Stato della preparazione per il Run 2017: Tracking e BTagging

Erica Brondolin
On behalf of TRK-POG

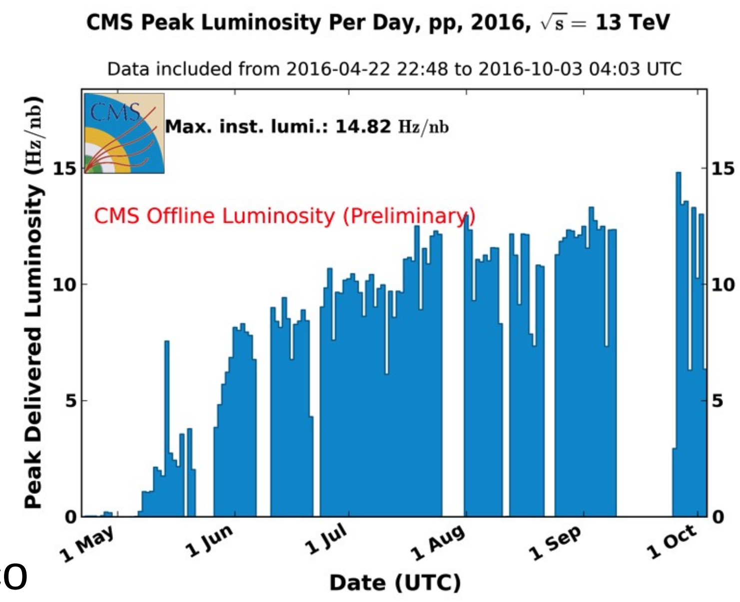
Riunione CMS Italia 2016
14th December 2016

2016 data taking

2016 Run

2016 data taking:

- Peak values higher than ever:
  - Peak inst. luminosity ~1.5×10^{34} cm^{-2} s^{-1}
  - Peak pileup ~48
- Both Pixel and Strip detectors behave nicely with such a large luminosity and pileup



What did we know?

- 25 ns bunch crossing
- Designed <PU> ~ 25
- Calibrations and alignment updated for the data re-reco
- Pixel dynamic inefficiency (non-negligible)

What did we NOT know?

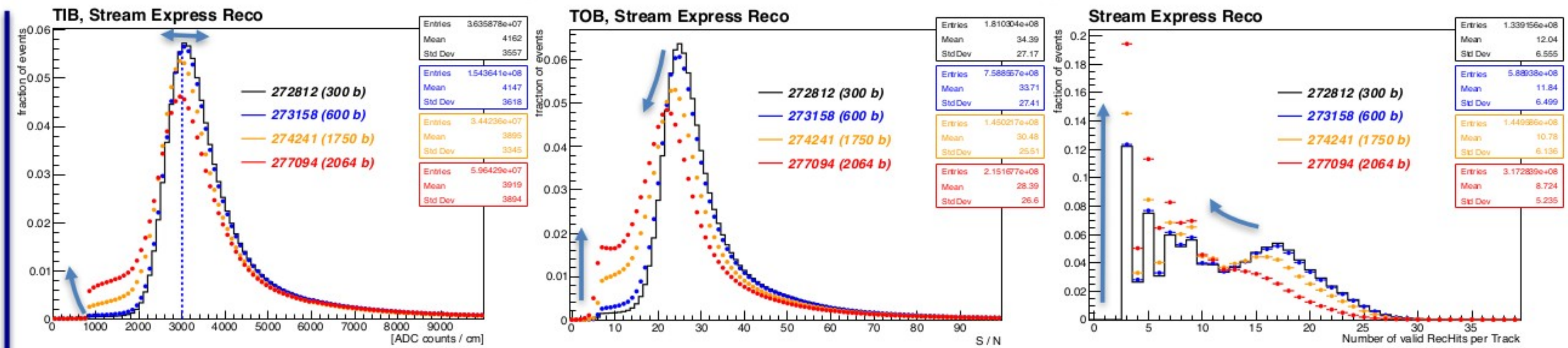
- Strip hit inefficiency problem
- <PU> conditions well above the design ones

How did we solve it?

- Strip ClusterChargeCut activated for 25 ns running.
- Monitor continuously data quality, HLT, prompt reconstruction.
- Relax the cuts related to the hit multiplicity in b-tagging track selection.
- Simulation efforts to better quantify the hit ineffeciency are underway (both the chip side and physics one).

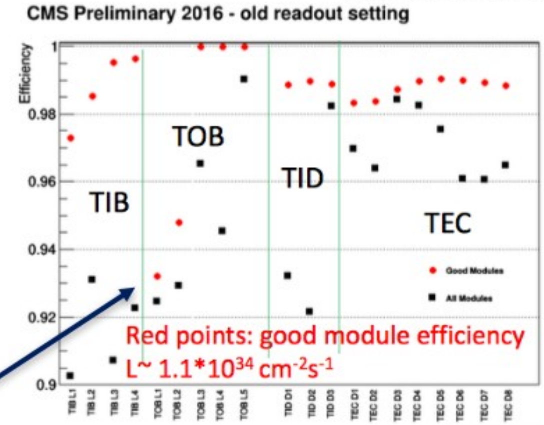
Monitoring hit inefficiency

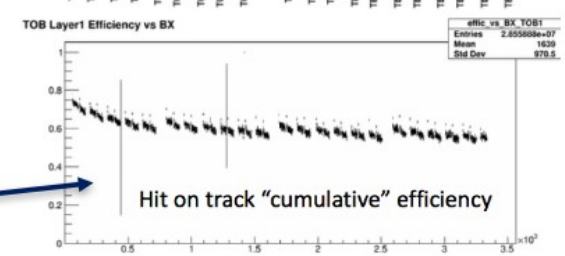
Strip hit inefficiency



In 2015 and 2016 higher and higher inefficiency was observed in the strip tracker

- Too low hit signal
  - hits are either not reconstructed or not used in reconstruction;
- Number of hits on track reduced
  - not good for B-tagging
- Tracking efficiency affected
- Main features:
  - effect increase with luminosity
  - TOB layer 1 and 2 are the most hit by this
  - dependence on the position of the LHC orbit

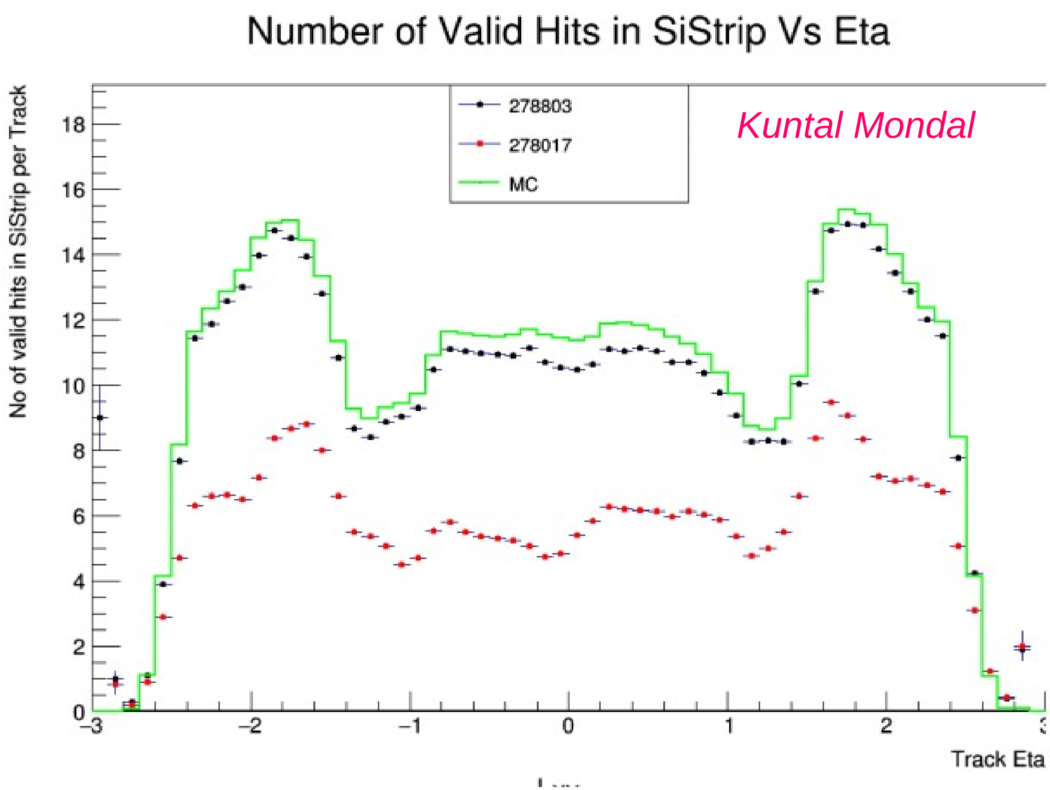


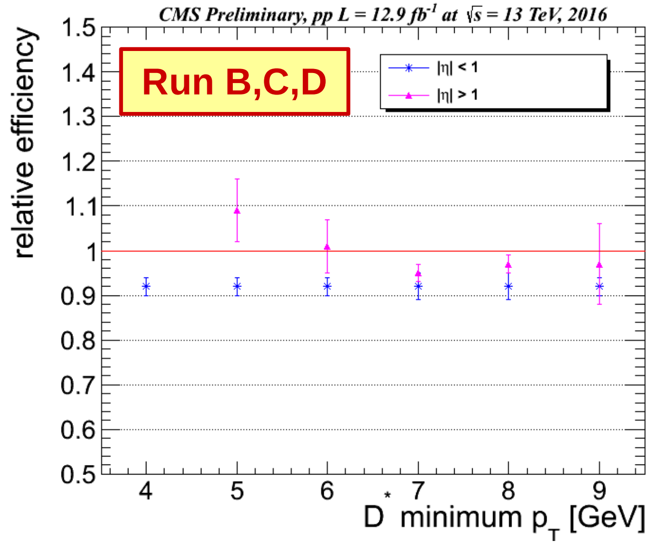


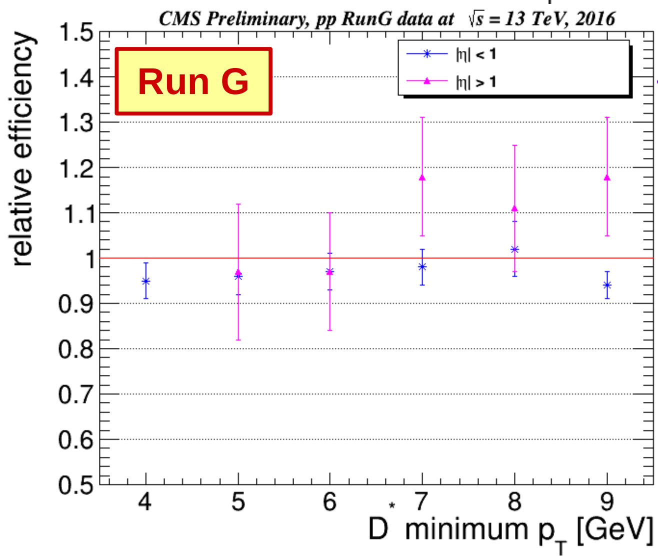
Monitoring hit inefficiency

<u>Standard candle for tracking:</u>

- **Agreement with MC**
- Beam spot measurement
- **D* analysis**
- Low-mass resonances
- Muon tracking efficiency
- Vertex resolution







Valentina Mariani

With APV fix → agreement improves greatly

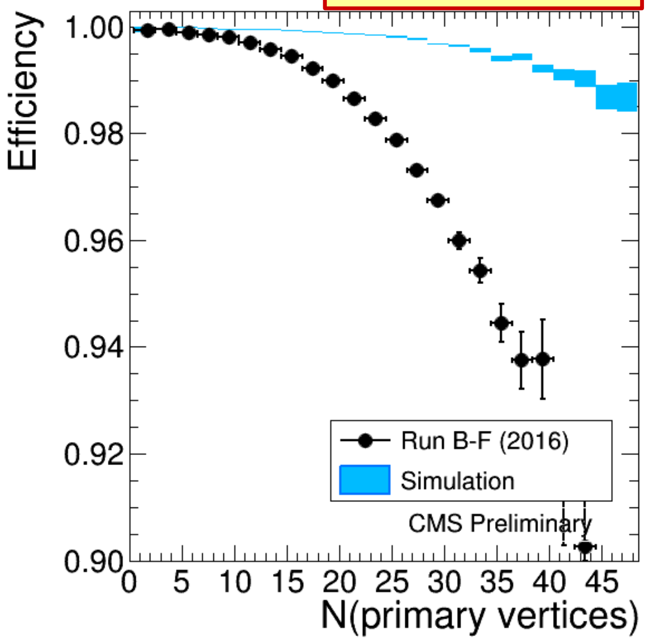
Monitoring hit inefficiency

Standard candle for tracking:

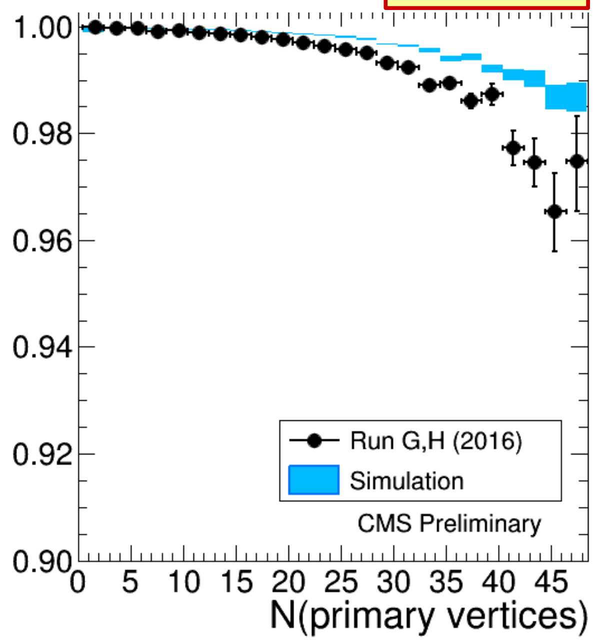
- Agreement with MC
- Beam spot measurement
- D* analysis

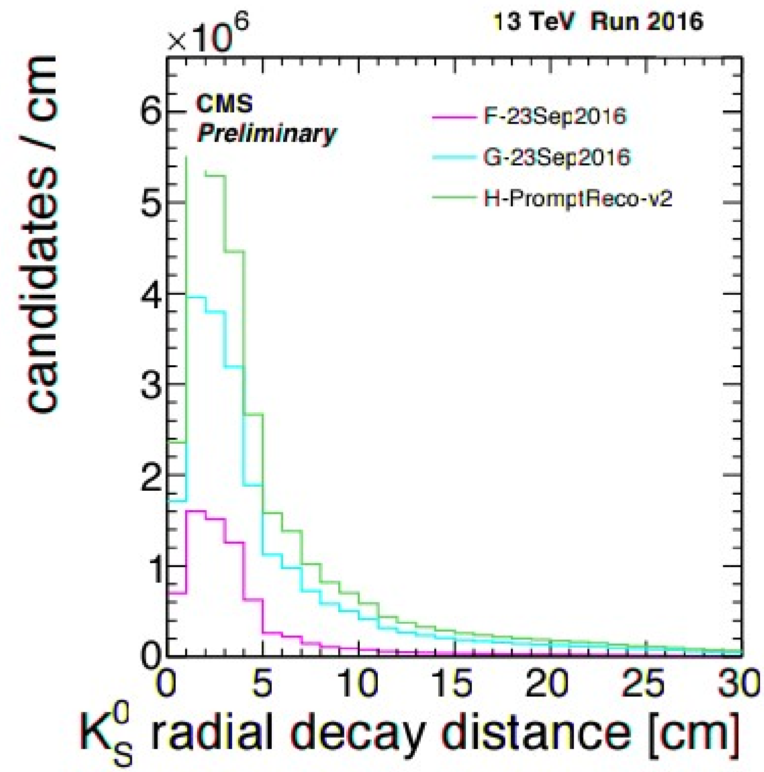
- **Low-mass resonances**
- **Muon tracking efficiency**
- Vertex resolution

Run B,C,D, E, F



Run G, H





Erica Brondolin, Giovanni Petrucciani

Frank Owen Jensen, Kevin Stenson

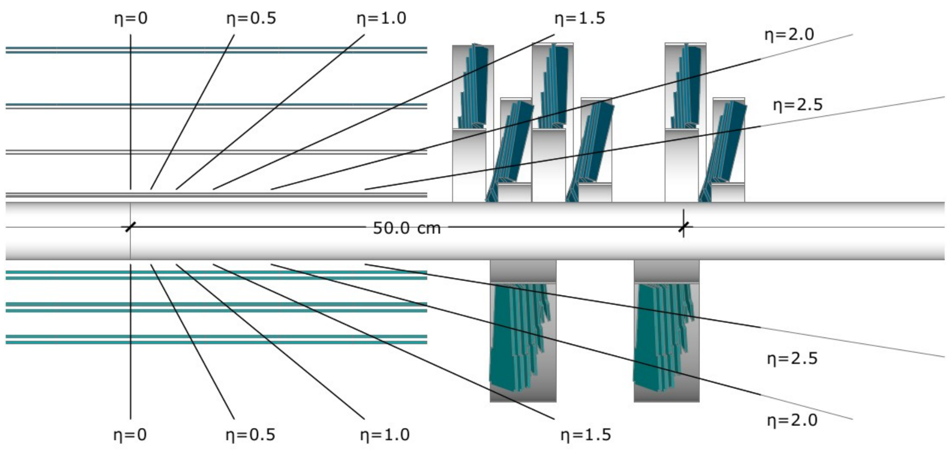
With APV fix → agreement improves greatly

Phase 1 readiness

Introduction

2017 data taking:

- Expected the same integrated luminosity to be collected;
- Peak values higher than those in 2016:
  - Peak inst. luminosity ~1.9×10^{34} $cm^{-2} s^{-1}$
  - Peak pileup ~55



Phase 1 pixel detector is expected to Improve:

- Impact parameter resolution and b-tagging
- Projection into strip tracker (4th layer)
- Efficiency and fake rate (4th pixel hit)
- Hit efficiency (improved ROC)
- No more pixel dynamic inefficiency

- Goal for tracking has been to take the advantage of the 4th pixel layer already in seeding.

Software development schedule:

- Development of reconstruction and calibration are tied with the CMSSW schedule.
- Tracking for run2, phase1 and phase2 are now running in parallel within the same release.
- Software needed to be develop following geometry and material budget changes.
- Obvious constraints from eventual MC production.
- Phase 1 tracking largely unchanged since 810pre7 but some developments coming soon.
- Targeting releases for developments and data taking: 90x and then 91x.

Tracking for Phase 1

Matti Kortelainen

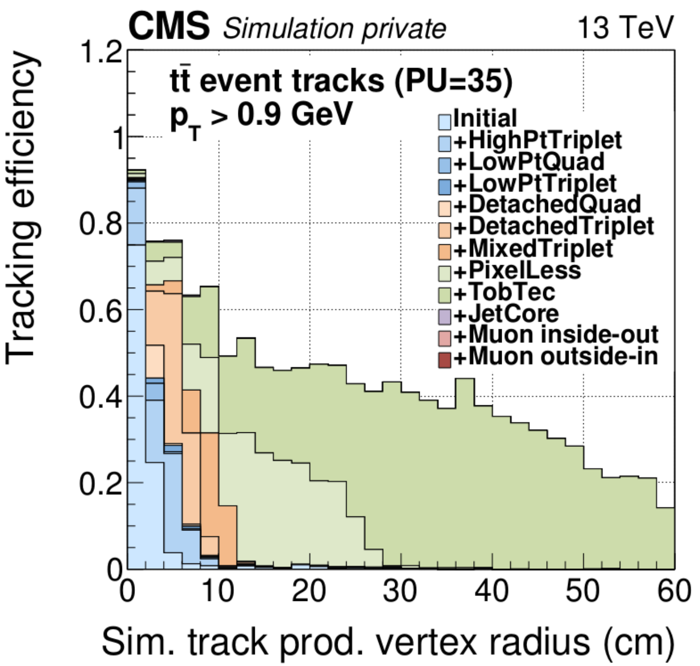
step name	seeding	target track
Initial	pixel quadruplets 3)	prompt, high p_T
LowPtQuad	pixel quadruplets 2)	prompt, low p_T
HighPtTriplet	pixel triplets	prompt, high p_T recovery
LowPtTriplet	pixel triplets	prompt, low p_T recovery
DetachedQuad	pixel quadruplets 2)	displaced--
DetachedTriplet	pixel triplets	displaced-- recovery
MixedTriplet	pixel+strip triplets	displaced-
PixelLess	inner strip triplets	displaced+
TobTec	outer strip triplets	displaced++
JetCore	pixel pairs in jets	high p_T jet
Muon inside-out	muon-tagged tracks	muon
Muon outside-in	standalone muon	muon

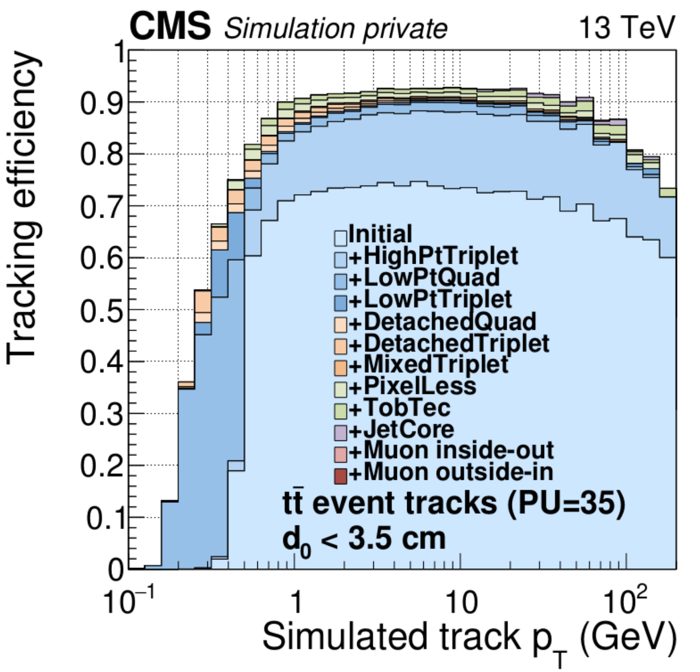
2) Triplet propagation
3) Pixel seed extension

- Currently using 2016 track selection MVA out-of-the-box
  - Works well for pixel quadruplet-seeded iterations
  - But high fake rate for pixel triplet-seeded iterations

⇒ using cut-based selection for HighPtTriplet and LowPtTriplet

- MVA is being retrained





Quadruplet seeding

Different candidates have been exploited:

Vincenzo Innocente, Matti Kortelainen

1) Triplet merging: used in Phase1 TDR (and Phase2 TP)

- Create hit triplets from all 3-layer combinations of the 4 layers
- Merge triplets sharing 2 hits to quadruplets
- First approach, efficient but slow...

2) Triplet propagation:

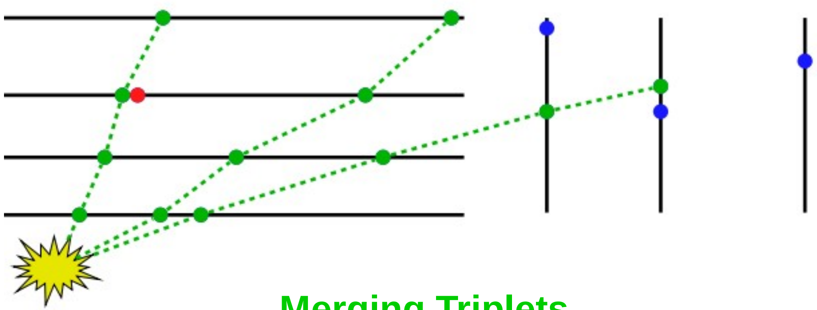
- Propagate 1-2-3 triplet to 4th layer and search for compatible hits
- Candidate also for HLT pixel tracking
- 2x faster than "merging"

3) Pixel seed extension:

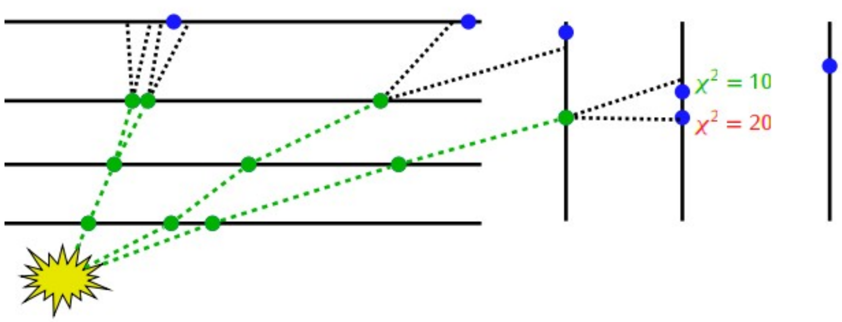
- Variation of previous: use Kalman filter for the propagation
- Seeded by triplets from layers 1-2-3
- In pattern recognition, stop trajectory propagation if no 4th pixel hit right after the seed

4) Cellular automaton (CA):

- Candidate also for HLT pixel tracking
- Details on the next slide



Merging Triplets



Triplets Propagation

Quadruplet seeding with CA

Marco Rovere,
Vincenzo Innocente,
Felice Pantaleo

Basic idea:

1) Create hit pairs from layers 1-2, 2-3, and 3-4
2) Join compatible pairs that share hits
   - Compatibility checked e.g. with Δθ
3) Then, create quadruplets from these triplets
4) Lastly, quadruplets are filtered based on quality of approximate fits of the Trajectory

For more technical information see for example slides by Felice.
For information available in slides form the PixelTrackingGPU meetings.

Pro:

- Calculations are simple, and localized in memory
- Straightforward to parallelize efficiently

− Important property for modern computing architectures

Status:

- Introduced new seeding framework.
- First goal of the project is to produce pixel tracks from RAW in GPUs in HLT in 2020.
- A CPU version is being tested for **HLT pixel tracking** already for 2017
- Goal to make CA seeding the default in 90X and hence for 2017 data (and MC)
- First tuned sequence is being integrated (likely for 900pre3) for the offline tracking.

Performance

Comparison between:

- **Run2 tracking with current detector** vs **Phase1 tracking** in CMSSW_8_1_0_pre9

- Same software release, similar input

− ~9k events of ttbar + <PU>=35
− Different geometries

- Slightly different conditions:

− Phase1 has ideal pixel alignment
− Phase1 has all pixel modules fully functional

- Phase 1 quadruplet seeding used: ***Triplet propagation*** and ***Pixel seed extension***
  → results are preliminary.

- Comparing high-purity tracks with selections indicated in the plots

- Reconstruct ~15% more tracks in Phase 1

- Timing increment expected not only by the increase of performance but above all because of the larger cluster multiplicity due to the new geometry.

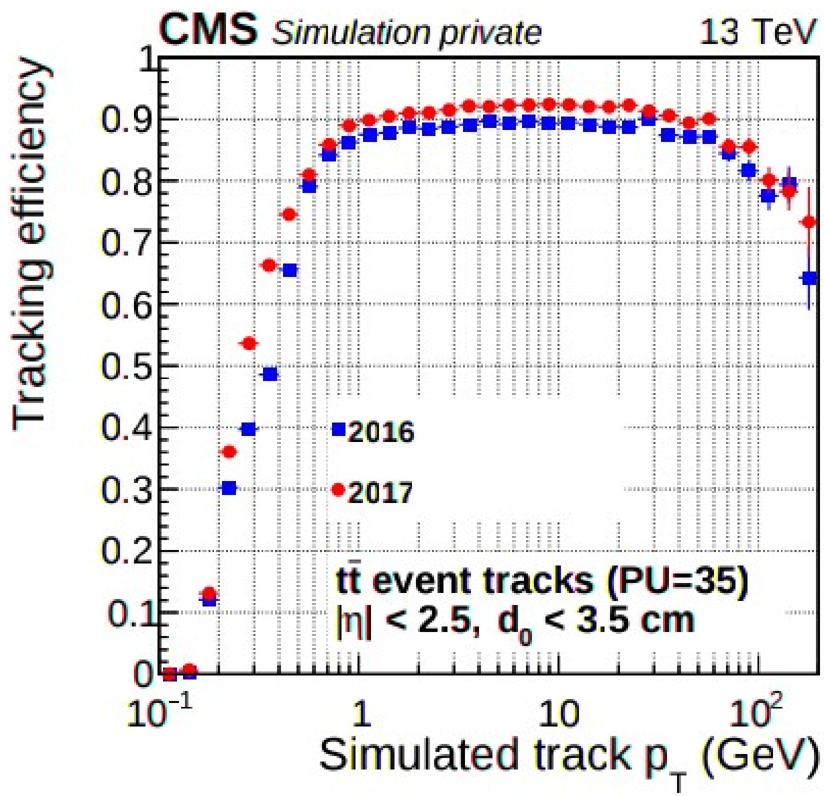
- First timing result:
  Run 2 tracking: 3.3 s/event
  Phase 1 tracking: 3.7 s/event

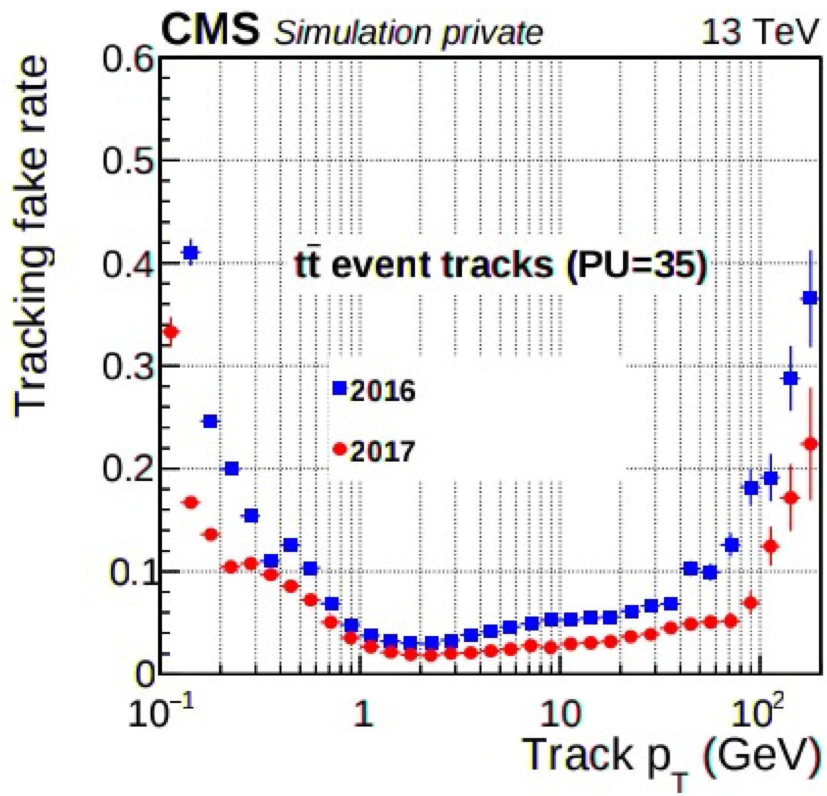
→ Some room for speedups → first version of CA seeding: entire tracking sequence ~10% faster

Performance

Efficiency and fake rate vs. p_T:

Matti Kortelainen



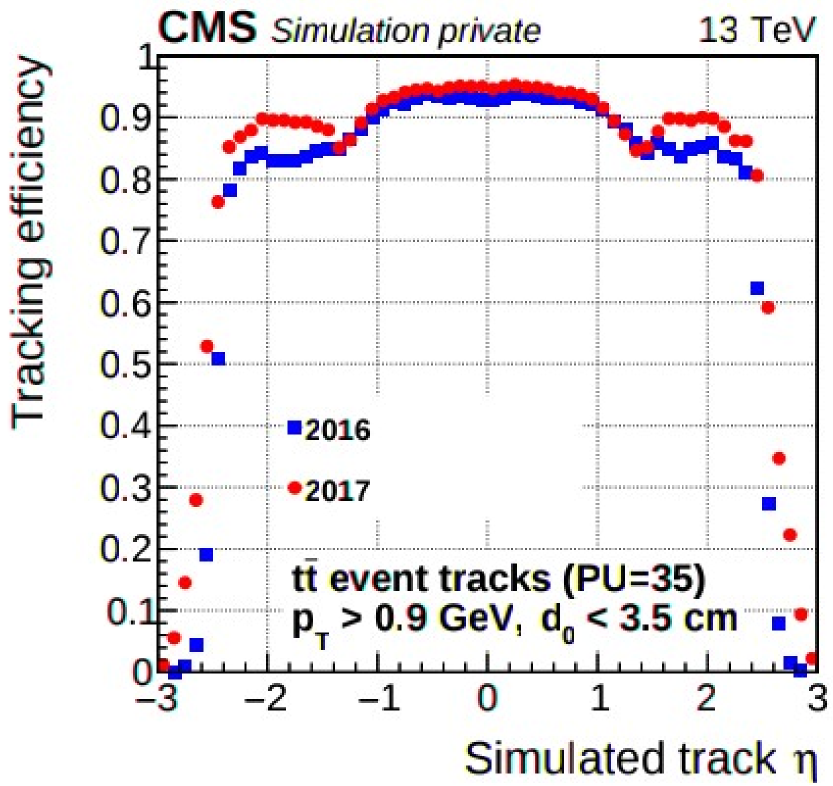


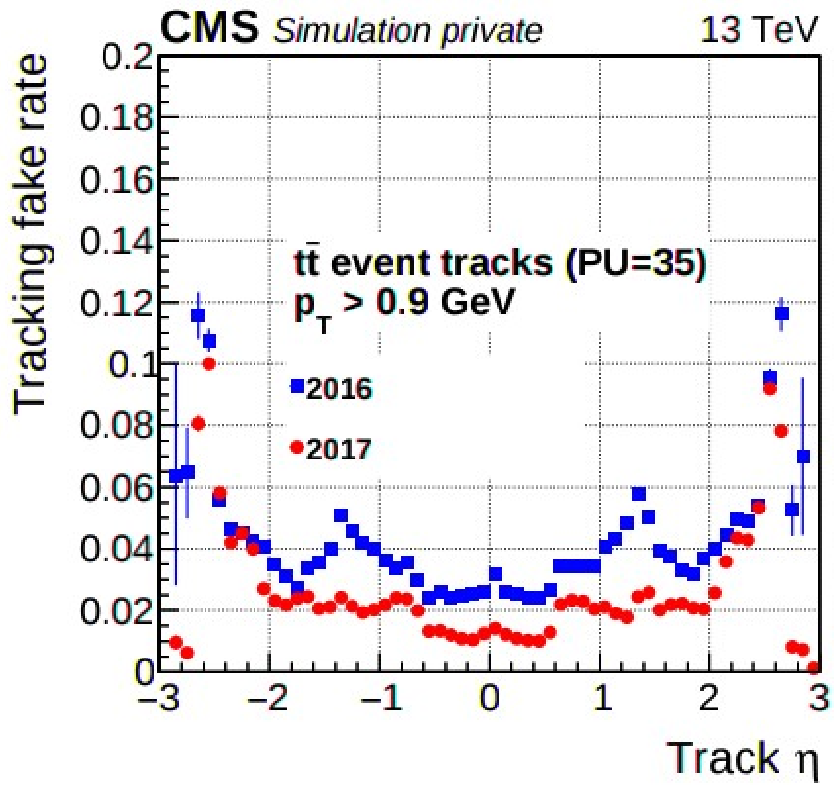
- In general higher efficiency and lower fake rate

Performance

Efficiency and fake rate vs. eta:

Matti Kortelainen



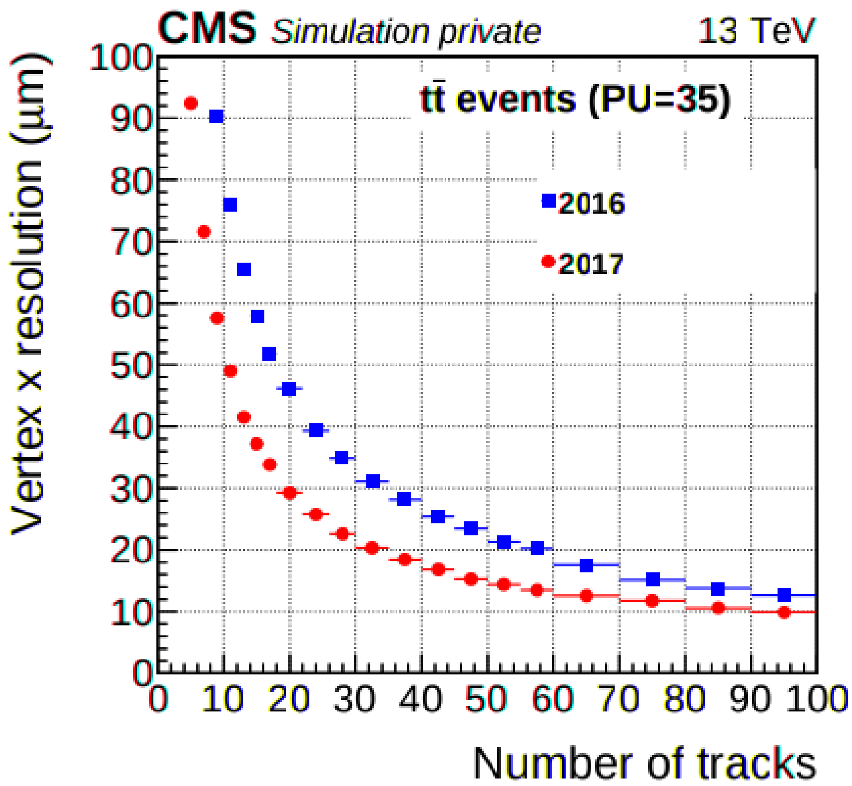


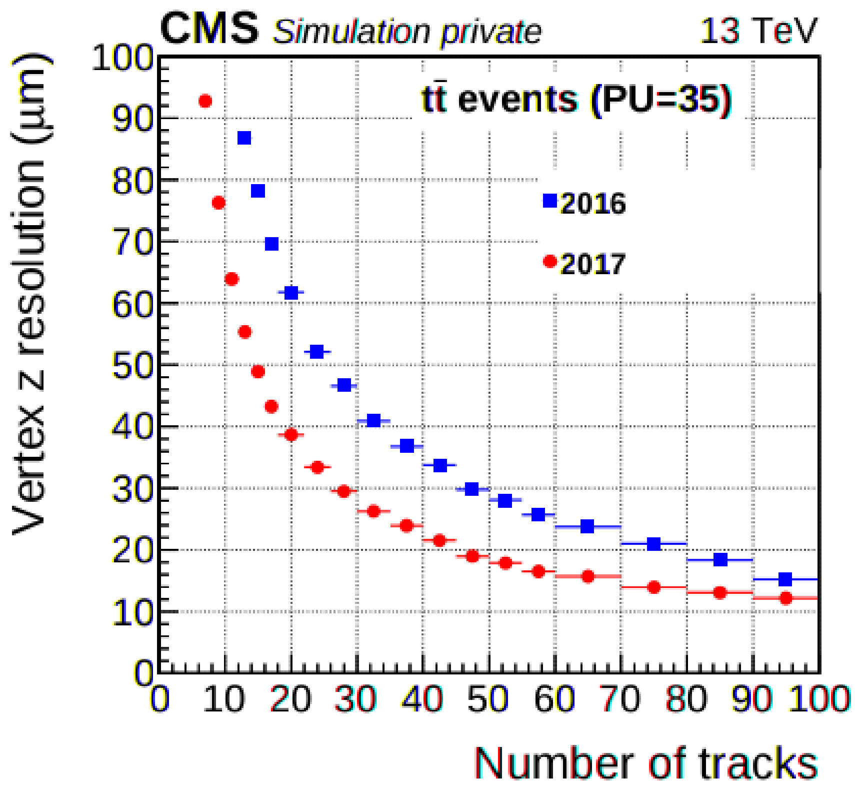
- Set of tracks with p_T > 900 MeV
- In general higher efficiency and lower fake rate

Performance

Vertex resolution:

Matti Kortelainen





- Better resolution (20-40%)

Tracking @ HLT

Strategy:

Mia Tosi

- First version produced and tested.
- Pixel hits used for e.g. pixel tracks and vertices, and seeding.
- Started with same strategy as offline tracking

− Pixel triplets → pixel quadruplets
− Pixel pairs → pixel triplets

CURRENT

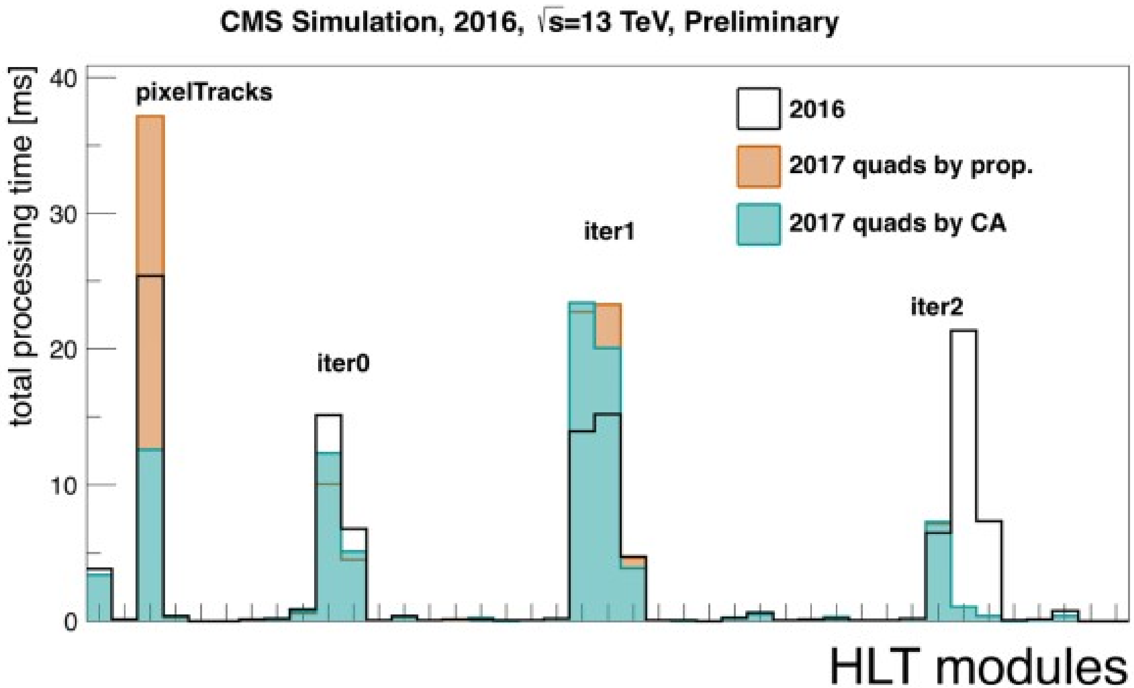
N	Step Name	Seeding	Target Track	Phase-space
	pixelTracks	**pixel triplets**	all	global, constraint to beamspot
0	iter0	pixel tracks	prompt, high p_T	global, constraint to sub set of PV
1	iter1	**pixel triplets**	prompt, lowish p_T	regional, constraint to sub set of PV
2	iter2	**pixel pairs**	high p_T recovery	regional, constraint to sub set of PV

OPTION

N	Step Name	Seeding	Target Track	Phase-space
	pixelTracks	**pixel quadruplets**	all	global, constraint to beamspot
0	iter0	pixel tracks	prompt, high p_T	global, constraint to sub set of PV
1	iter1	**pixel triplets + extension**	prompt, lowish p_T	regional, constraint to sub set of PV
2	iter2	**pixel triplets**	high p_T recovery	regional, constraint to sub set of PV

Using CA @ HLT

- Development ongoing with the end goal of using GPUs for pixel tracking in the HLT in 2020
  - Highly parallelizable algorithm to make use of GPU architecture
  - However, CPU version available for use in 2017 since CMSSW_8_1_0_pre9



Mia Tosi

	pixelTracks	all tracking
2016	25.4 ms	172 ms
2017 quads by propagation	37.1 ms	153 ms
2017 quads by CA	12.6 ms	129 ms

- Performance of different seeding approaches (same condition in slide 12)
- Compare 2016 setup to quadruplet seeding using T*riplet propagation* or *Cellular Automaton*
  → *Triplet propagation* slow in pixel track reconstruction due to overlapping modules
- *CA* better than other options, especially in pixel tracks

HLT pixelTracks performance

Comparison for pixelTracks:

- Performance evaluated w.r.t 2016
- As good or higher efficiency.
- ~2-5x lower fake rate.
- CA shows very good performance

− Especially in forward region

Comparison for iterative tracking:

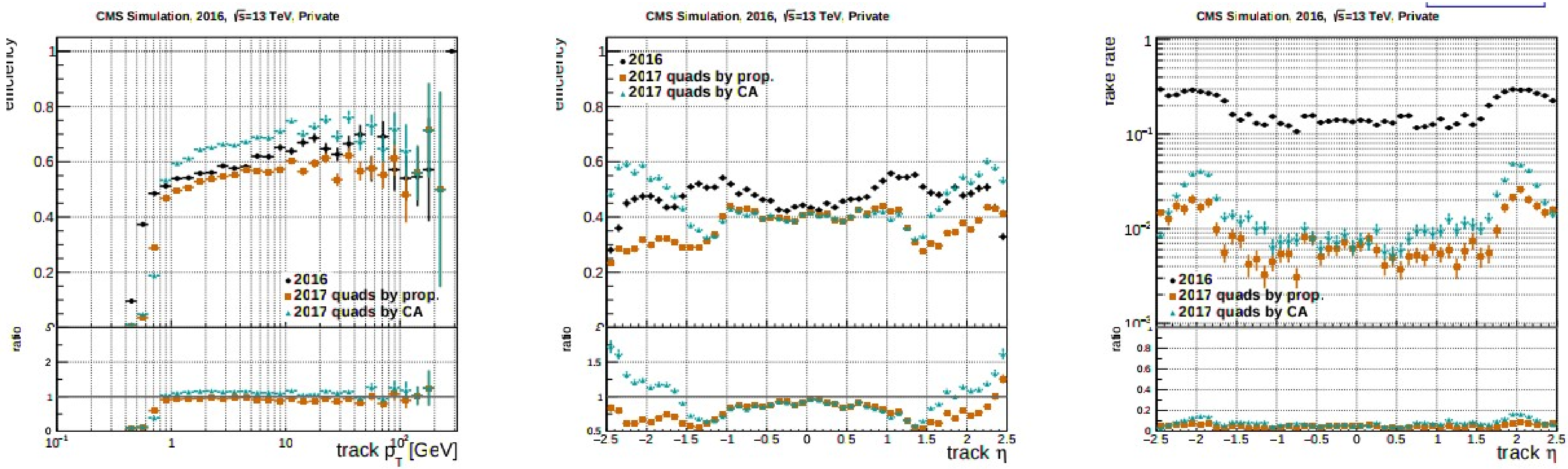
- Performance evaluated w.r.t 2016
- No clear difference between the two quadruplet algorithms.
- Timing with two quadruplet seeding algorithms:

− “quads by prop”: 10 % faster than 2016

− “quads by CA”: 25 % faster than 2016

Performance for pixelTracks

Mia Tosi



- CA quadruplets the most promising approach for HLT.
- Final working points (and seeding of iter1 and iter2) are still to be tuned.

− Higher efficiency and/or lower fake rate have a cost in time

- **Main developments are starting now!**
- Exploit the CA even in the regional approach which characterizes the HLT.

To do's list

- Parametrization of multiple scattering for seeding
− Possibly not needed if we go with cellular automaton seeding

- Re-train track selection MVA for Phase1
− On-going

- Continue development and tuning of cellular automaton quadruplets
− Especially for HLT, but also for offline

- Continue development and testing of tracking for HLT
− First optimized version: ~now.

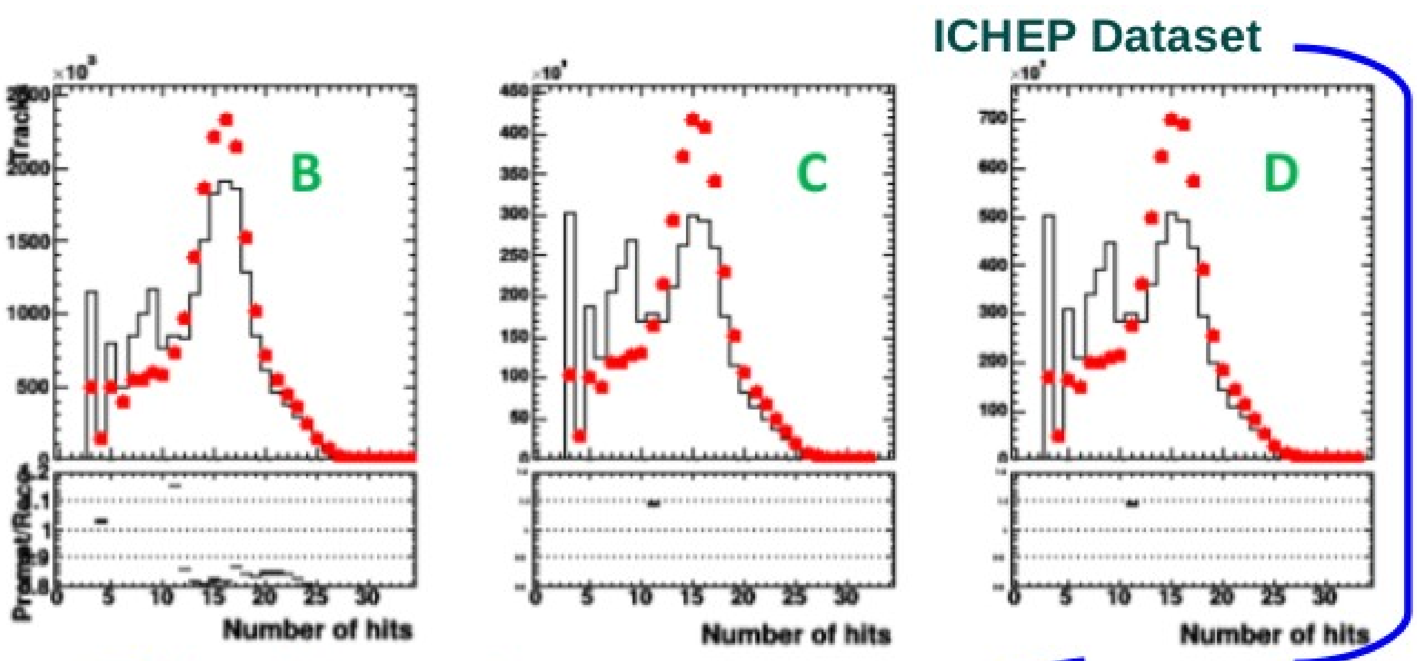
- Develop a proper treatment of module overlaps for FPix
− Even if CA pixel tracking without the treatment is already faster than 2016 pixel tracking

- Re-tune primary vertex reconstruction parameters

- Re-evaluate the performance with
− Pixel alignment scenarios (realistic, “day-0”, ...)
− Detector failure scenarios
− High pileup (50, possibly 70)

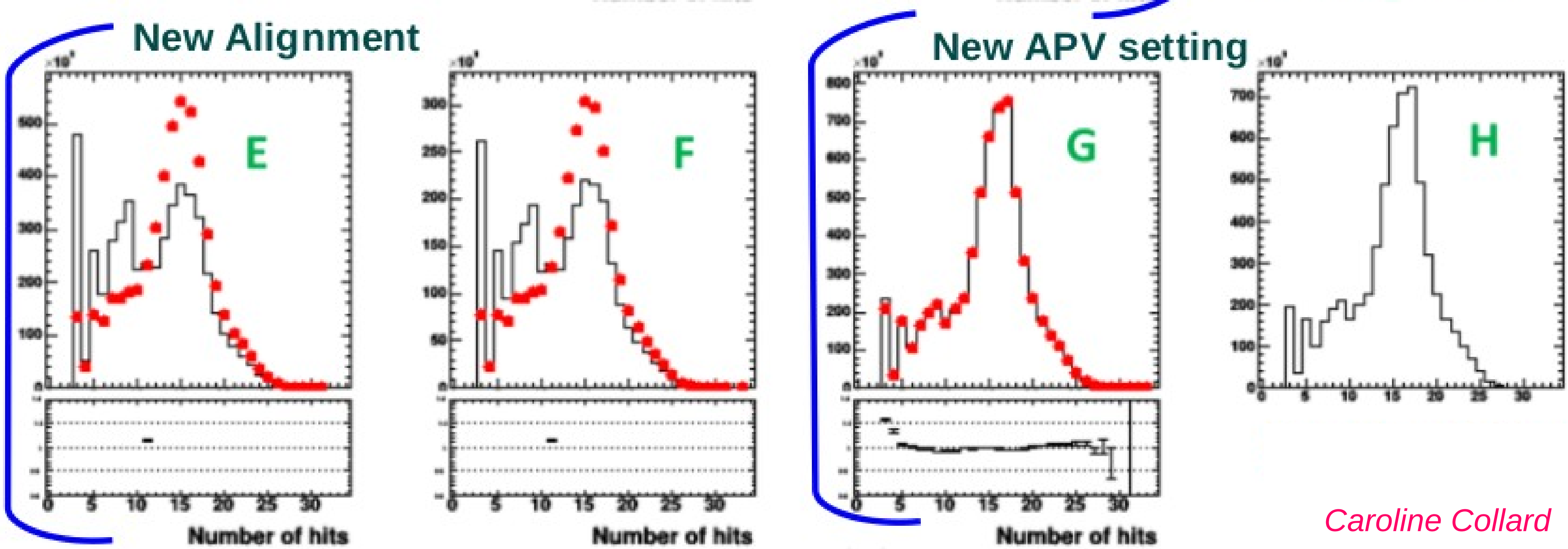
b-tagging

First Look at Re-reco data



Comparison:

- Looked at 23Sep2016 re-processed data
- Comparison of **23SepRe-reco** (with tracking and BTV HIP mitigation) vs **Prompt-reco** (with BTV HIP mitigation)
- Clear positive effect from the TRK mitigation
- Recover for the APV setting problem



First Look at Re-reco data

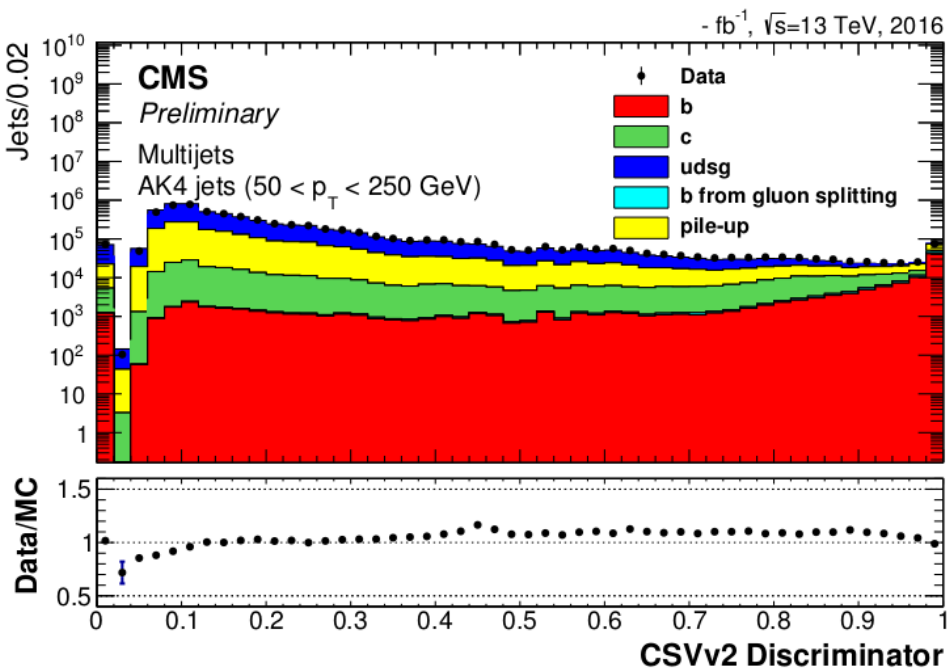
<u>**BTV group main activities:**</u>

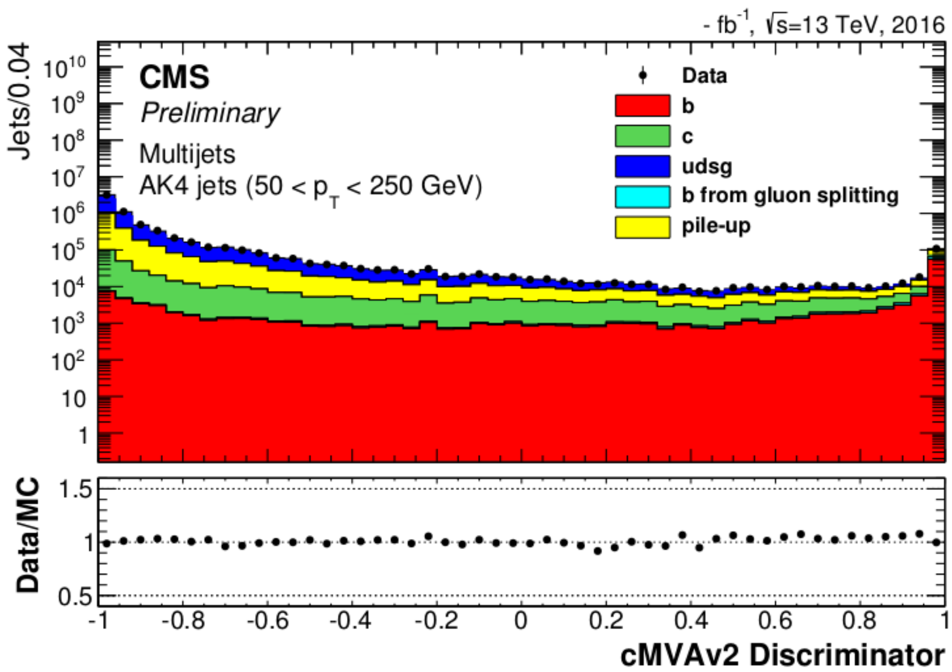
Brieuc François

- b-tagging for ak4 jets and subjets of ak8 jets
- c-tagging for ak4 jets (BTV-16-001) new in 2016
- double-b tagging for ak8 jets (BTV-15-002) new in 2016

<u>**Comparison:**</u>

- Moriond 2017 BTV campaign and commissioning on Re-reco data/Summer16 MC started, being discussed right now at the BTV meeting: https://indico.cern.ch/event/593959/
- Good data/MC agreement, no bad surprised (here examples from the QCD inclusive channel)

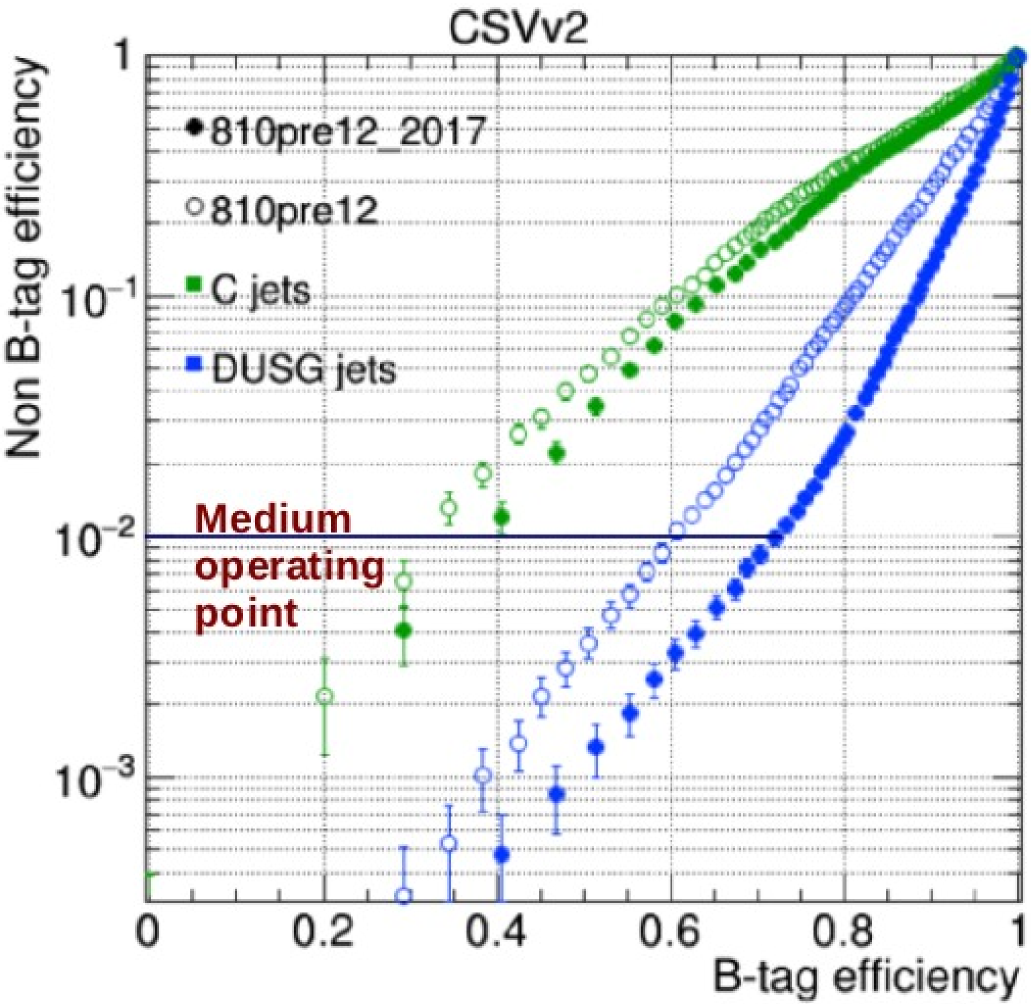


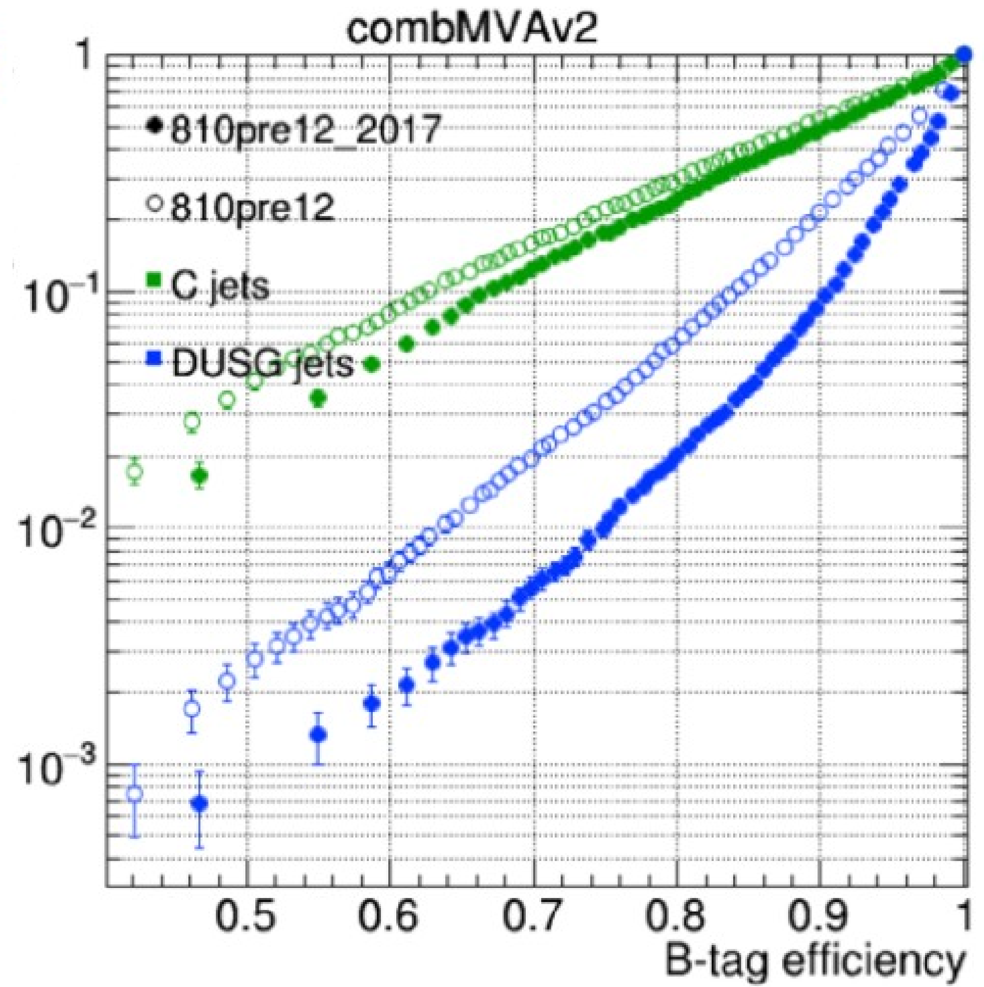


2017 studies

- Large performance **improvement** with Phase-I tracker
- b-tag algorithms used for these studies are **not optimized** or **retrained** for Phase-I conditions: larger MC statistic needed (January)
- Gain of 10% absolute efficiency at medium operating point (1% mistag rate)
- Software developments are needed to fully take advantage from the new pixel detector.
- Taking good data is important, because the offline training will be used also online.

Petra Van Mulders





On-going developments

Track selection optimization:

Mauro Verzetti

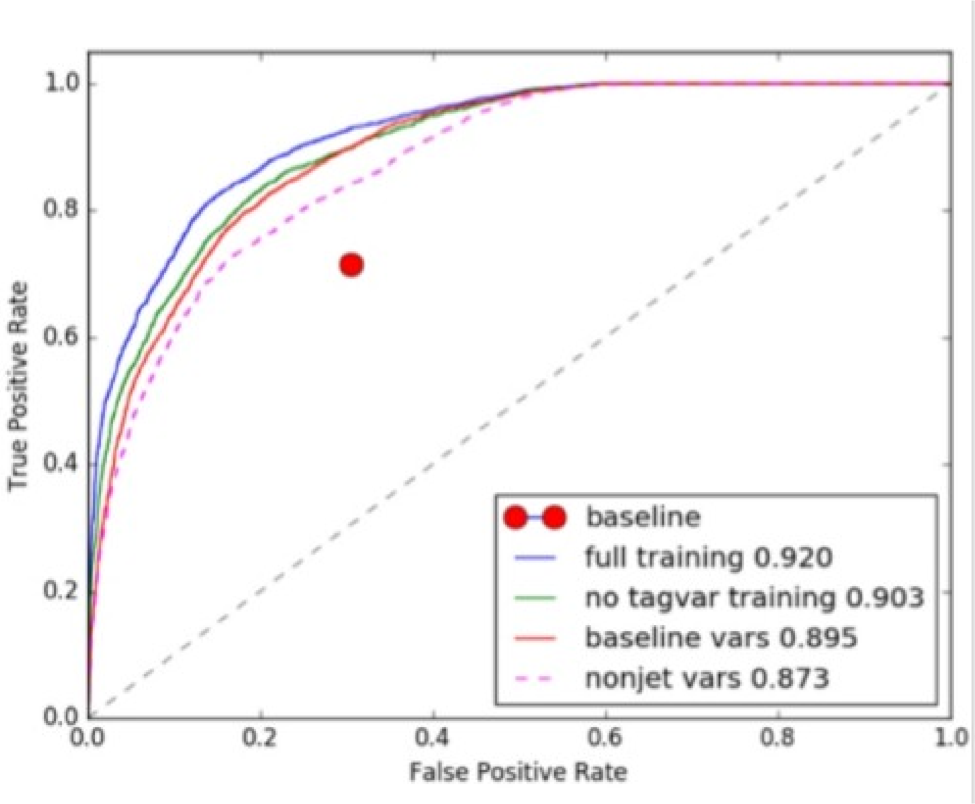
- CSV does not use all the tracks in the jet.

- Used SciKitLearn to train the track selection using four sets of variables.
  − Baseline selection: Only selected tracks are provided as input to CSV
  − Custom variables such as number of lost hits, inactive hits etc + track variables which are used in CSV

- Area under ROC curve is used as figure of merit.

- Initial tests show great improvement w.r.t. baseline selection.

- Machinery is in place for optimization of track selection for Phase1.

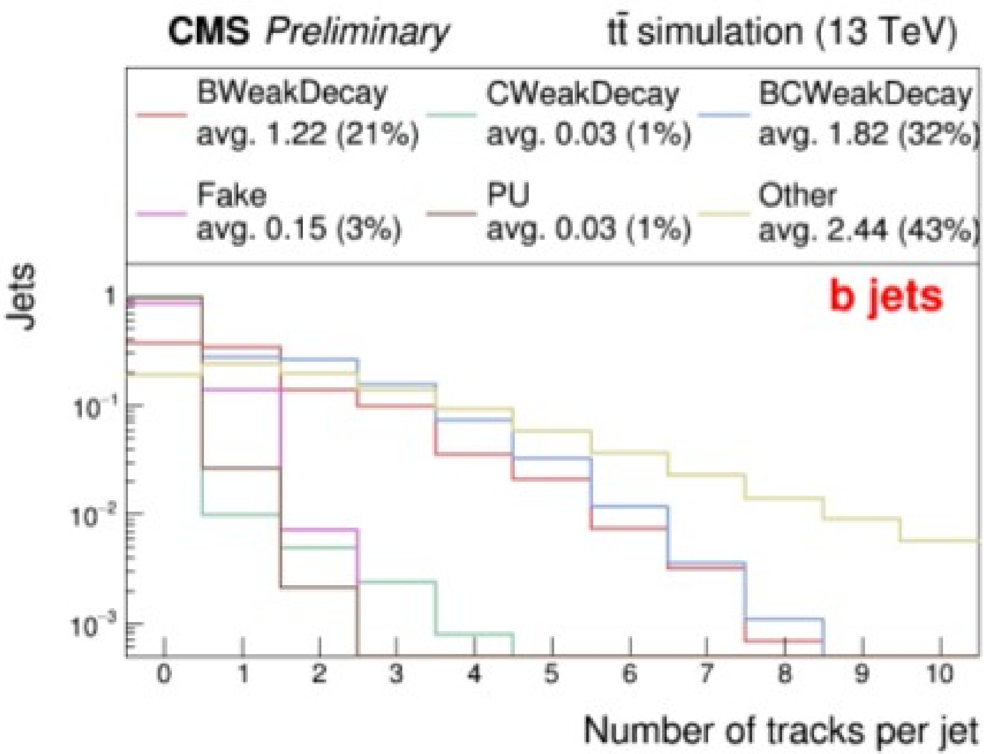


On-going developments

Tracking Truth studies:

Seth Moortgat

- b- and c-tagging rely heavily on the reconstruction of tracks and secondary vertices coming from decays of bottom and charm hadrons.
- Information on the track and vertex history in simulations allows us to monitor the reconstruction and properties of these specific tracks and vertices.
- Plan to monitor reconstructed tracks and vertices from B and C hadron decays as well as the differences between their reconstructed properties and those at truth level as part of the release validation.
- This well be of great help to identify the source of possible (b- and c-tagging) performance losses in new software releases.



On-going developments

DeepFlavour based on DeepLearning:

- For DeepNN used same input variables as CSV (Deep CSV) + few additional variables as in cMVAv2 (Deep cMVA).

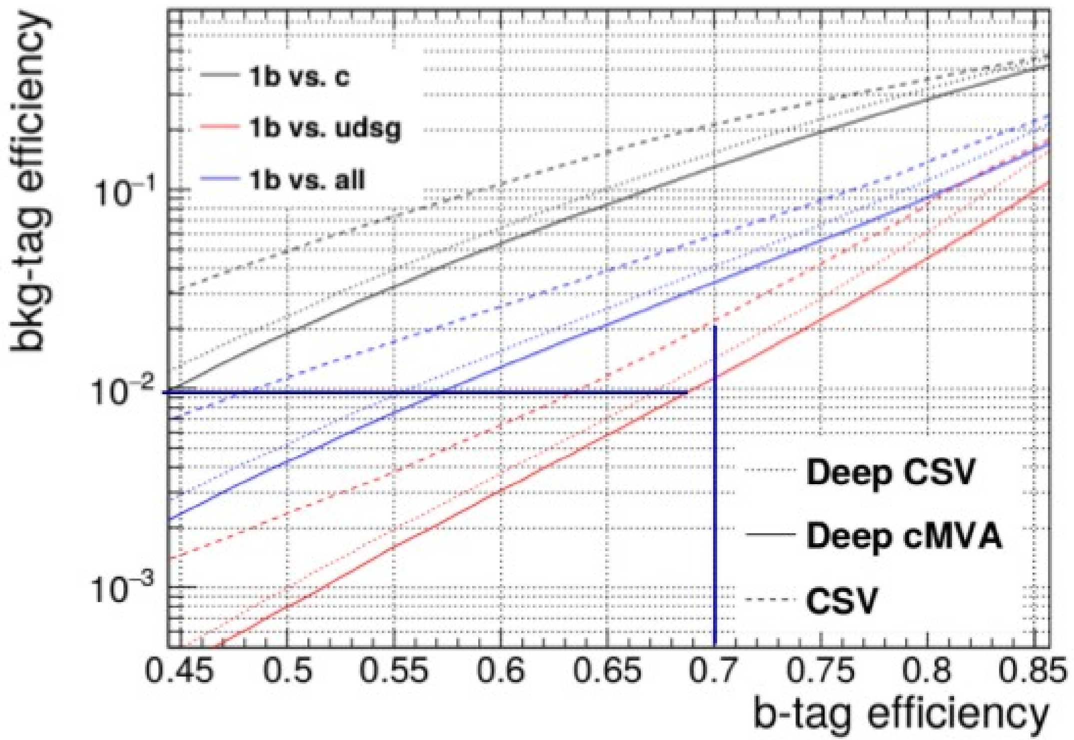
- Gain of ~5-7% absolute efficiency at medium operating point (1% mistag rate) for DeepCSV compared to CSV.

- Trying to commission DeepCSV for Moriond
– will provide customized recipe for analyzers
– a couple of analyses expressed interest to test it

- There are still many open possibilities to explore.

- DeepNN can easily accommodate a large amount of variables.

Marcus Stoye



Conclusions

2016 summary:

- First look at the reprocessed data was presented for both tracking and b-tagging

– No odd surprise with the Re-reco data
− Measure Scale Factors based on the Re-reco and the new MC which includes realistic pixel dynamic inefficiency

2017 readiness:

- We have already a solid tracking for 2017
- Compared to tracking in 2016:

− Significantly better physics performance
− Slower by ~ 10 % (in offline)

- Configuration kept stable for 810 since pre7, but have some improvements in pipeline:

− Cellular automaton quadruplet seeding
− Proper treatment of (FPix) module overlaps in seeding

- HLT developments:

− Tracking for HLT is also on its way
− b-tagging working on online integration of offline developments
– Preparing for 2017 online implementation of FastPV

- b-tagging developments on many front:

− Higher statistics MC samples needed to tune b-tag selections and retrain the taggers
− Exploring Deep Learning and trying to commission DeepCSV for Moriond
− Including tracking truth studies and optimize track selection

Thank you for the attention