



Track Merging Status

B. Oberhof
LNF-INFN Frascati, Italy

Bi-weekly Tracking Meeting
10th July 2015



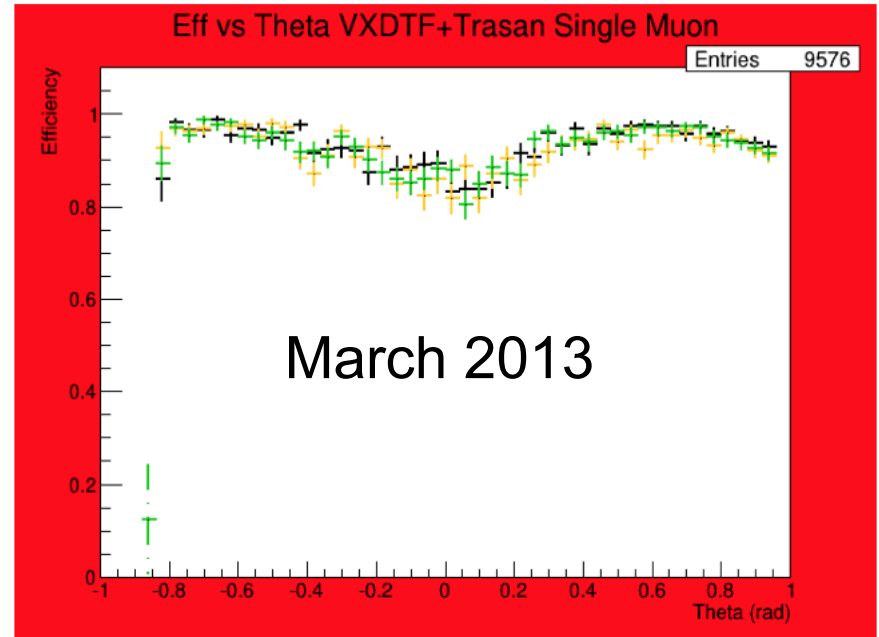
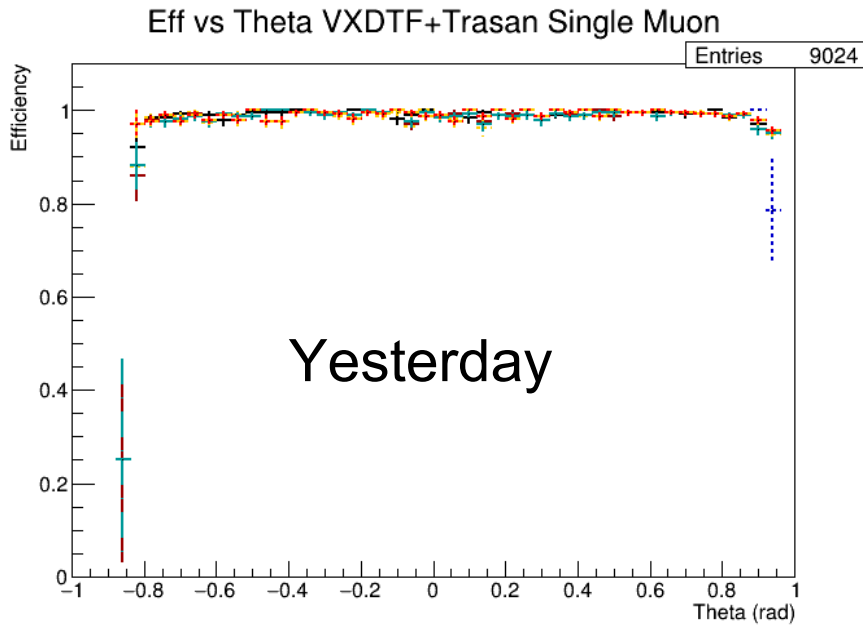
What's new

- Last work mainly oriented to the increase of efficiency
- Recovery mode implemented → if $\text{Chi}^2 > \text{Chi}^2_{\min}$ we check how “far” the 2 tracks are, if $d < d_{\max}$ (module parameters) and one of the tracks is not matched to any other they are merged
- To avoid efficiency decrease due to multiple TrkCand for same MCParticle in current code we require no more than 1 merged track per MCCandidate → loopers lost (?)
- Started tests with beam bkg mixed in
- Validation plots are available on the web page (mostly scans as $f(p_T \text{ and } \theta)$) for a number of TF and generator combinations:
 - TruthTF+TruthTF, EvtGen & single muons
 - VXDTF+Trasan, EvtGen & single muons
 - Multiple muons + beam bkg
- More scripts have been prepared and will be added soon
- Plots using pattern reco look much better w.r.t. the past



Single particle

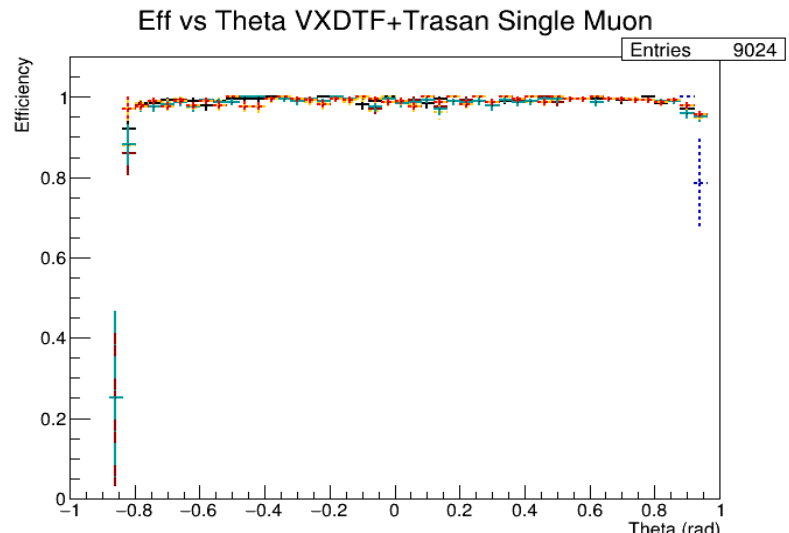
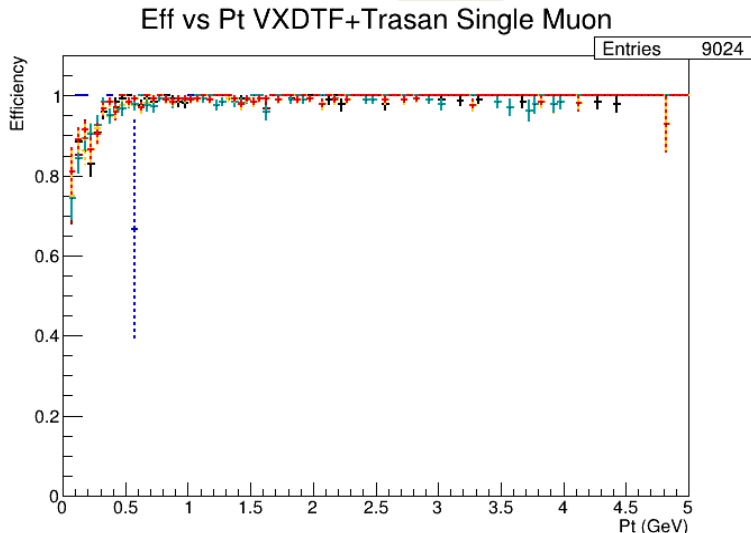
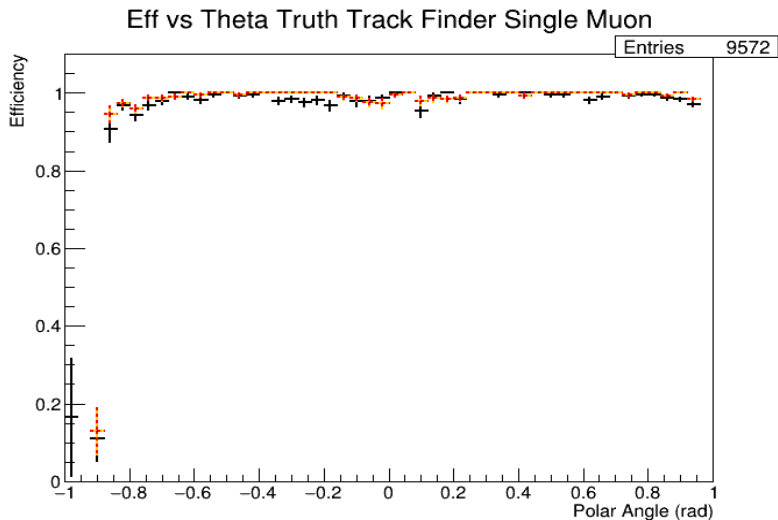
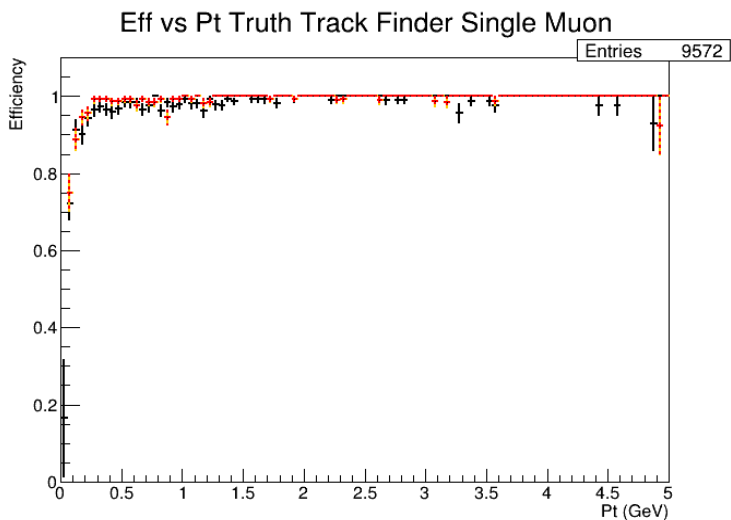
- To avoid possible factorization basic tests are performed using single muon events $p=[0, 5]$ GeV or $p=[0, 1]$ GeV and $15 < \theta < 150$
- Default geometry & simulation
- Two TF configurations: Truth+Truth, VXDTF+Trasan



$$\text{Efficiency} = \frac{\text{Correctly Merged}}{\text{Truly Matched}} \text{ Tracks}$$



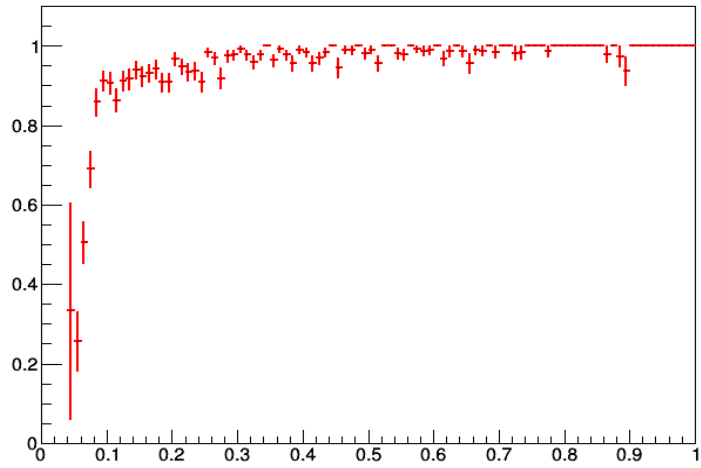
Single particle



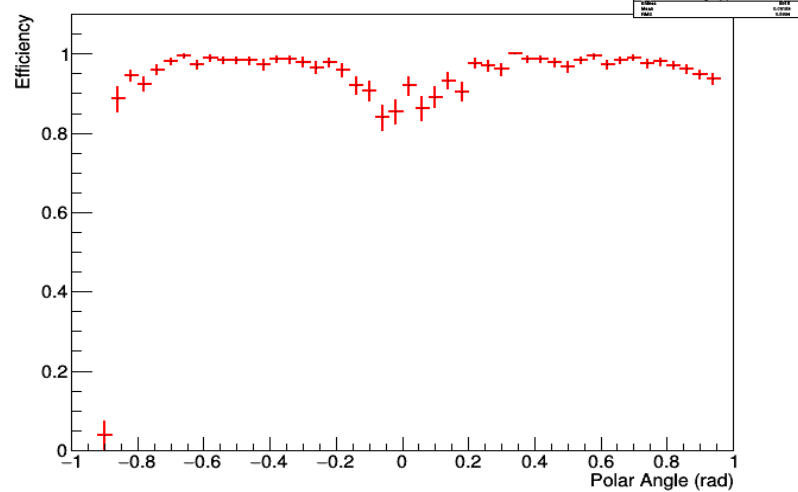


Low p_T range, $[0, 1]GeV$ muons

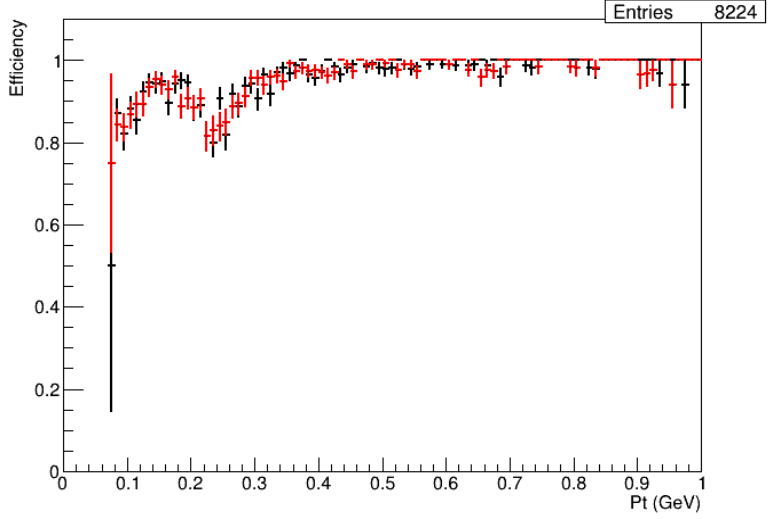
Eff vs Pt Truth Track Finder Single Muon



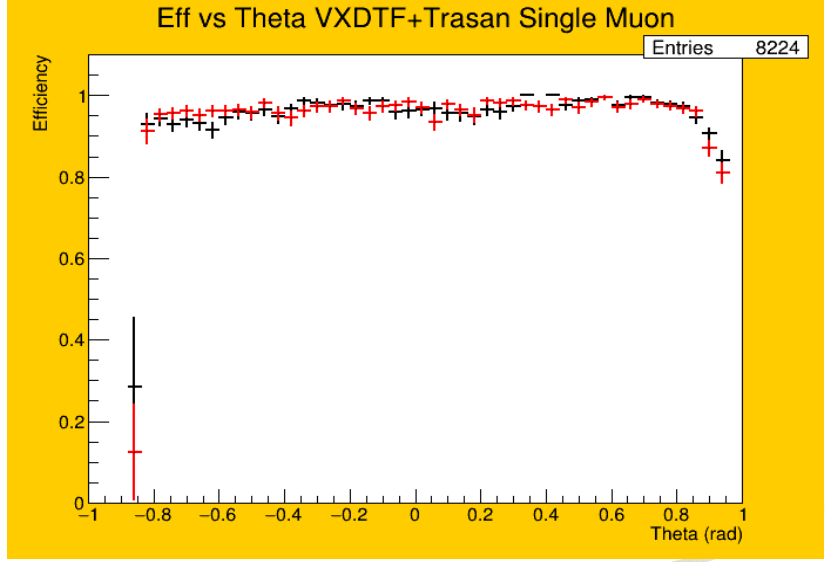
Eff vs Theta Truth Track Finder Single Muon



Eff vs Pt VXDTF+Trasan Single Muon



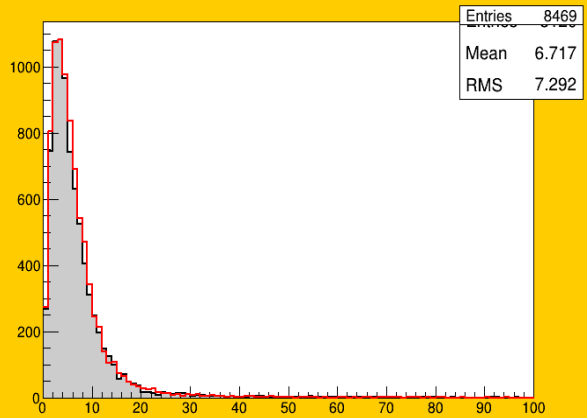
Eff vs Theta VXDTF+Trasan Single Muon



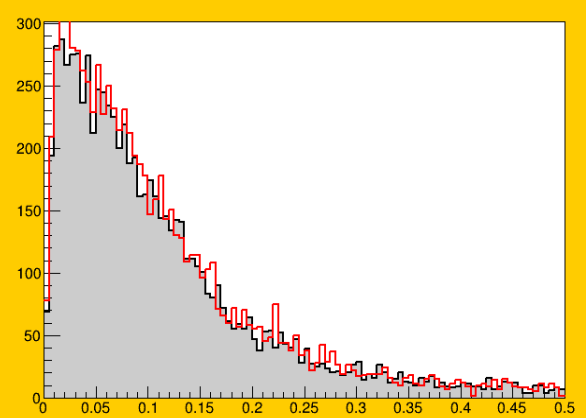


Residuals, χ^2

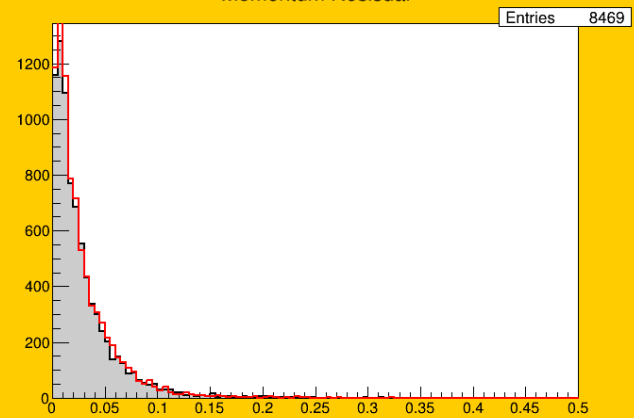
Chi2



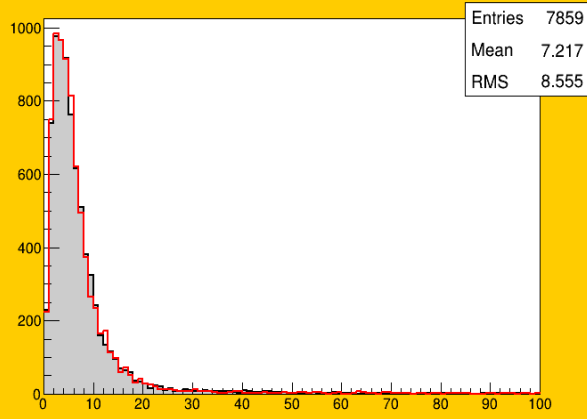
Position Residual



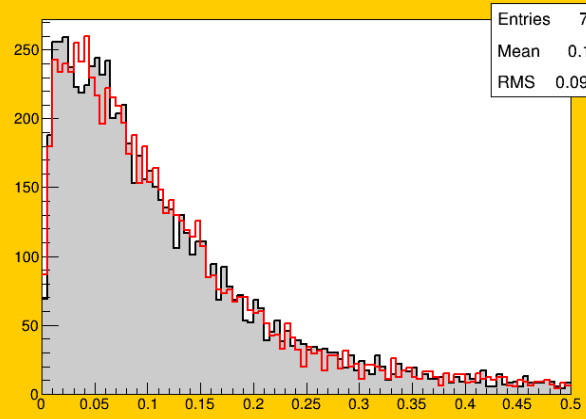
Momentum Residual



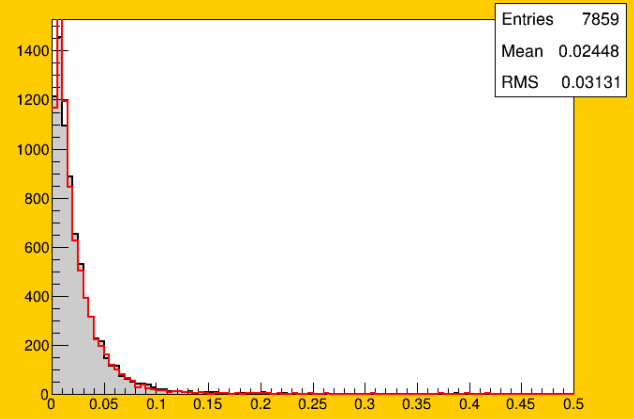
Chi2



Position Residual



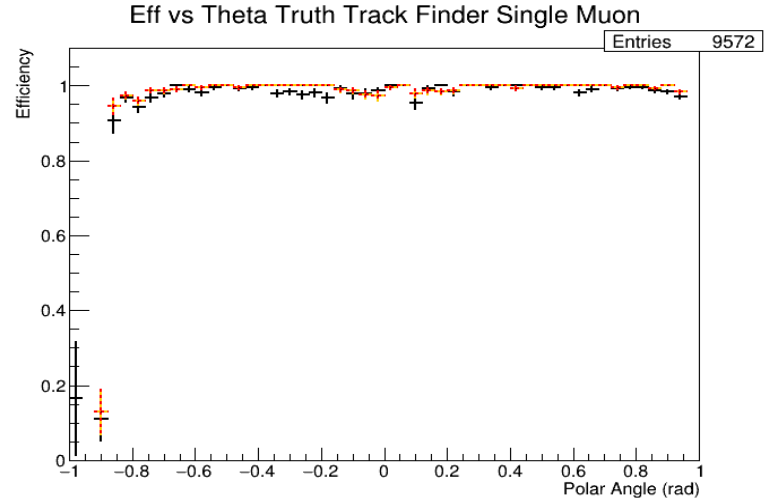
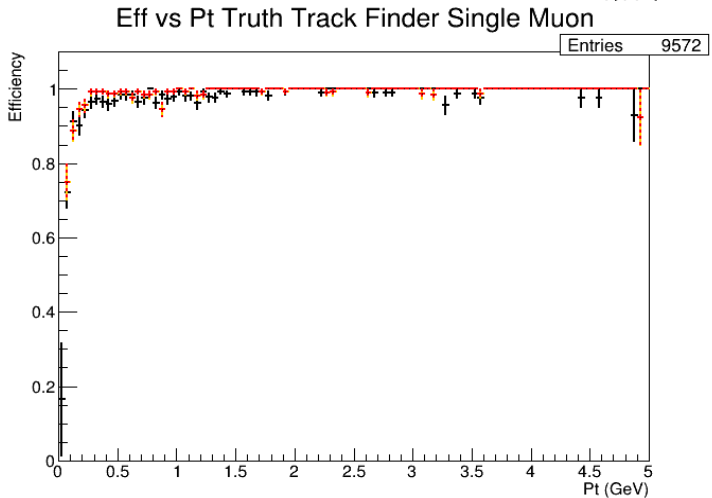
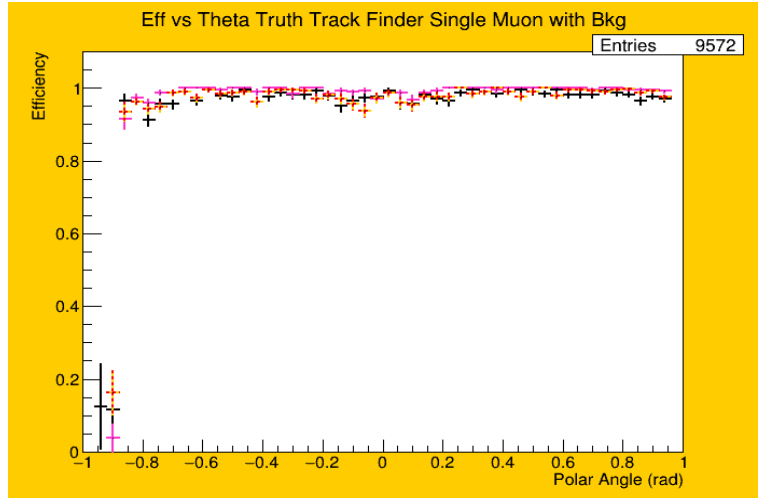
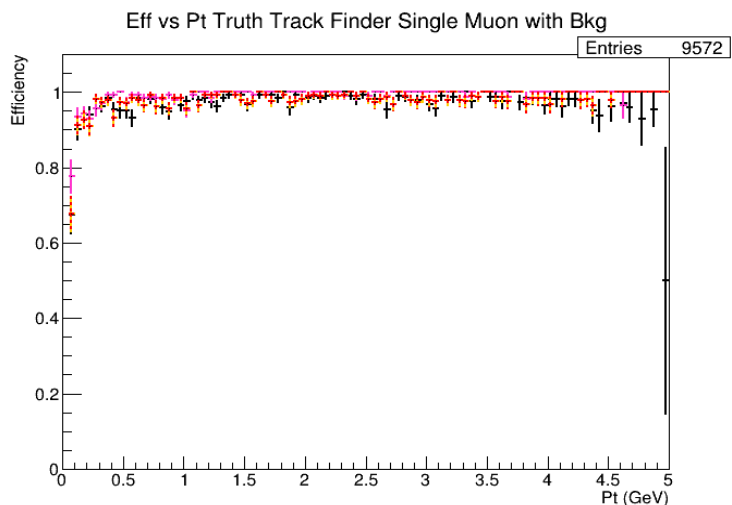
Momentum Residual





Muons + bkg

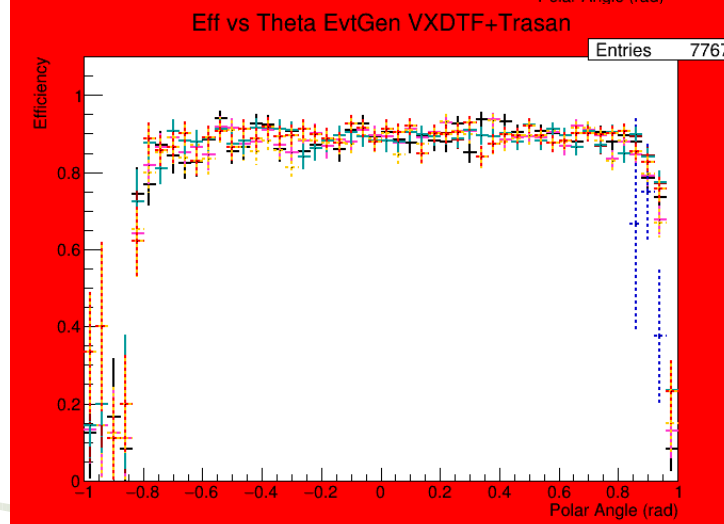
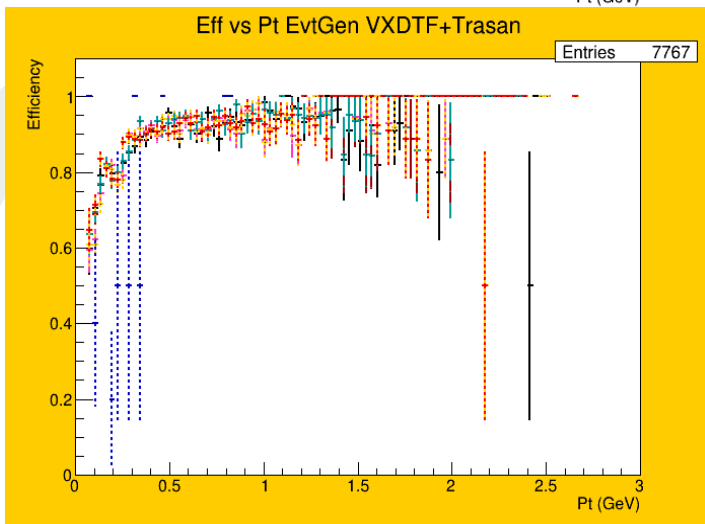
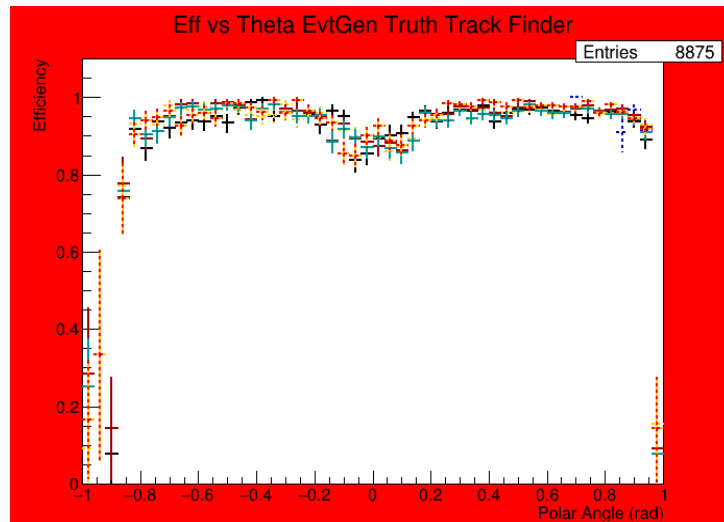
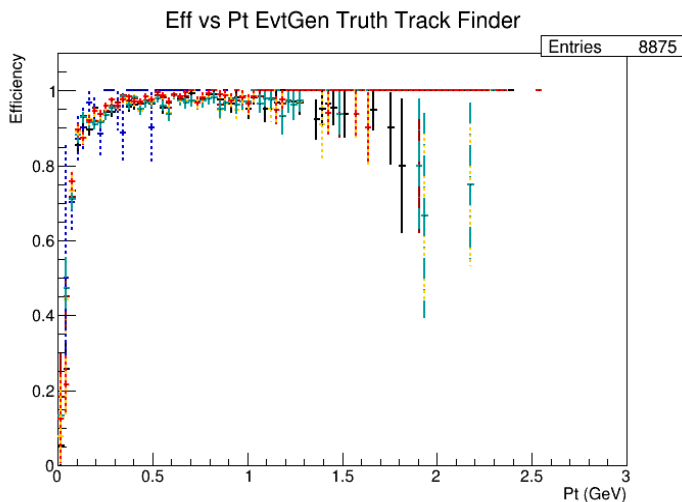
- Events with 10 [0, 5] GeV muons+11th campaign beam bkg (1 → 10 due to save simulation time)





EvtGen

- Default configuration
- Same TF comparisons as for muons





Conclusions & ToDo

- Urgent:
 - Understand eff loss around $\theta=0$ for TruthTF
 - Understand eff loss for EvtGen with pattern reco (might be related to particle type? Related to high multiplicity?)
- Test physics generators with bkg
- Less urgent: correct treatment for loopers
- Module code (merge & analysis) refactoring
- Your suggestions



Backups



Track Merging

Status Update

B. Oberhof
LNF-INFN Frascati, Italy

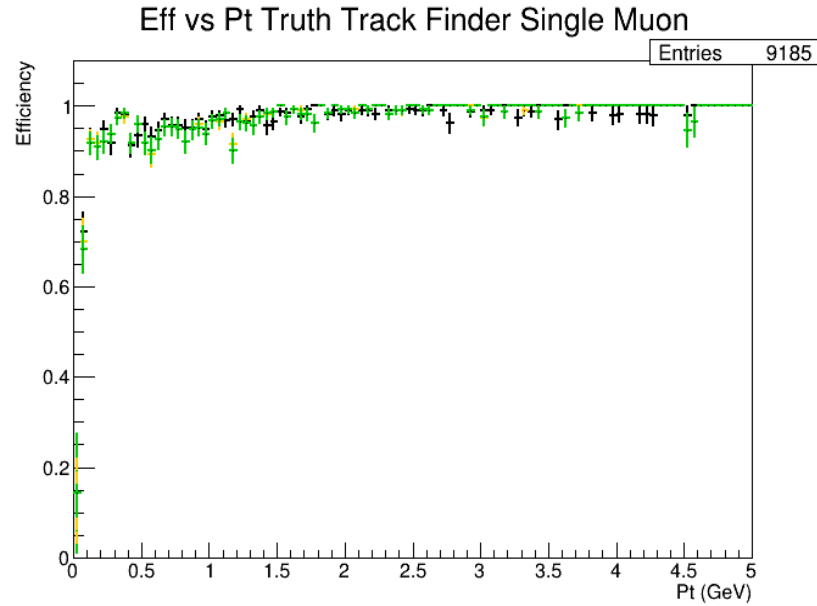
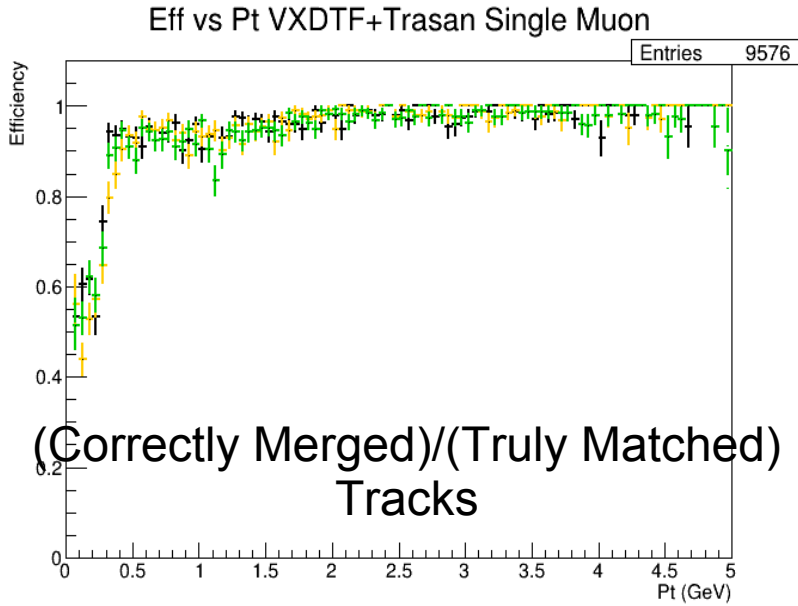
Tracking Meeting
27th March 2015



Single particle efficiency

- During last F2F meeting the problem of proper (out-)factorization of track finder and genfitter efficiencies was discussed
→ this doesn't seem to be the problem
- Plots for single muon/evt [0, 5 GeV] are now available on validation page (no factorization possible, track is found or not in CDC/VXD, and if possibly merged)
- Note: MCInfo from TrackCands pointed by GFTracks, i.e.:

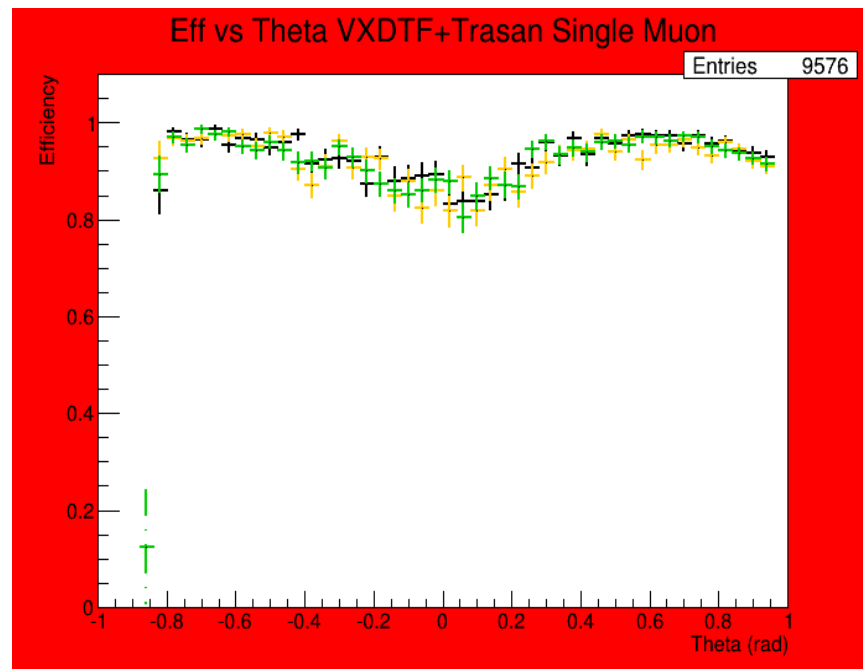
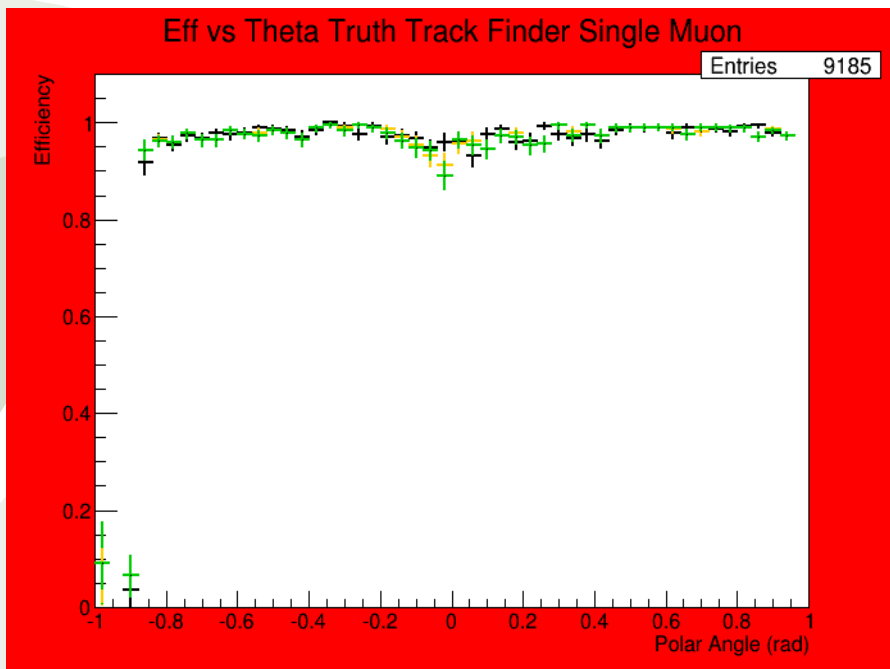
```
cdc_mcp_index = cdc_TrkCandPtr->getMcTrackId();  
const genfit::TrackCand* cdc_TrkCandPtr = DataStore->  
getRelatedToObj<genfit::TrackCand>(GFTrk, m_TrackCandColName)
```





Angular modulation

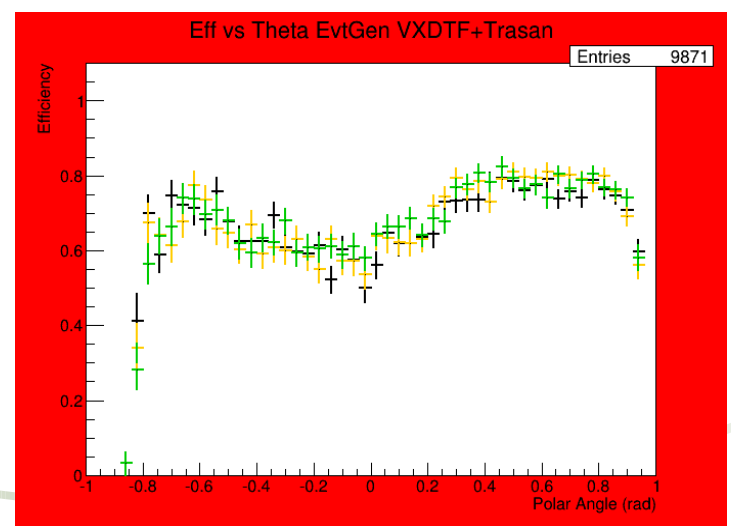
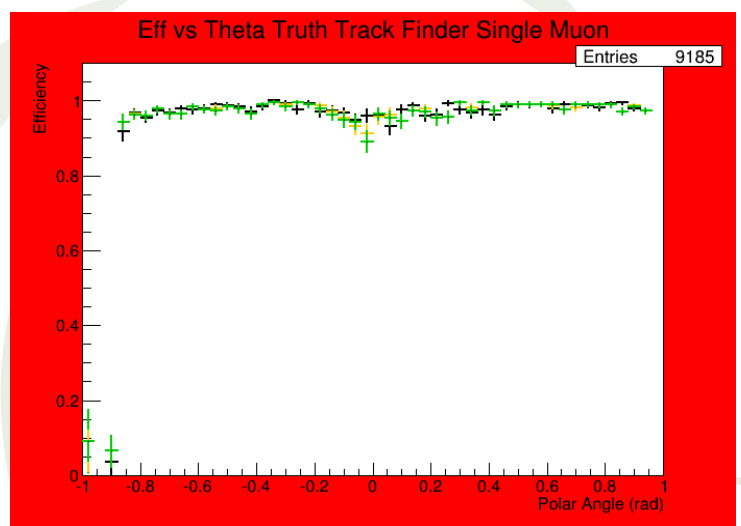
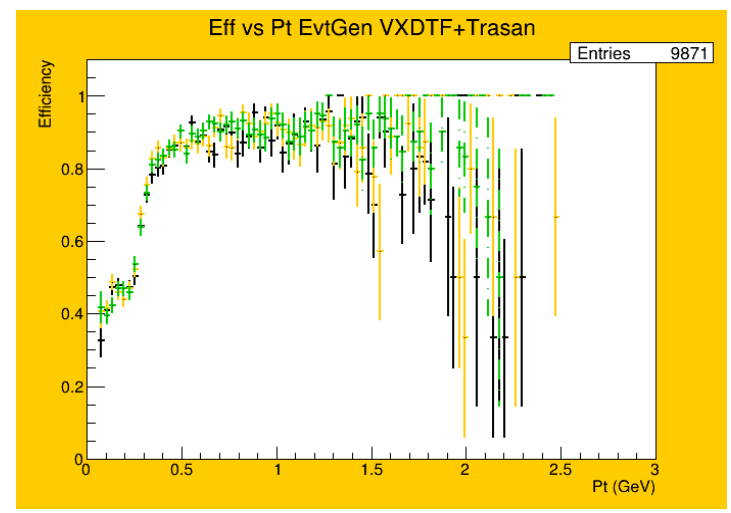
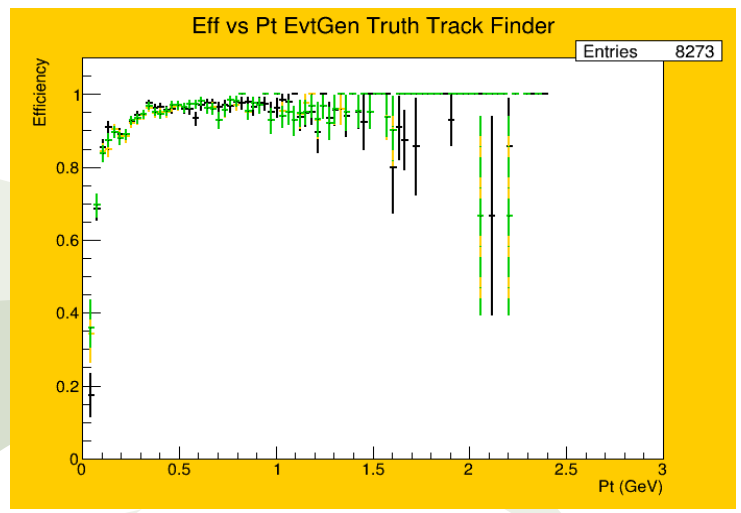
- Still efficiency drop around 90 degrees
→ improper curler treatment?
- Bad track quality (for some reason)?





$Y(4s)$ events

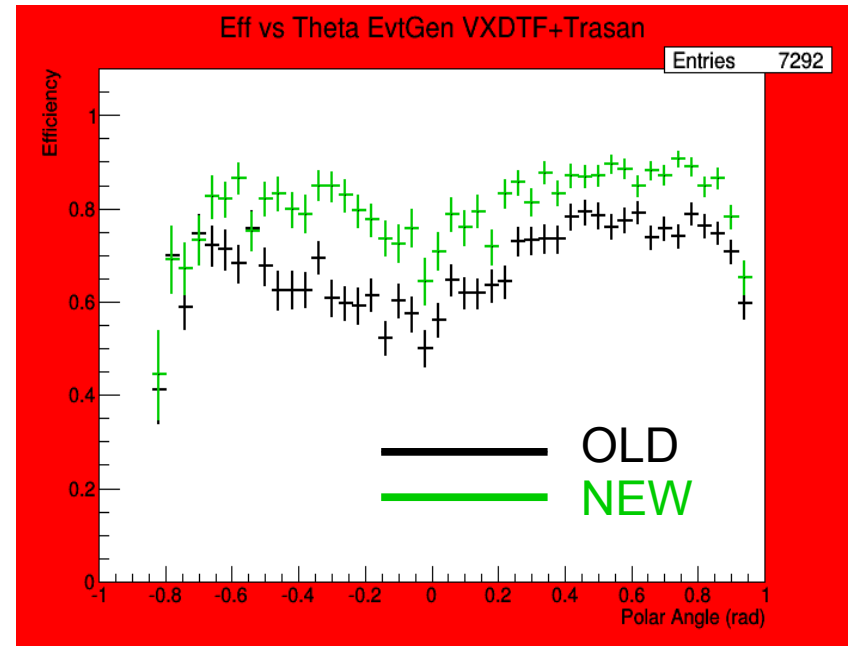
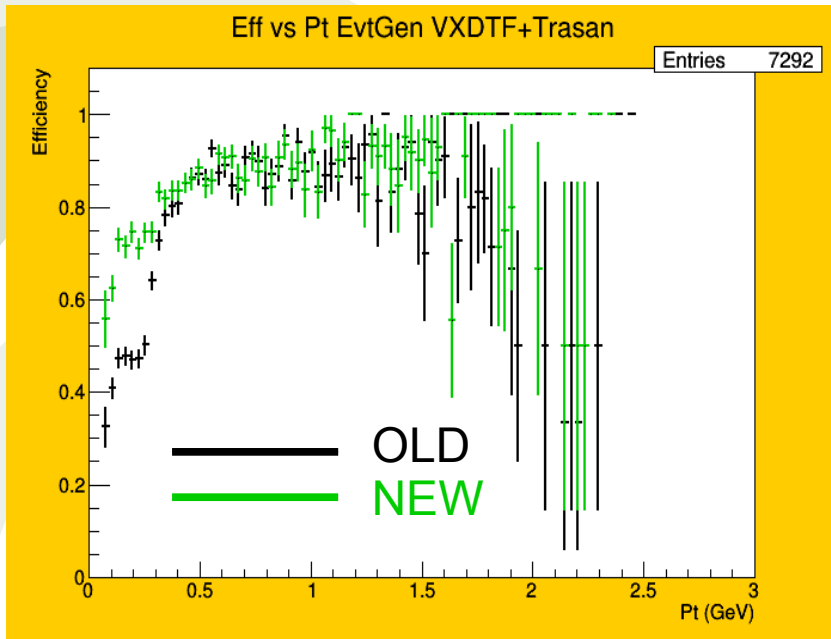
- Efficiency decrease is dramatic when TruthFinder \rightarrow VXDTF+ Trasan





Efficiency normalization

- Guess: might it be a normalization problem?
- I can have more TrkCandidates related to same MCCandidate
→ I modified the code (not committed) to allow max 1 merged track for each MCCandidate
- It seems to be the right direction, however efficiency is still low





Issues

- Obviously the former requirement does not allow treatment of curlers (2 VXDTrkCand associated to same MCParticle)
- I have to find out a proper way to not merge Tracks from same MCParticle which are “reasonably near” in momenta space
- Moreover: weird effects (i.e. efficiency decrease) are observed using previous code with TruthTrackFinder

