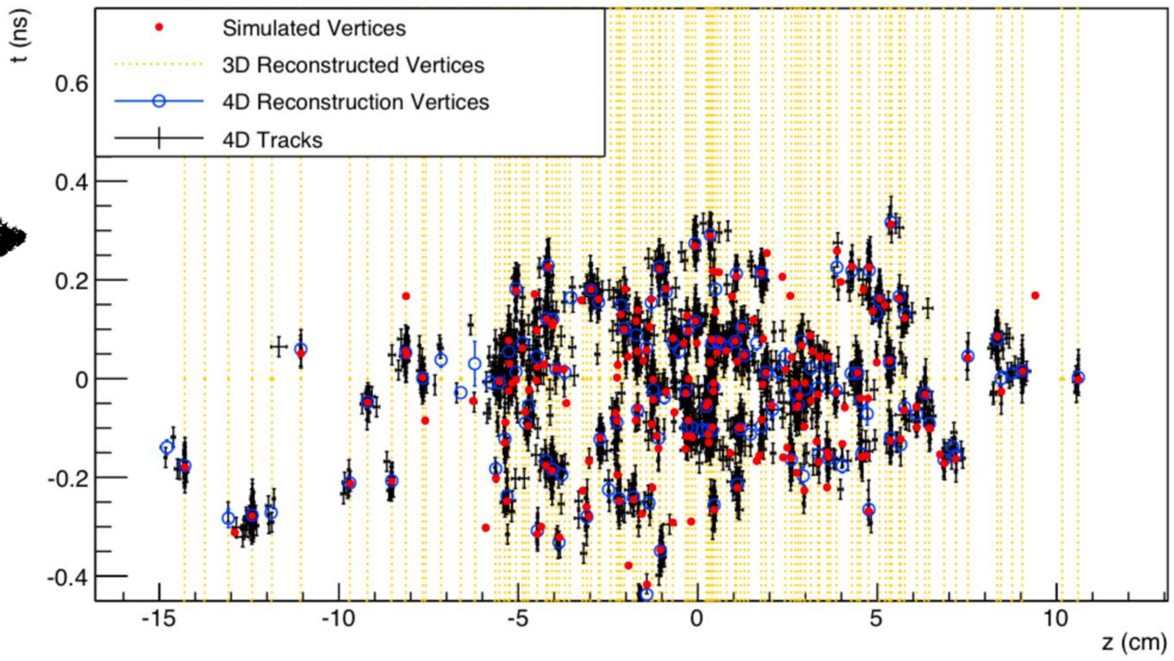
[CMS Luminosity - Public Results]



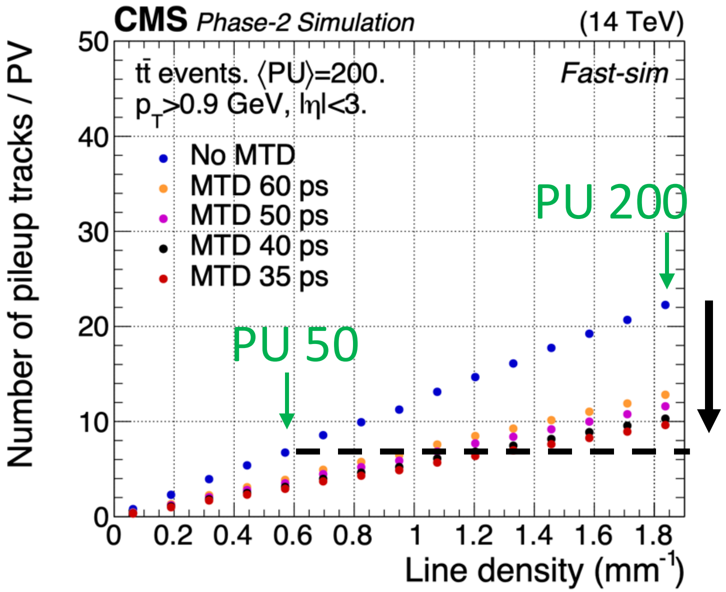
~130 pp collisions - recorded by CMS in 2016 during a high PU run

Precision timing at CMS in HL-LHC

- Use timing information to **separate vertices** that **overlap in space**
- Modern detector technologies allow **~30 ps time resolution →** smaller than the pp collision spread in time of 180-200 ps (longitudinal spread around 5 cm)
- Possible effective separation!



CMS MTD Technical Design Report



CMS MTD Technical Design Report

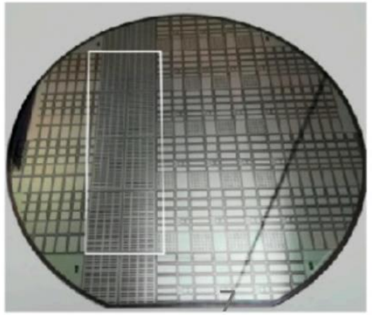
From 3D to 4D vertex reconstruction
→ effective PU as in Phase-1

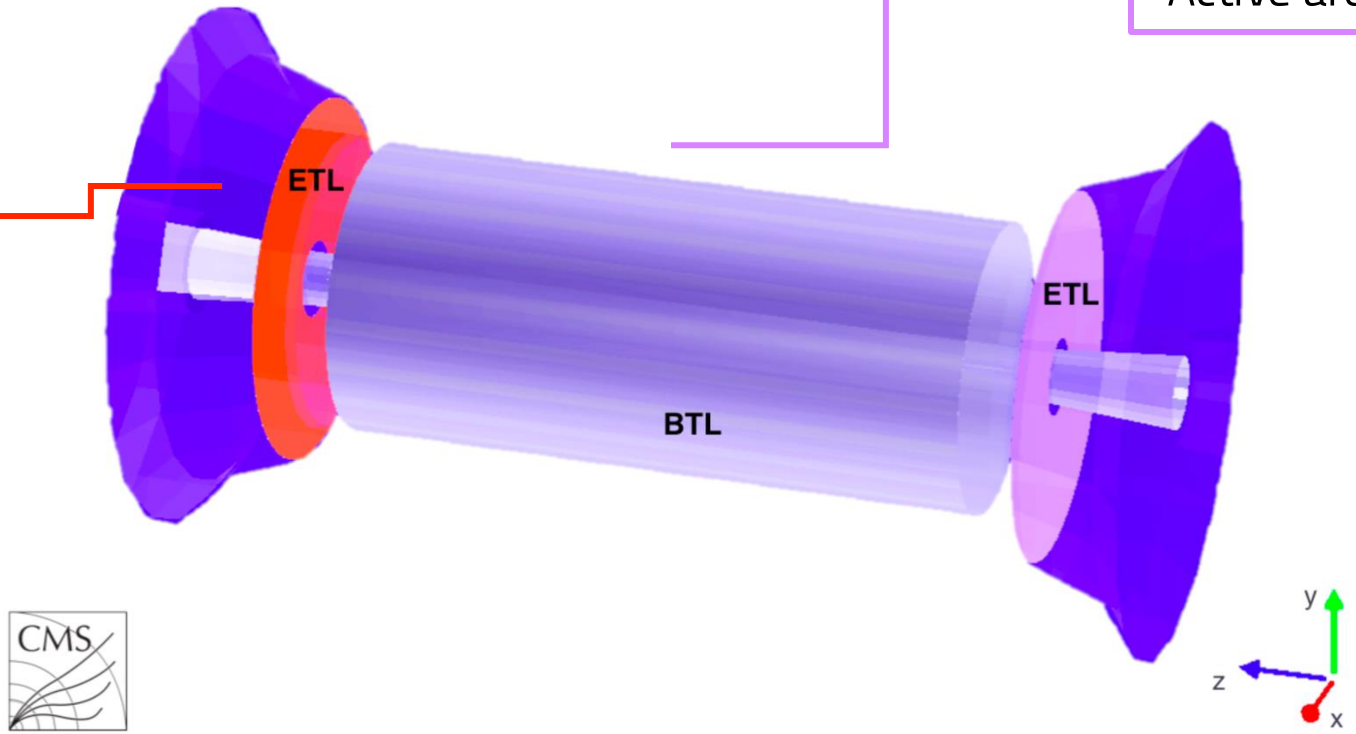
Mip Timing Detector

- Mip Timing Detector (MTD) [CMS MTD Technical Design Report] to measure time of charged particles
- Placed between tracker and calorimeter
- Almost hermetic coverage with $|\eta| < 3$

Endcap Timing Layer (ETL)

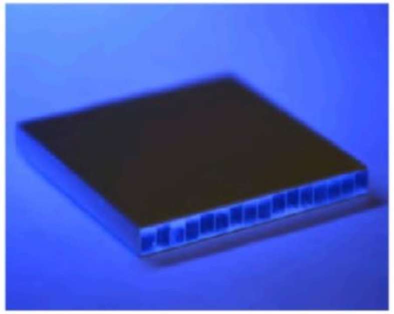
- LGADs
- $1.6 < |\eta| < 3$
- Active area ~14 m^2





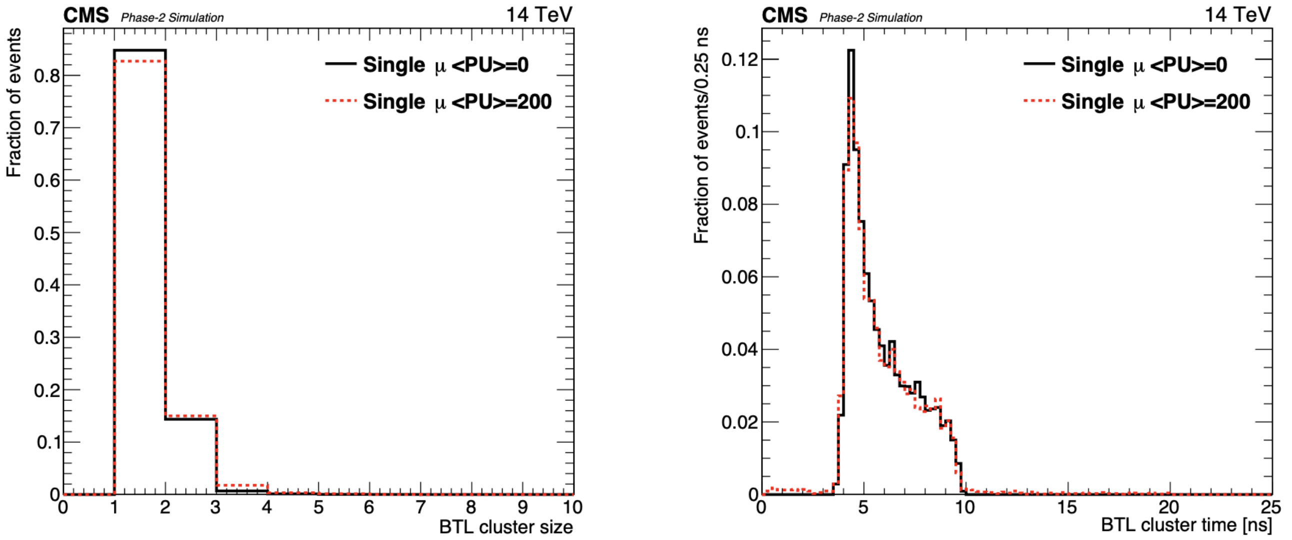
Barrel Timing Layer (BTL)

- LYSO bars + 2 SiPM/bar
- $|\eta| < 1.45$, $p_T > 0.7$ GeV
- Active area ~38 m^2



Track reconstruction with timing - 1

- **Local reconstruction**:
  - In BTL: single crystal measurements
  - $t_{av} = (t_L + t_R)/2$ (left and right SiPMs)
  - In ETL: pixel measurement



CMS MTD Technical Design Report

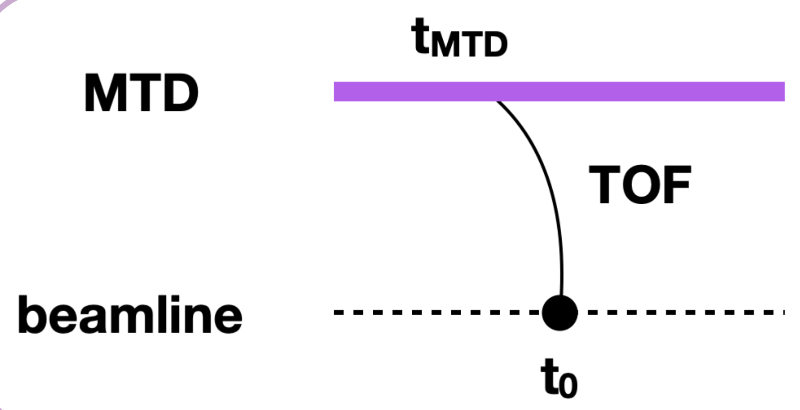
- **Topological clustering** of adjacent MTD hits:
  - In BTL: frequent multiple hits, especially at high η
  - In ETL: mostly single-hit clusters
  - Cluster time: weighted average of single-hit times

- Propagation of *tracker* **tracks** to **MTD** and **matching** with clusters
  - **Spatial** matching based on χ^2 of track extrapolation
  - **Time** compatibility with beamspot constraint
    - To suppress background, to be opmitized for searches

BDT for track-MTD cluster matching quality

Track reconstruction with timing - 2

- **Tracker + MTD track**:
  - Re-fit track parameters including additional spatial measurement
  - Compute path length
- Given path length and momentum, velocity depends on mass hypothesis, a-priori unknown
- **Propagation** to point of closest approach to beamline in various mass hypotheses (π, K, p)
  - Dedicated to primary vertex reconstruction
  - Similar approach may be used for secondary vertices exploiting their known position



$$t_0 = t_{MTD} - TOF(\pi, K, p)$$

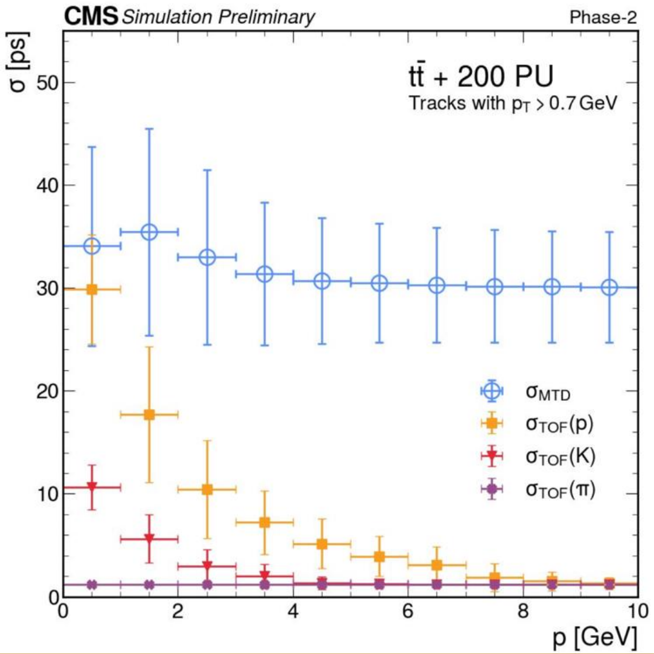
Track time uncertainty

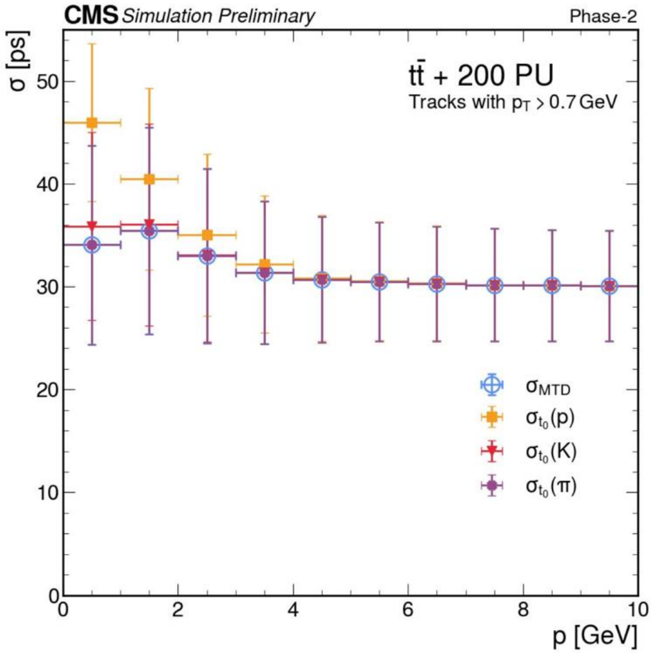
- Total **track time uncertainty**:

$$\sigma(t_0) = \sigma(t_{MTD}) \oplus \sigma(t_{TOF}) \oplus \Delta(TOF_p - TOF_\pi)$$

- $\sigma(t_{MTD})$: estimated MTD time uncertainty
- $\sigma(t_{TOF})$: TOF uncertainty
- $\Delta(TOF_p - TOF_\pi)$: inflated incertainty, due to π approximation

- $\sigma(t_{TOF})$ propagates the uncertainty on particle's velocity derived from the reconstructed track momentum, that is particle hypothesis dependent
- Studies show that it has negligible impact [CMS-DP/2024-048]





CMS-DP/2024-048

4D vertex reconstruction

- **4D vertex reconstruction** and particle identification (**PID**) go hand in hand
- Measure track time @ MTD and momentum, velocity depends on the mass hypothesis
- Obtain time from constraint that tracks from same vertex have same time
- Until recently (2024) vertex reconstruction - **legacy 4D** - in 2 steps

1st step

- Cluster vertex using **π *hypothesis*** with inflated uncertainty
  $\sigma(t_0) = \sigma(t_{MTD}) \oplus \sigma(t_{TOF}) \oplus \Delta(TOF_p - TOF_\pi)$
- calculate vertex time and perform PID*

→

2nd step

- Cluster vertex using ***updated track times*** and remove inflated uncertainty
- calculate vertex time and perform PID

*perform PID: check track compatibility with vertices in all hypotheses

The **legacy 4D** algorithm is sub-optimal:

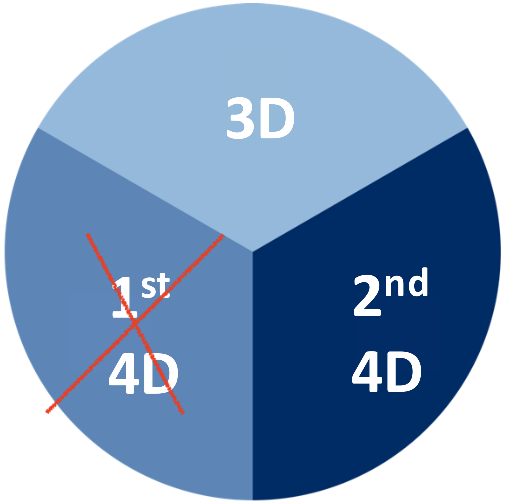
- CPU-time consuming: in 1st step, inflated uncertainty dominates over MTD uncertainty at low momenta, making time usage of limited benefit
- lower efficiency/purity than 3D

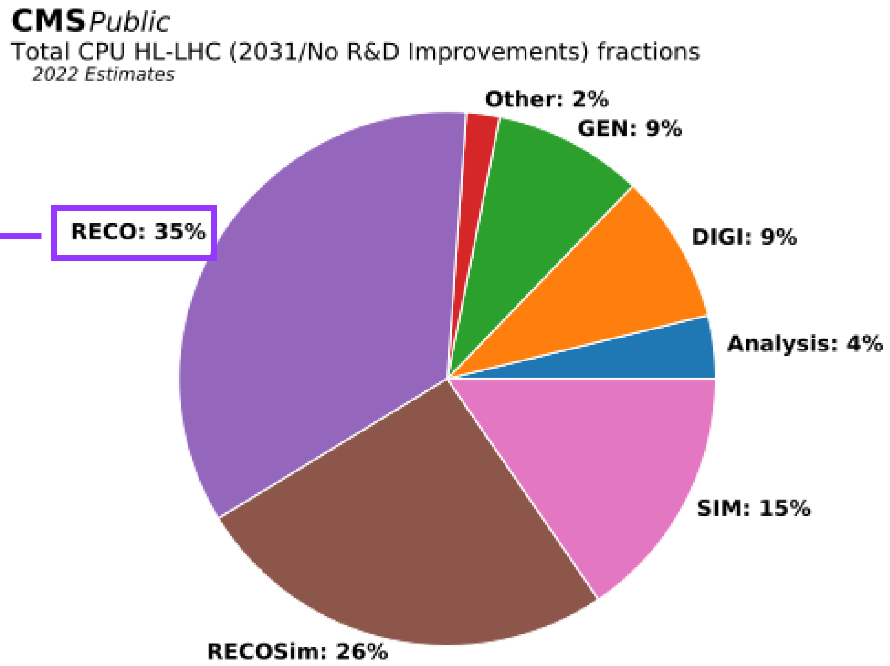
The updated 4D algorithm

- The **updated 4D** algorithm [CMS-DP/2024-085] replaces the 1st step of **legacy 4D** with 3D vertices:
  - Possible thanks to **time** computation available for 3D vertices as well (**3Dt**)
  - **Reduce** the vertex reconstruction **CPU-time** by 30% without loss in performance
- The **time computation algorithm** is updated
  - **Improved** vertex time resolution, pull and bias

RecoVertex ~10% of RECO time

Each vertex reco step ~same CPU-time
➜also with new time computation





HL-LHC projections
CMS Offline and Computing - Public Results

Vertex time computation

- In **legacy 4D**: compute time using only the mass hypothesis assigned after PID with a simple **weighted average**

$$t_{\mathrm{vtx}} = \frac{\sum_i \frac{1}{\sigma_{t,i}^2} \cdot t_i}{\sum_i \frac{1}{\sigma_{t,i}^2}}$$

t_i = track time
$\sigma_{t,i}$ = track time uncertainty

- **New**: time computed with a **deterministic annealing (DA) time algorithm** using all 3 mass hypotheses, minimizing the cost function:

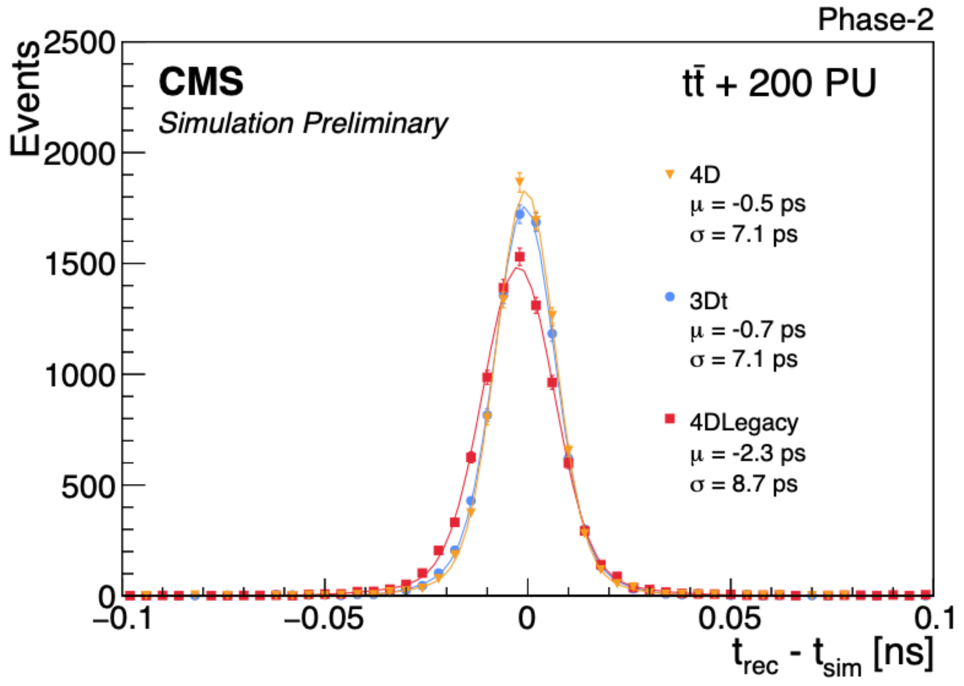
$$F = -T \sum_{\mathrm{tracks},i} w_{0,i} \log\left(Z_0 + \alpha_\pi e^{-\frac{(t_i(\pi)-t_v)^2}{2T\sigma_{t,i}^2(\pi)}} + \alpha_K e^{-\frac{(t_i(K)-t_v)^2}{2T\sigma_{t,i}^2(K)}} + \alpha_p e^{-\frac{(t_i(p)-t_v)^2}{2T\sigma_{t,i}^2(p)}} \right)$$

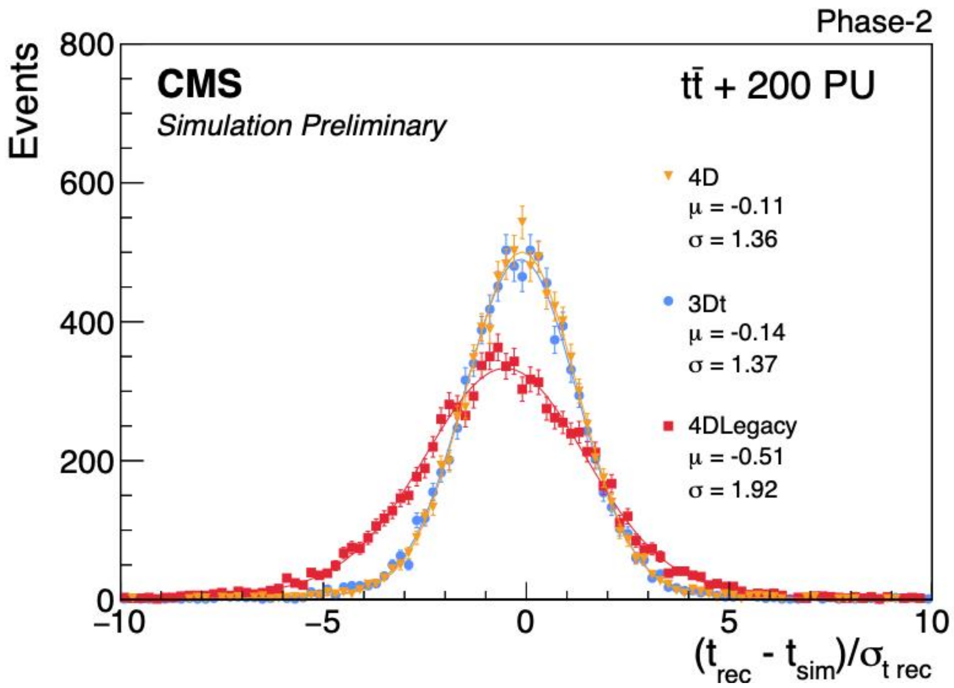
$w_{0,i}$ = track weight from adaptive vertex fit
$t_i(\pi, K, p)$ = track time for π, K, p
$\sigma_{t,i}(\pi, K, p)$ = track time uncertainty for π, K, p
$\alpha_{\pi,K,p}$ = a priori probability for π, K, p (0.7,0.2,0.1)

- This algorithm can be applied to a reconstructed vertex regardless of the use of time in its clustering and fitting:
  - **3Dt** vertex with the DA time calculation
  - updated **4D** with the DA time calculation in 2^{nd} step

Signal vertex time

- The vertex time resolution and pull for signal vertices, the distributions are fitted with a double Gaussian: the parameters shown refer to the narrowest one
- The updated **4D** and **3Dt** algorithms show an **improvement** in time **resolution** and **pull** wrt the **legacy 4D** algorithm
- The negative **bias** in **legacy 4D**, due to possible misassignment to K/p hypothesis, is **reduced** in the **3Dt** and updated **4D** thanks to time computation always accounting for all mass hypotheses



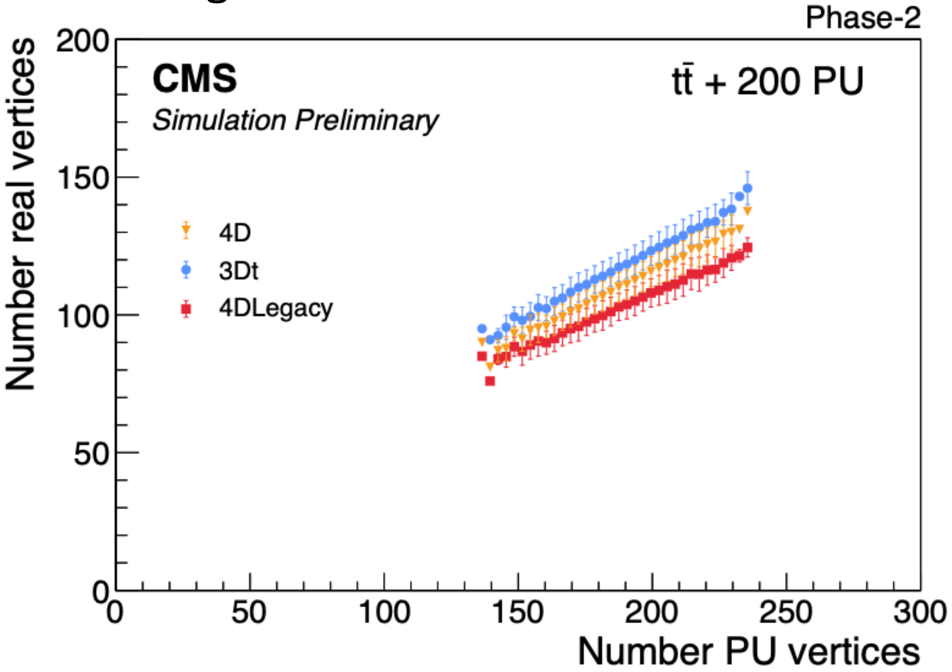


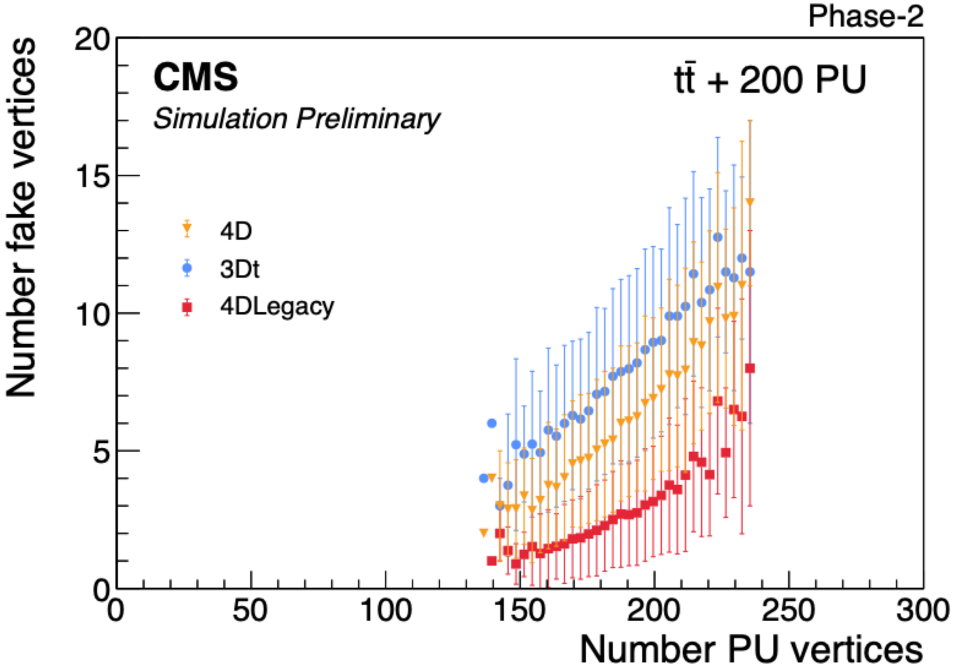
CMS-DP/2024-085

Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

Number of vertices

- The number of reconstructed vertices as a function of the number of PU vertices for **real** and **fake** vertices
- The erorr bars indicate the RMS of the distributions
- Classification based on **matching to MC truth** - both true tracks and vertices (details in backup)
- The **3Dt** reconstructs more real vertices, but also more fakes
- The updated **4D** algorithm shows a **higher number of vertices** than the legacy, with a performance in between the **4D legacy** and the **3Dt** algorithms

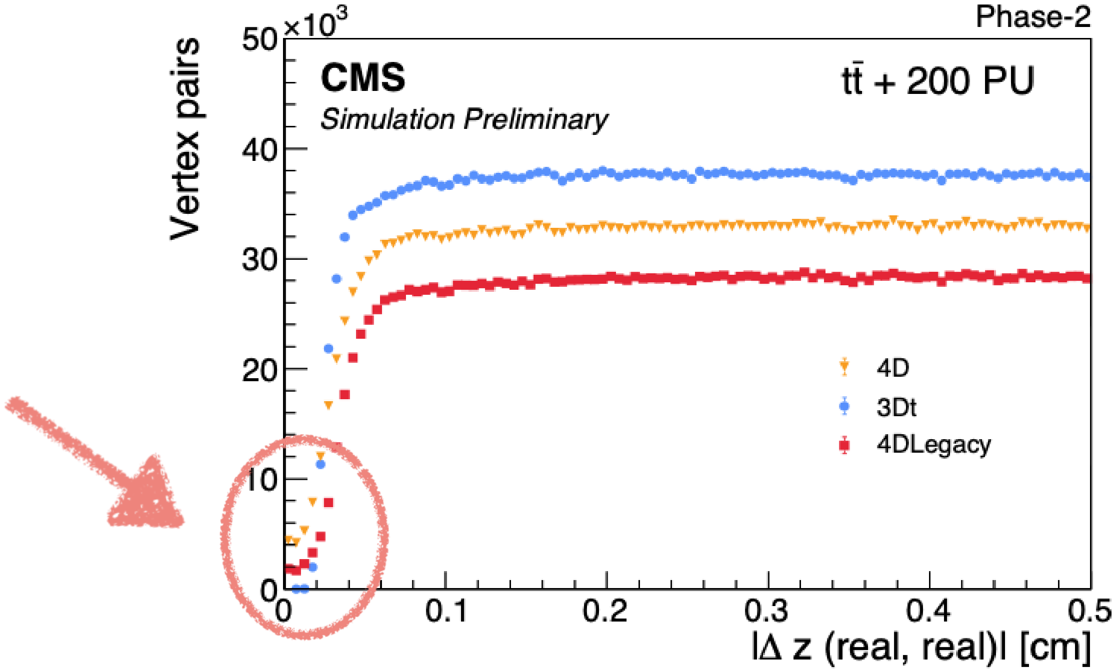


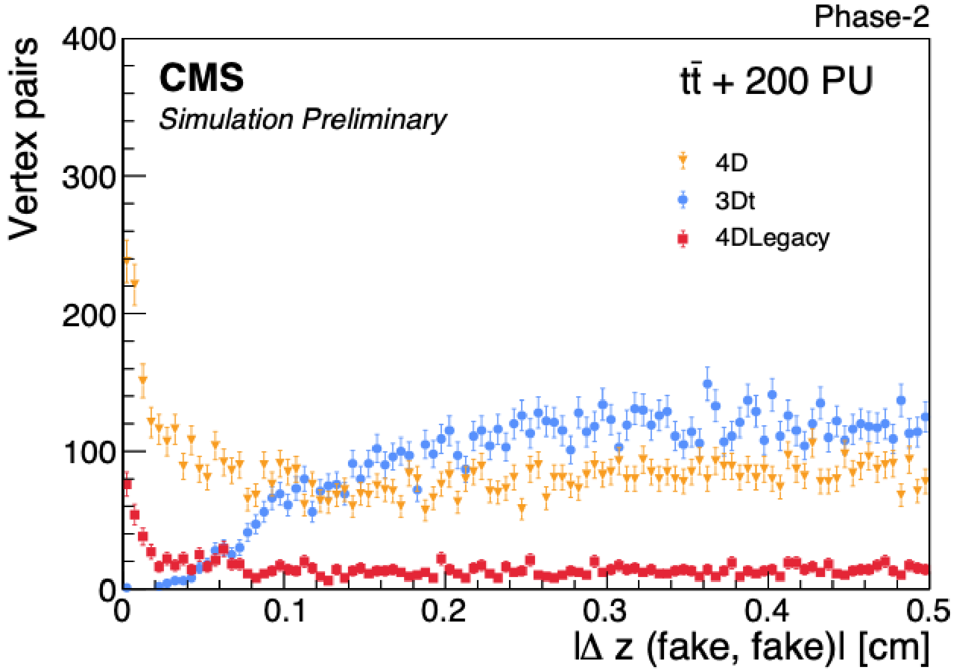


CMS-DP/2024-085

Distance in z

- Distance between pairs of reco vertices: the updated **4D** algorithm shows more real-real vertex pairs close in *z* than **legacy 4D**, but also more fakes
- The **3Dt** algorithm is not designed to reconstruct vertices with separation less than ~0.3 mm
- The **improvement** in the new algorithm is visible especially for **real vertex pairs** with **Δ*z* close to 0**
- Advantage in the use of **timing**: vertices that overlap in space can be separated in time

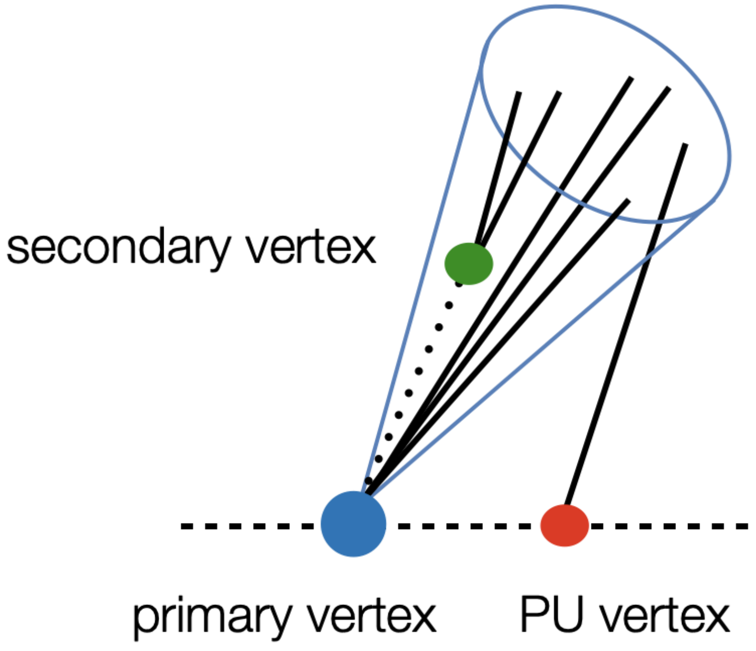




CMS-DP/2024-085

Pileup contamination

- Compare vertex algorithms in terms of **PU rejection**:
  - crucial for object reconstruction in HL-LHC
  - primary goal of MTD
- Monitor **track-based** and **jet-based** observables



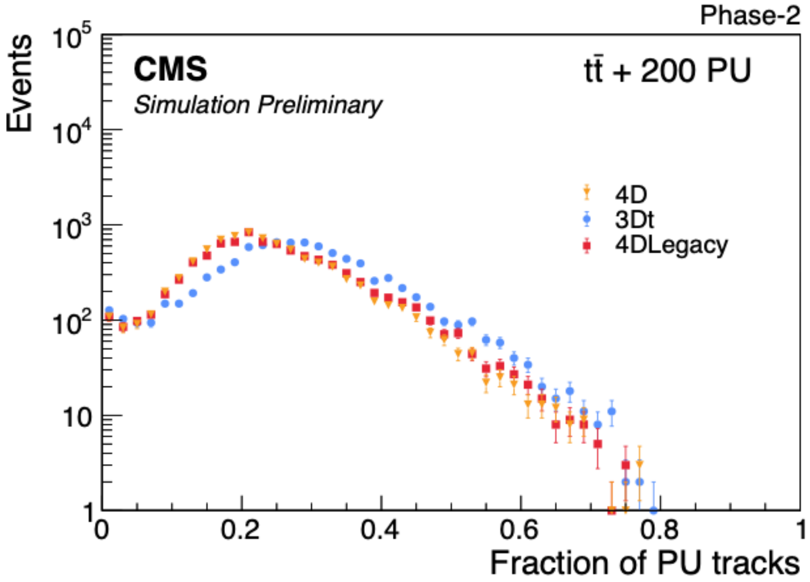
- **Tracks** associated to a reconstructed vertex are classified based on MC truth matching as:
  - Track from primary vertex
  - Track from secondary vertex
  - PU track
  - Fake track, not matched to any true particle

- **Track-jets** are built by clustering reconstructed charged tracks originating from the same vertex
- The relative **contribution of PU** to track-jet-based quantities is estimated by clustering jets without the PU tracks and recomputing the observables

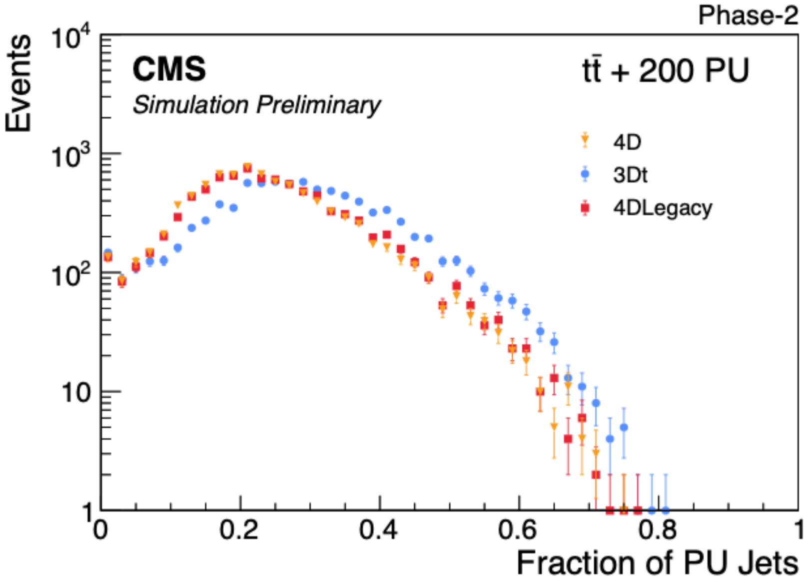
Pileup contamination for the leading vertex

- Impact of PU on track multiplicity, track-jet multiplicity and sum of p_T^2 of track-jets
- On one hand, a general **reduction** in the **PU contamination** of ~10-15% can be seen in the **4D algorithms** with respect to the **3Dt** one
- On the other, variables like the sum of track-jet p_T^2, used in the vertex sorting, are less sensitive to the vertex reconstruction algorithm, as expected because of the natural precense of high-p_T jets in the $t\bar{t}$ signal component compared to PU

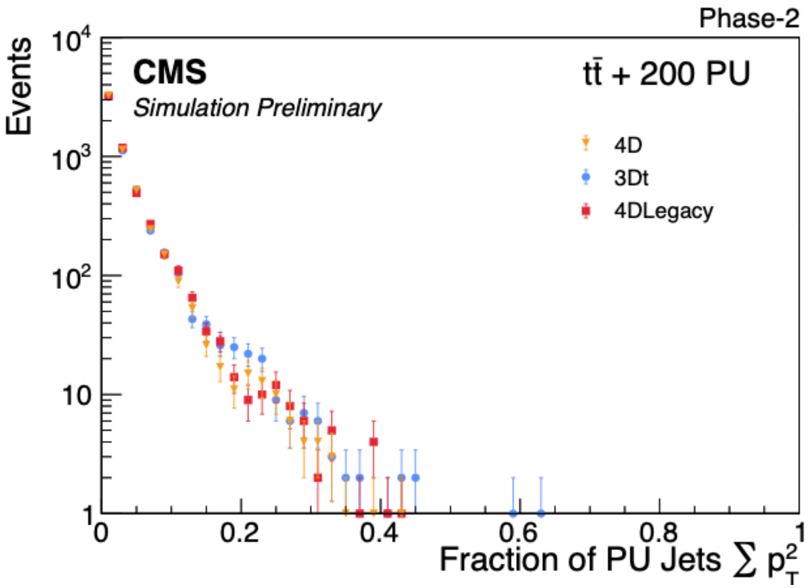
N_{PUtrks}/N_{trks}



$(N_{jets}-N_{jets\_noPU})/N_{jets}$



$(\Sigma_{jets}\, p_T^2 - \Sigma_{jets\_noPU}\, p_T^2\,) \,/\, \Sigma_{jets}\, p_T^2$



CMS-DP/2024-085

Conclusions

- The timing information of **MTD** is important to **mitigate** the effects of **PU** in HL-LHC
- A set of tools has been developed for evaluating the performance of different vertex reconstruction algorithms in terms of:
  - Vertex **time resolution**
  - Number of reconstructed **true** and **fake** vertices
  - **PU rejection**
- They serve as bechmark for future developments and exploration of new techniques
- The optimization of the **4D** vertex reconstruction is presented
- The results highlight the advantage in the use of **timing**:
  - **vertices overlapping in space can be separated in time improving PU rejection**

BACKUP

References

[1] CMS Collaboration, "A MIP Timing Detector for the CMS Phase-2 Upgrade", technical report, CERN, Geneva, 2019.

[2] CMS Collaboration, "Improved use of MTD time in vertex reconstruction", CMS DP-2024-048 (2024).

[3] CMS Collaboration, "Update of the vertex reconstruction using track time from MTD", CMS DP-2024-085 (2024).

Vertex association to MC truth

To evaluate the performance of the vertex reconstruction, an algorithm has been developed to **match reconstructed vertices** to **MC truth**, based on the common origin of tracks in the reconstructed and simulated vertices. The reconstructed tracks are matched to the true simulated charged particles, and the simulated vertices from which they originate define the set of true primary vertices in the event. The matching algorithm is based on the sum of weights:

$$W_{\text{os}} = \frac{w_{trk}}{\sigma^2_{z,trk}} \frac{1}{\text{erf}(\sigma_t/\sigma_T)}$$

where w_{trk} is the weight assigned by the adaptive vertex fit, $\sigma_{z,trk}$ the track resolution, σ_t the track time uncertainty and σ_T the time width of the beamspot. The time dependent part is present only for tracks with time information.

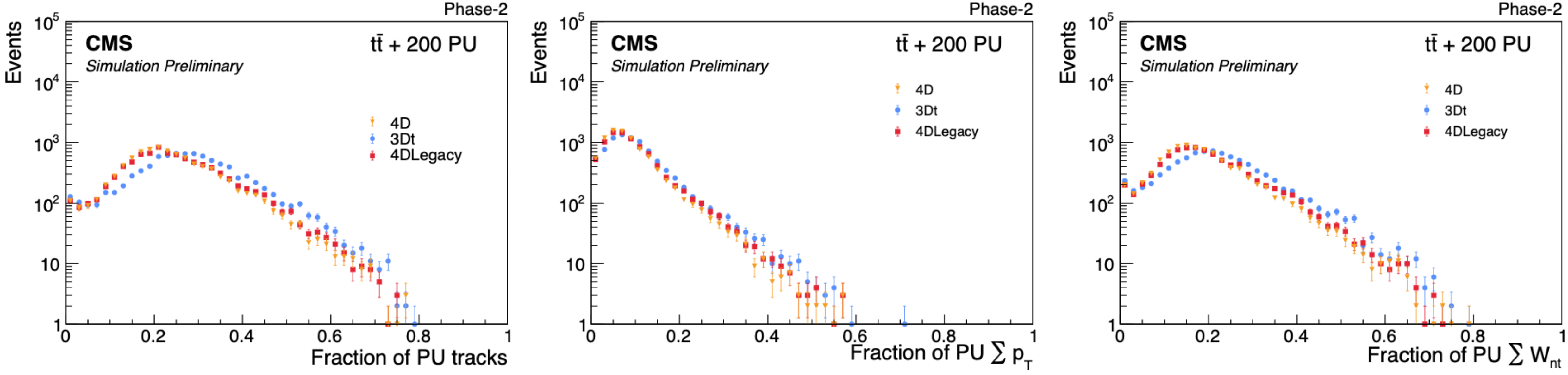
A **one-to-one matching** is performed: for a given reconstructed vertex, the dominating simulated vertex is the one with largest sum of Wos. Whenever possible, the dominating simulated vertex is matched to the reconstructed one. If a simulated vertex dominates more than one reconstructed vertex, the match is made between that simulated vertex and the reconstructed one that receives the largest weight among the dominated reconstructed vertices. The algorithm proceeds in an iterative manner for all the reconstructed vertices. Depending on the outcome of the algorithm, vertices are classified as:

- **real**: a good matching is found within the maximum allowed number of iterations (set to 8),
- **fake**: no matching can be found, meaning that there is no simulated vertex dominating the reconstructed vertex that does not dominate other vertices more.

CMS-DP/2024-085

Pileup contamination – track observables

- Impact of PU on track multiplicity, sum of track p_T and $\sum_{trk} W_{\mathrm{nt}}^{trk} = \sum_{trk} w_{trk} \times \min(p_{T,trk}, 1)$, the sum of weights of tracks, where low momentum tracks are downgraded, an auxiliary weight in the matching algorithm
- A **PU reduction** of ~10-15% can be seen in the **4D algorithms** with respect to the **3Dt** one

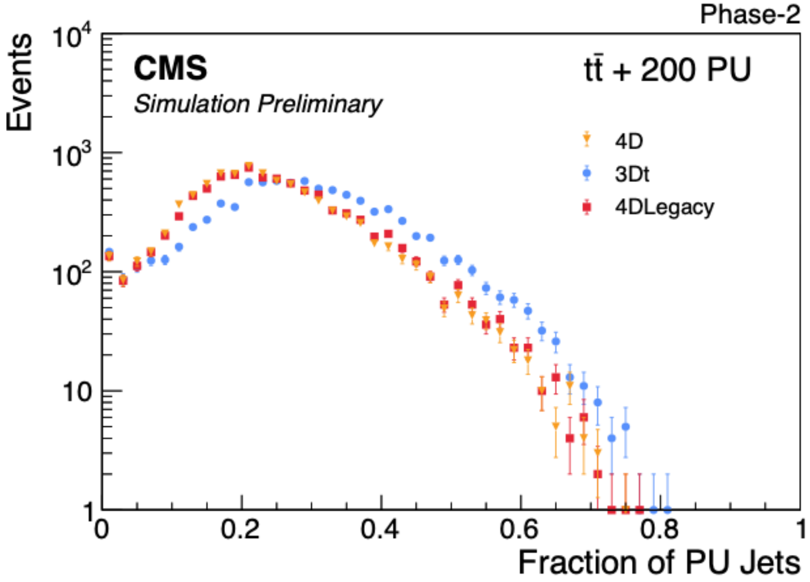


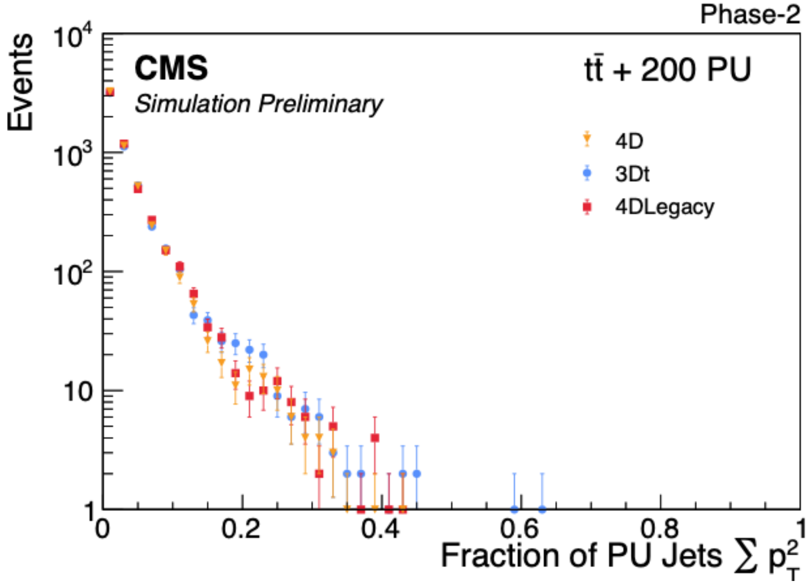
CMS-DP/2024-085

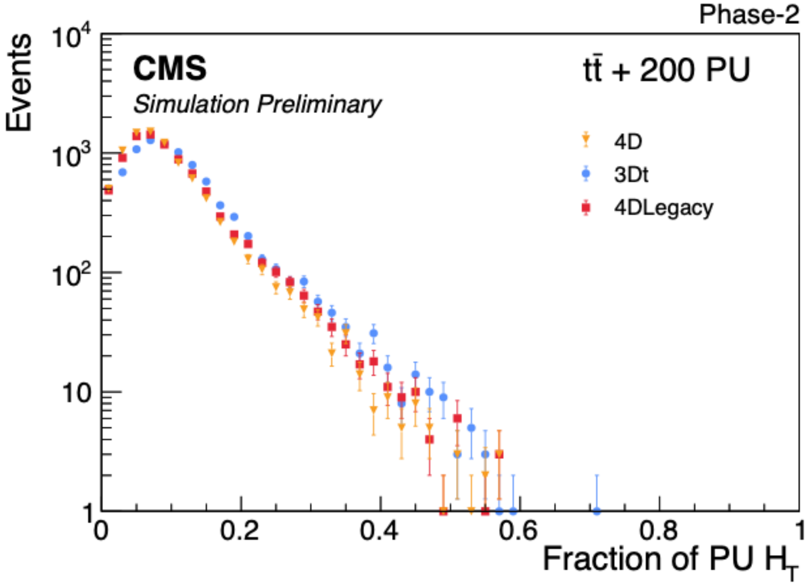
Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

Pileup contamination – track-jet observables

- Impact of PU on track-jet multiplicity, sum of p_T^2 of track-jets and H_T, the sum of the transverse energy of track-jets
- A **PU reduction** of ~10-15% can be seen in the **4D algorithms** with respect to the **3Dt** one in some variables, while the sum of track-jet p_T^2, used in the jet sorting, is insensitive to the vertex reconstruction algorithm







CMS-DP/2024-085