Update on FTKSim

F. Crescioli, M. Dell'Orso, P. Giannetti, G. Punzi, **G. Volpi**

Summary

- Study about FTKSim efficiency.
  - Bug search
  - Improvements on Sectors and Patterns efficency
  - Study on single track samples
- FTKSim memory optimization:
  - Smaller number of sectors
  - Smaller number of patterns

Bug Search

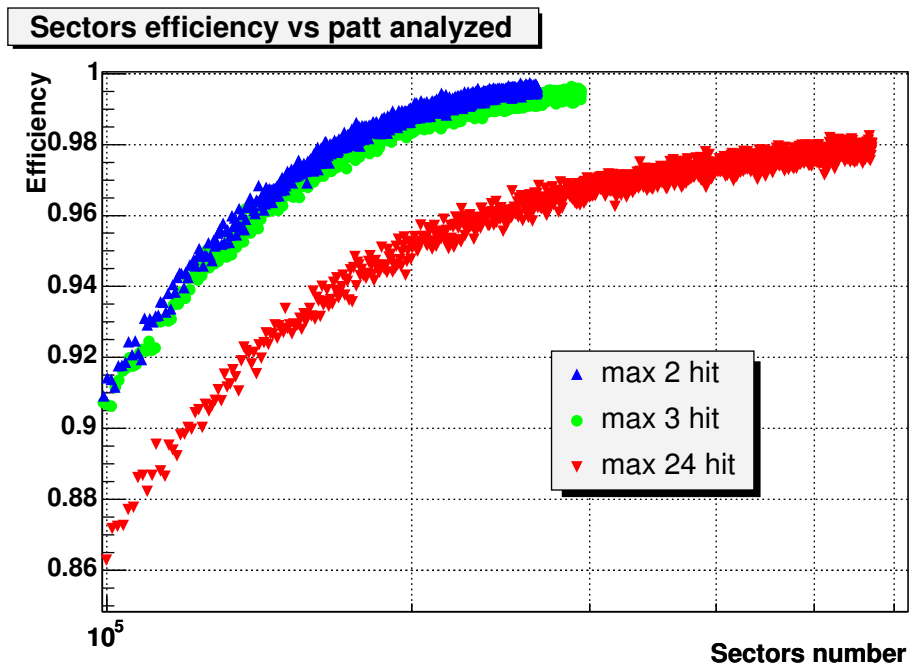
- An important bug in pattern load: only an half of patterns bank was loaded.
- Some other minor bugs, not important to explain the low efficiency

Efficiency factors

To maximize the efficiency we generate a new set of patterns and constants. Now in the training use only tracks with max 2 hits in a plane.

The plot show the efficiency vs number of sectors for the various criteria.

The sectors are also filtered after the constants generation to exclude many inefficient sectors.



New sectorization have an efficiency of $\varepsilon_S = \mathbf{98.6\%}$ with smaller number of sectors: $\mathbf{157838}$ vs ~ 340000.

Pattern banks efficiency is $\boldsymbol{\varepsilon_{\mathbf{P}} = 96.1\%}$. We exclude the patteren not within a sector so we reduce the their number from $1,500,000$ to $1,000,000$.

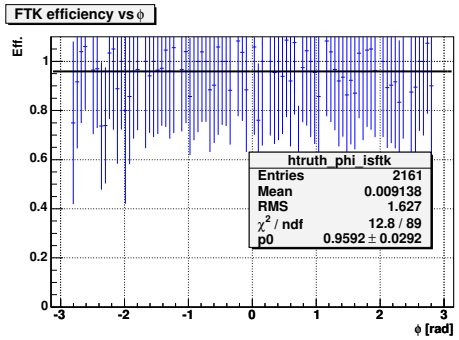
The probability for a track to have 6 hits in all 6 detector plane is $\boldsymbol{\varepsilon_{6/6} = 79\%}$ (evaluated from single muon track sample).

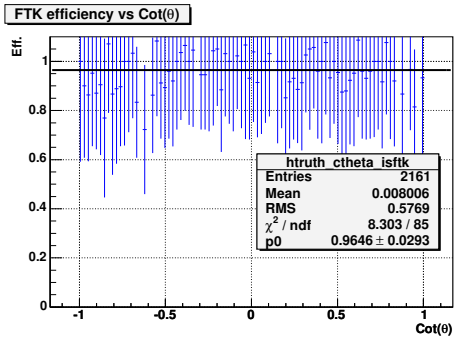
Single Track sample (filtered)

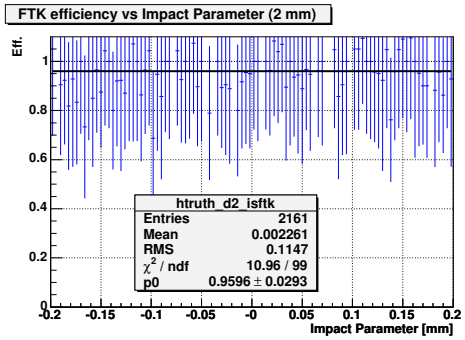
We filtered a single track data sample, not used for pattern or sectors generation, and we extract only the events that match Pattgen criteria: **6 hit in 6 layers**, **max 2 hit** in each plane.

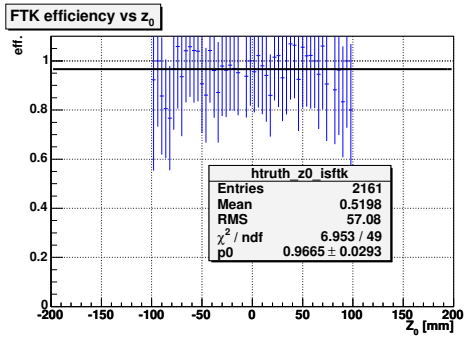
We measure an average efficiency of $\varepsilon_{\mathbf{S}} \cdot \varepsilon_{\mathbf{P}} = \mathbf{96\%}$ that not depends on various track parameters, like aspected.

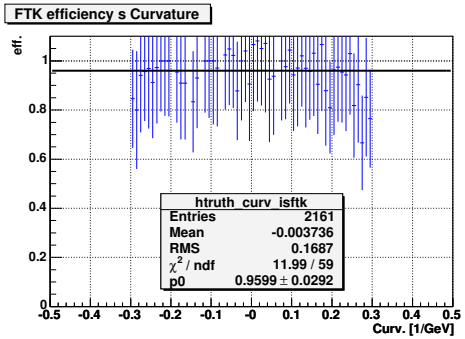
Update on FTKSim

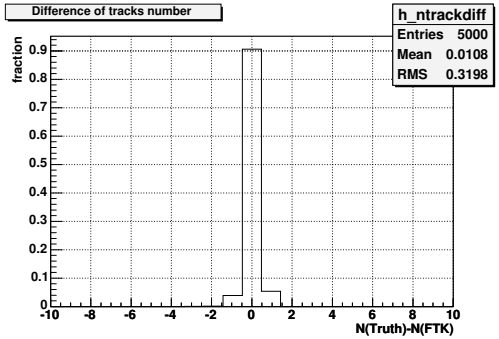










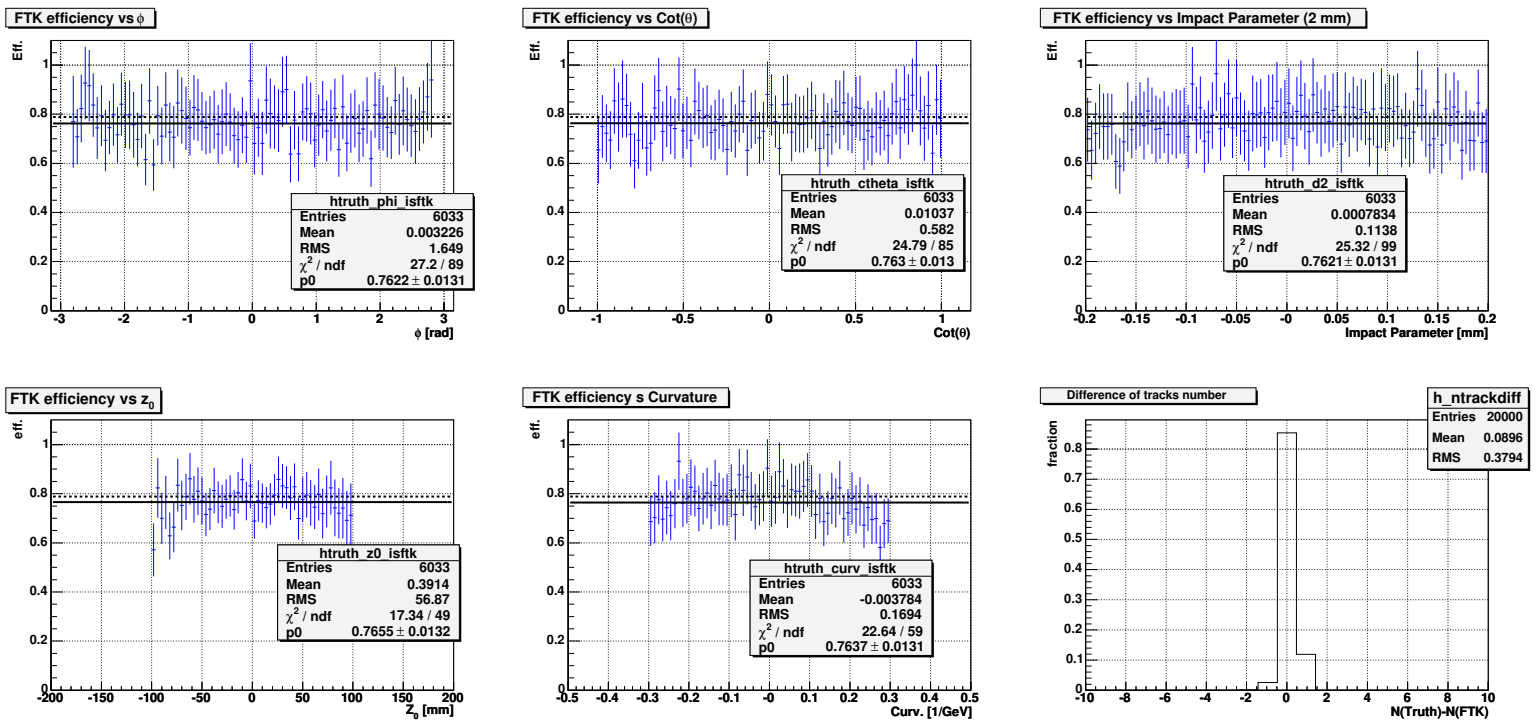


Single Track sample (not filtered)

We remade the previous exesercises for a single track, without excluding any events.

We expect an efficiency equal to $\varepsilon_{\mathbf{S}} \cdot \varepsilon_{\mathbf{P}} \cdot \varepsilon_{\mathbf{6/6}} = \mathbf{75}\%$. The results confirm this expectation.

Update on FTKSim



For the single tracks sample the main reason of inefficiency is the majority logic.

$B_s \to \mu\mu$ sample

We repeat the study of efficiency on tracks from $B_s \to \mu\mu$ events, with an average efficiency of $\mathbf{60\%} < \boldsymbol{\varepsilon_{\mathbf{S}} \cdot \varepsilon_{\mathbf{P}} \cdot \varepsilon_{\mathbf{6/6}}}$

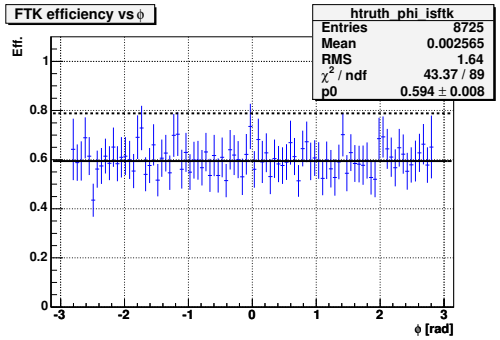
For this sample respect previous talk we increase the average efficiency from 50% to 60%. Problably due of the bug in pattern load.

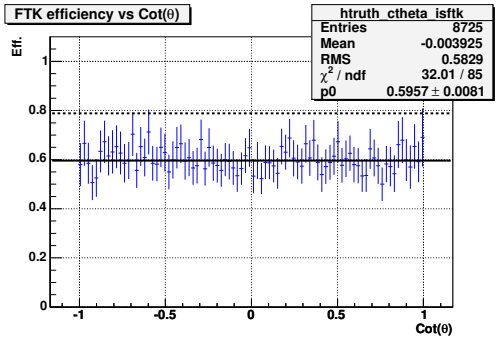
But in this sample there is not only μ so we can expect that for other particles the effect of the 6 over 6 make a factor smaller than .8

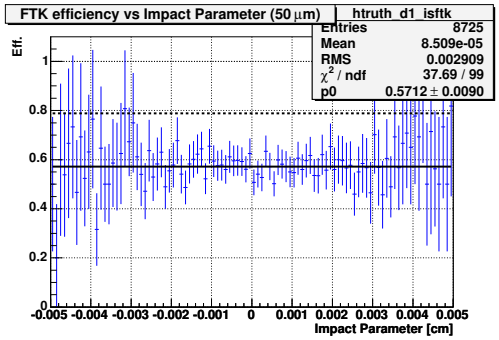
We looked also in single bad reconstructed events but inefficiency don't seems function of such special value of tracks' parameters.

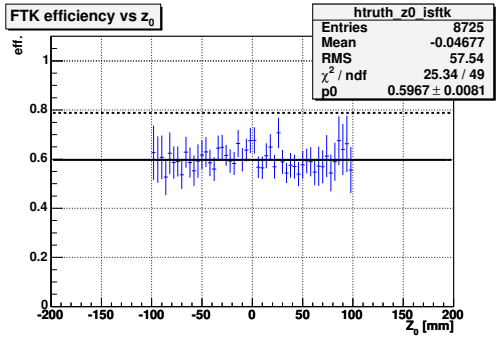
Need more study.

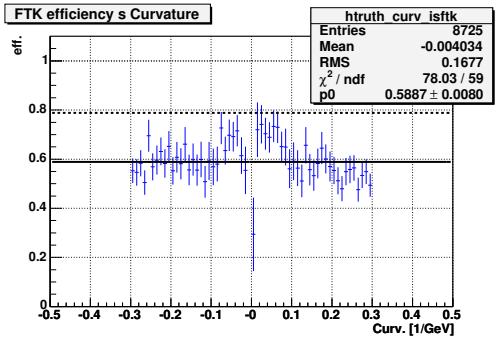
Update on FTKSim

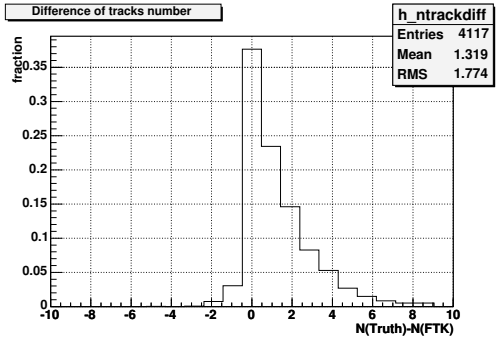












Memory optimization

In the previous generations of constants and patterns we overexistamete the number of sectors and patterns for 2 reasons:

1. Many sectors was very inefficient and did not have good linear fit constants.
2. Some patterns are out of a sector and during recostruction don't generate real tracks.

We reduce the sectors to # 160,000 and pattern to # 1M.

We are studing other optimization to reduce the ammount of memory (RAM+SWAP) required by FTKSim.

Conclusions

- We have an efficency on $\mu \sim 76\%$, explained by:
  - Sectors efficiency
  - Patterns efficiency
  - Probability of 6/6
- The smaller efficiency on B_s samples can be explained by a smaller probability of 6/6, to verify this hypotesis we can:
  - Verify the $\varepsilon_{6/6}$ for other particle, i.e. π
  - Implement Majority Logic and verify the effect on this type of events.

Future plan in Giovanni's talk.