Rising min-pt cut

Rising the min-pt cut

- The idea is to look at how rising the min-pt cut can help with the **dataflow**.
  - b-tagging / tau-id performance can be evaluated by applying an analysis-level pt cut.
- I am considering 3 <u>analysis-level cuts</u>:
  - 1.0 GeV; 1.5 GeV; 2.0 GeV
- These correspond to these <u>training cuts</u>:
  - 0.8 GeV; 1.3 GeV; 1.8 GeV
- For reasons of speed, I reduced the amount of patt-from-const training for rised-pt banks in proportion with curvature phase space.

Bank configuration

- To study realistic dataflow numbers, I used one of our option-B configurations:
  - 64M 8L patterns with ss=16
  - 64M 4L patterns with ss=15x20 (pix) and 150I150
    - (this is one of the most optimal split arch points)
- Full set of split-arch banks was generated for each pt point on the grid.
- Efficiency was measured on 3E34 WH events (hard scatter only)
- With high-pt banks, we can realize dataflow savings by having a smaller overall number of patterns due to reduced curvature phase space

Results from min-pt banks

Scaled with curvature phase space
Default = 64M at 8L and 64M at 4L

Split arch efficiency over full eta
This includes a dip in transition region.

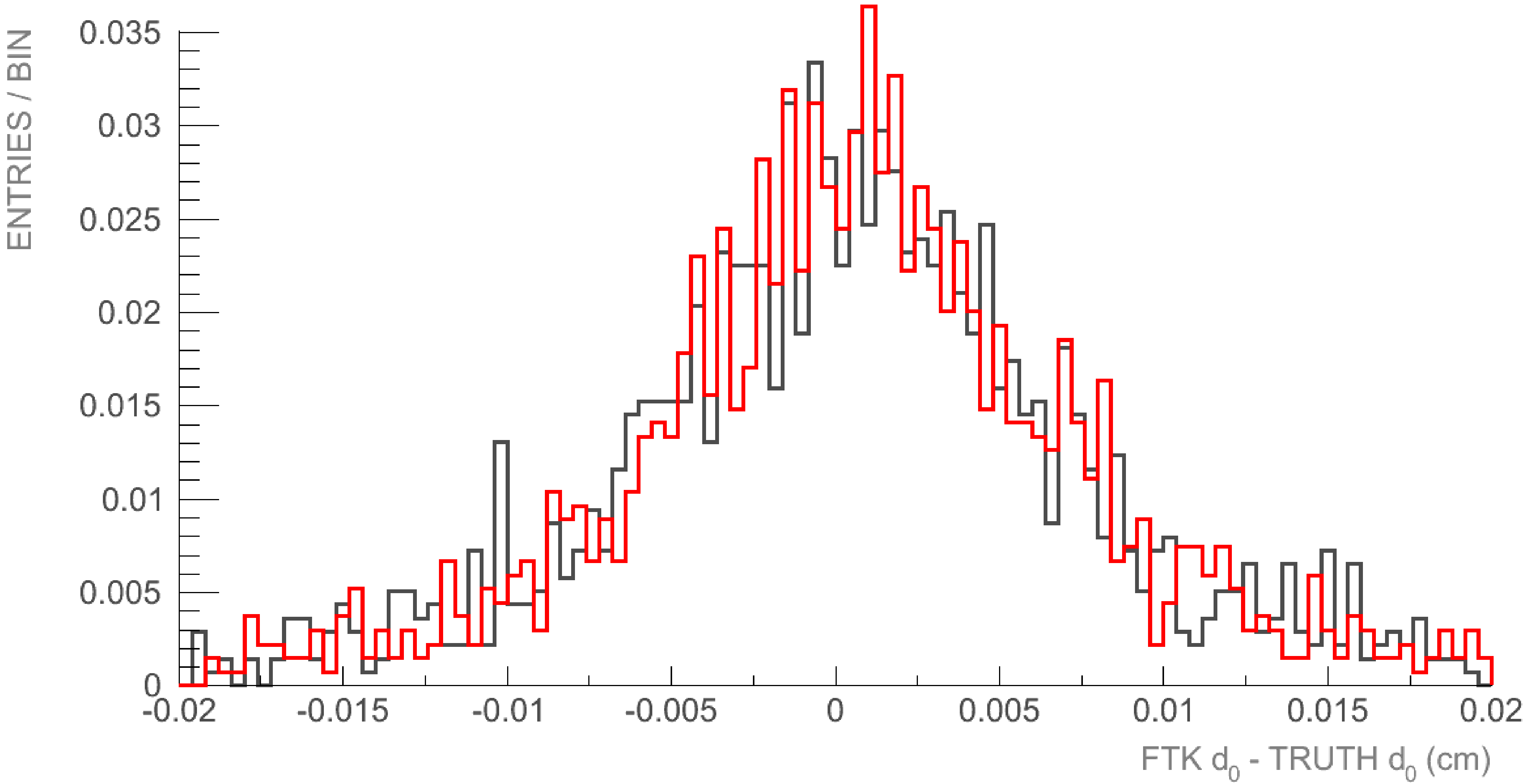
Bank min-pt	Ana min-pt	npatts	eff	nroads8L	nfits8L	ntracks8L	nroads4L	nfits4L
(in MeV)								
800	1000	64M	64.26%	52k	250k	3k	58k	115k
800	1500	64M	66.10%					
800	2000	64M	66.30%					
1300	1500	39M	64.70%	30k	147k	1.4k	23k	42k
1800	2000	28M	63.70%	22k	103k	0.9k	14k	24k

As expected, everything scales with the number of patterns.
The efficiency for tracks above 1.5 GeV is a little lower with the min-pt trained banks compared to the default full-pt banks: 64.7% vs 66.1% (Δ=1.4%). However, we could probably recover some of it by (1) training 1.5 GeV bank with more statistics, and (2) slightly increasing npatts at either 8L or 4L stage (or both).

d0 resolution for all tracks with pt>1.5 GeV for the following two banks:

*** default bank trained with pt>800MeV**

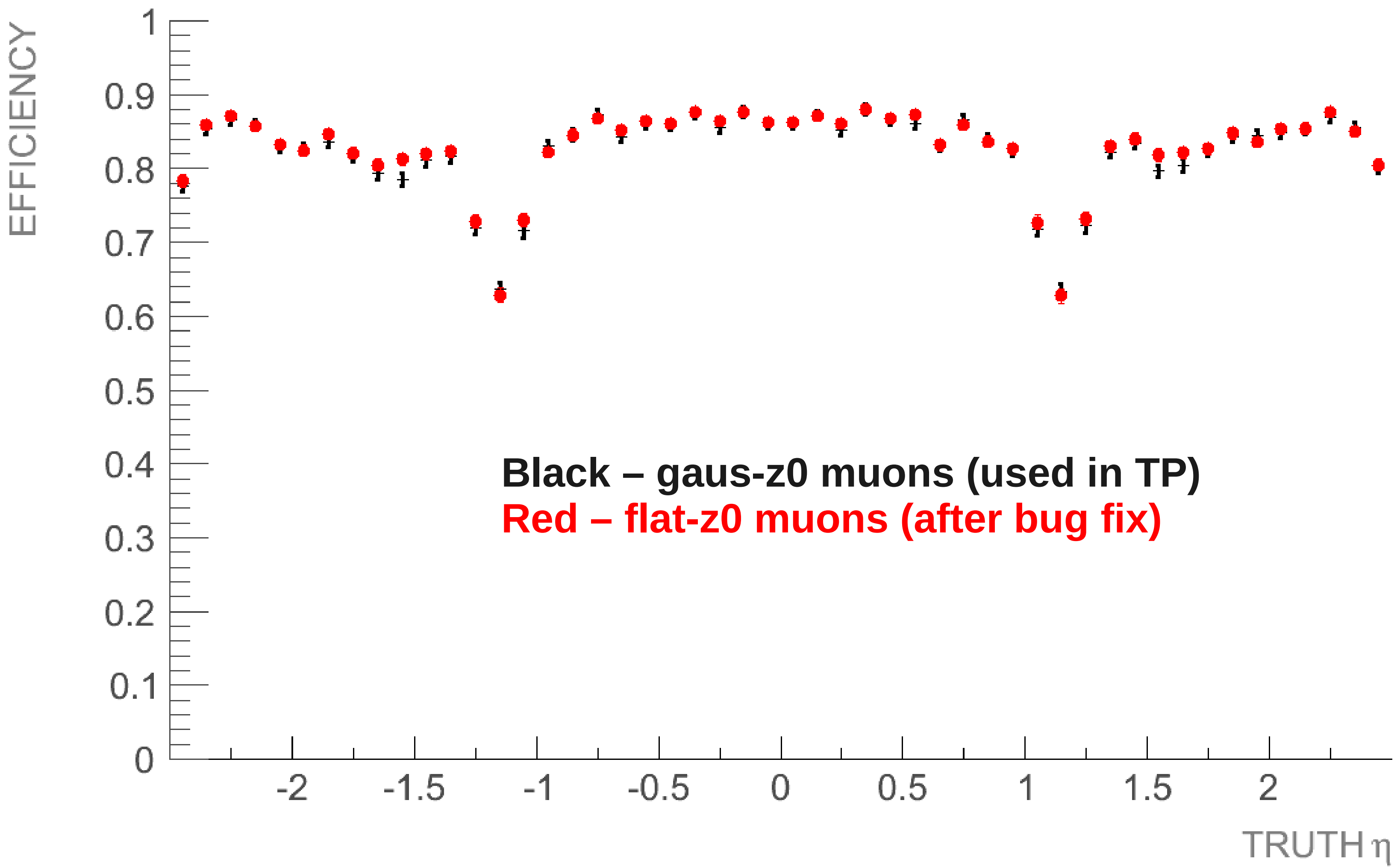
*** new bank trained with pt>1300 MeV**



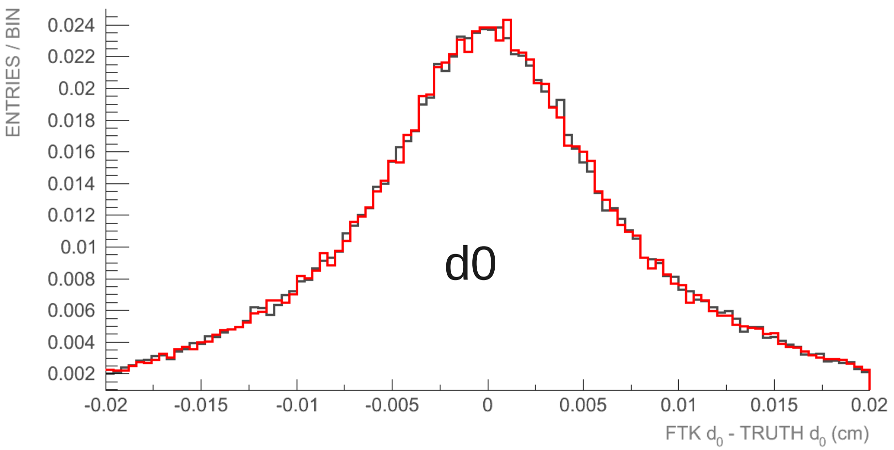
Sorry about low statistics. However, it seems that the resolution didn't improve with the high-pt trained bank. In other words, low-pt tracks don't pollute constants in high-pt sectors. If anyone is interested, I can make a similar plot for single muons with high statistics.

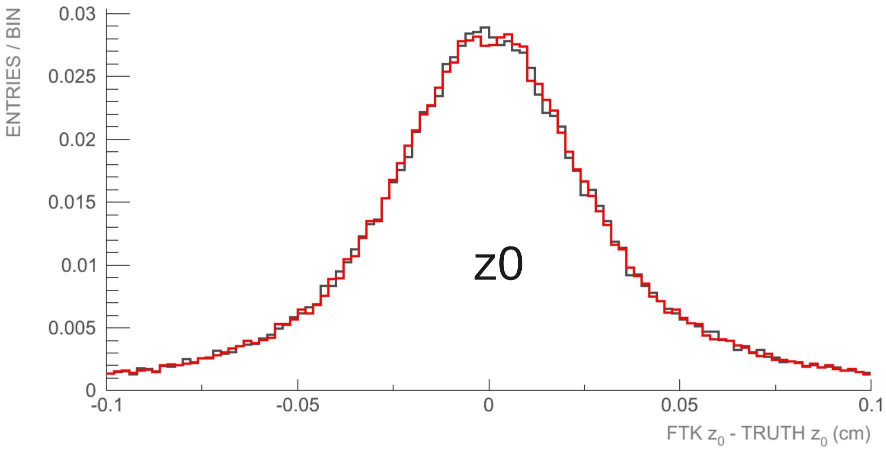
Validation of new flat-z0 muons and bank production machinery

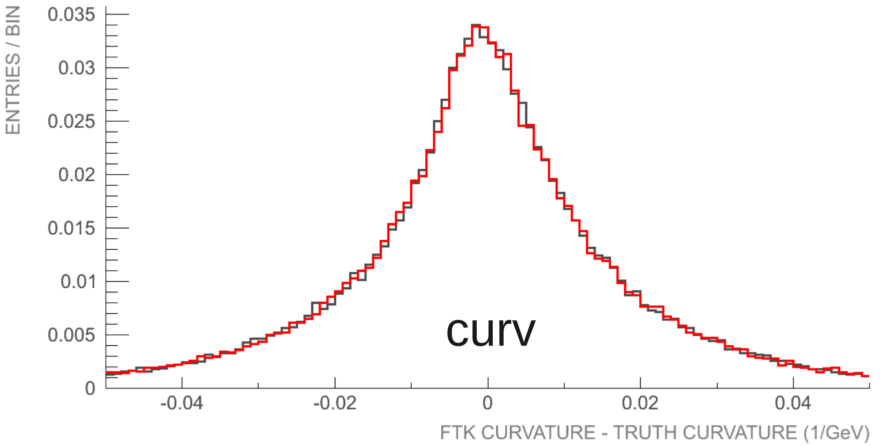
11L 50x64x144 efficiency (computed for flat-z0 muons)



11L 50x64x144 efficiency (computed for flat-z0 muons)







From the plots, it appears that there is no difference in using flat-z0 or gaus-z0 muons at the training stage.

Processing on the grid

Summary of grid experience

- Two weeks ago, I committed a rewrite of grid submission framework
- A number of enhancements were committed since then (mostly smarter error detection)
- I extensively exercised grid machinery since then (mostly on ANALY_MWT2)
  - 0.5% - 1.0% job failure rate
  - Smart error detection properly marks failed jobs as FAILED and allows to retry them
  - Bank production has been VERY stable and fast

Plan for large-scale processing

- We have about 50k Whuu and Whbb (each)
- For TP split arch, each node can run 130 3E34 events (running ALL subregions) in ~7 hours. 8 nodes will simulate the whole detector.
- Each user can submit (8 regions)*(50 input groups of 130 events each) = 400 jobs. They can all run in parallel and process 6500 events
- So, 8 users submitting in parallel should be able to process all 50k events overnights
  - In reality, it will be a factor of 2-3 longer due to finite CPU resources on ANALY_MWT2 (~1600 nodes)