Flavour Tagging Performance in the presence of Pair Background

2014/02/26

Tohoku University

Tatsuya Mori

Performance evaluation of flavor tagging

- evaluated cases
  - CMOS + current ILD tracking
  - CMOS + FPCCDTrackFinder
  - FPCCD + FPCCDTrackFinder
- considering pair-BG
- Sample: Z → bb, cc, qq @ 250 GeV
- Flavor Tagging Algorithm: LCFIPlus
  - sample: 2000 events individually
  - training sample: 14000 events individually (not overlaid with pair-BG)

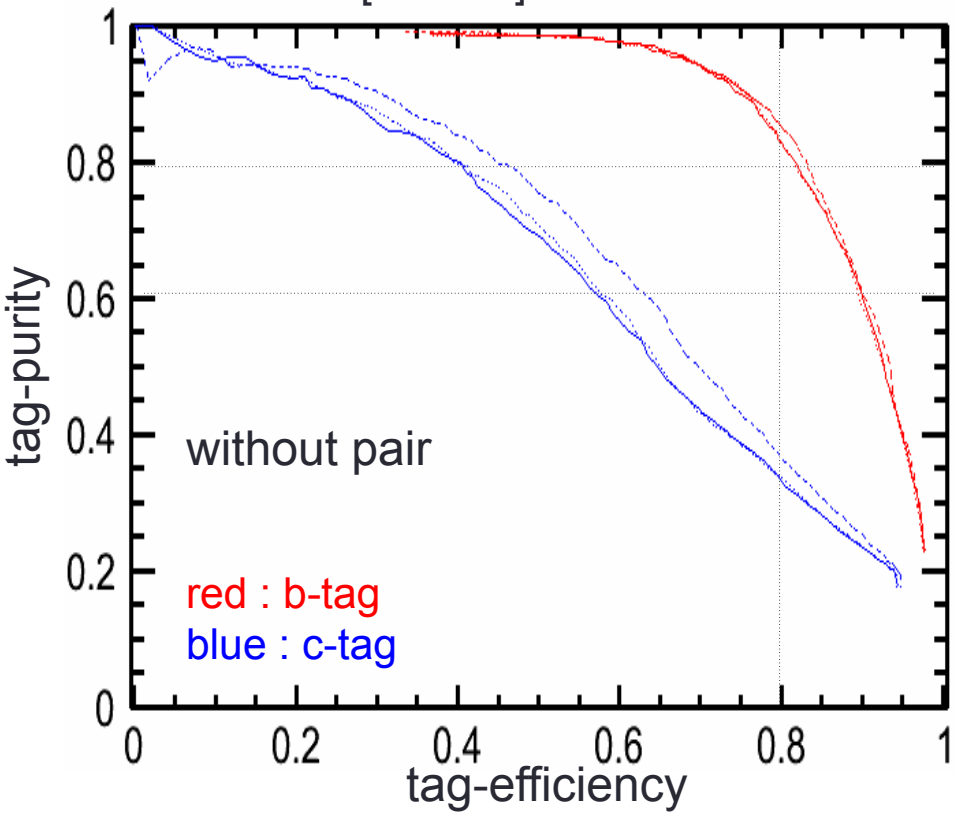
# of BXs / readout

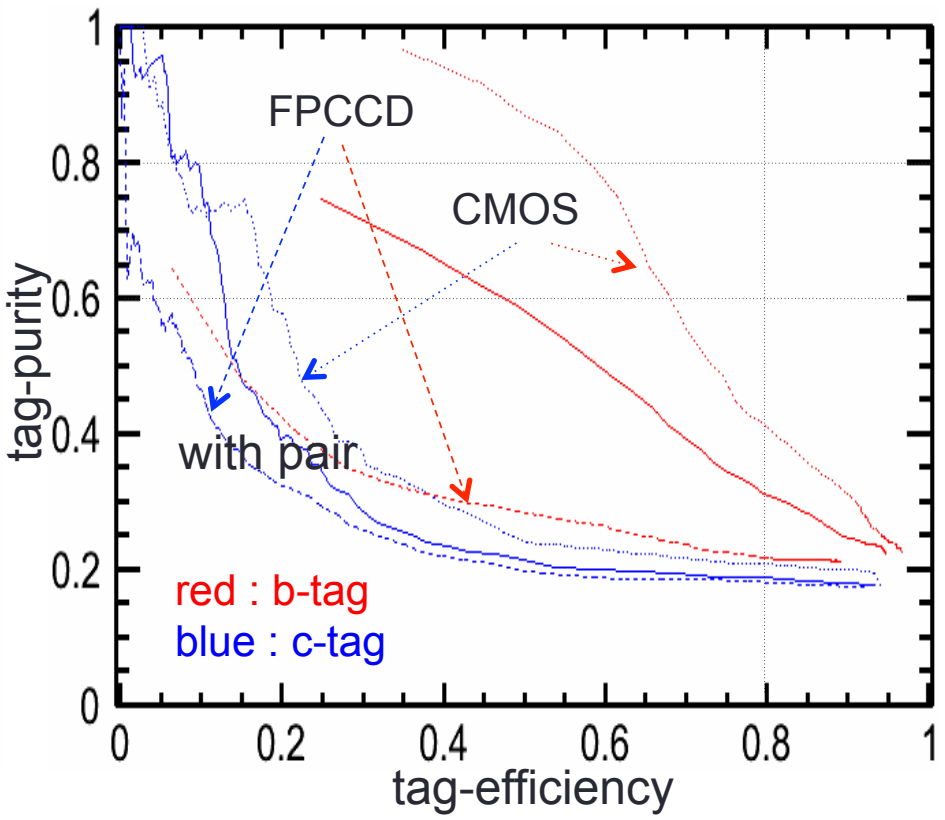
layer	FPCCD	CMOS
0	1312	90
1	1312	18
2	1312	180
3	1312	180
4	1312	180
5	1312	180

performance without and with pairs

$Z^* \rightarrow b\bar{b}, c\bar{c}, q\bar{q}$ (q : u, d, s) @ 250 GeV

[———] : CMOS + current ILD tracking
[. . . . .] : CMOS + FPCCD TF
[– – – –] : FPCCD + FPCCD TF





- The performance degrades by pairs
- FPCCDTrackFinder is better than current ILD tracking regardless of pairs
- CMOS is better than FPCCD in the presence of pairs

Degradation by pairs

VXD	tracking	pairs	b-tag purity [%] @ eff. 80 %	c-tag purity [%] @ eff. 60 %
CMOS	std	×	82.8	56.4
CMOS	std	○	52% down **30.4**	36% down **20.0**
CMOS	FPCCDTF	×	83.0	58.1
CMOS	FPCCDTF	○	42% down **40.8**	35% down **22.8**
FPCCD	FPCCDTF	×	85.5	63.9
FPCCD	FPCCDTF	○	64% down **21.5**	45% down **18.7**

- **Regardless of pairs, FPCCDTrackFinder is better that current ILD tracking**
  - **purity 10 % up @ efficiency 80 % b-tag**
  - **purity 3 % up @ efficiency 60 % c-tag**
- **With pairs, FPCCD is worse than CMOS**
  - **purity 19 % down @ efficiency 80 % b-tag**
  - **purity 3 % down @ efficiency 60 % c-tag**

Pt distribution of tracks in b-jet (without pair-BG)

Setup: FPCCD + FPCCDTrackFinder

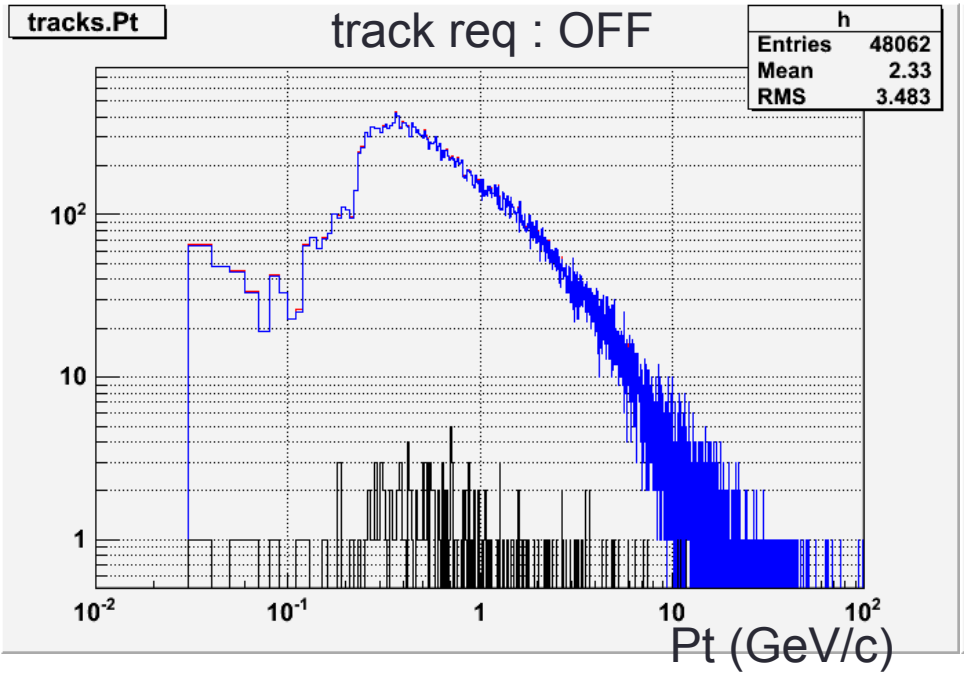
Track Requirement: SIT hit >= 1 || TPC hit >= 10 || |cosθ| > 0.9

→ Basically, most pair BG tracks don't have SIT or TPC hits.

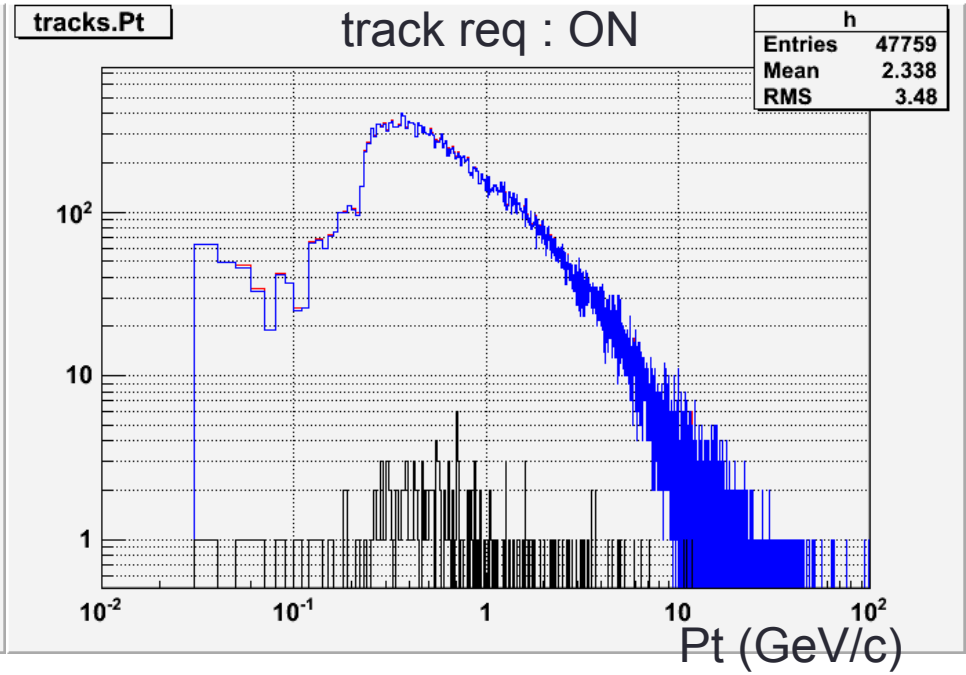
→ Most signal tracks with |cosθ| > 0.9 basically don't have SIT or TPC hits due to their coverage, so I set |cosθ| > 0.9 so that those tracks can survive in the above requirement.

red: all tracks
blue: tracks with purity > 0.75
black: tracks with purity [0, 0.75]

b-tag purity : 85.5% @ eff 80%



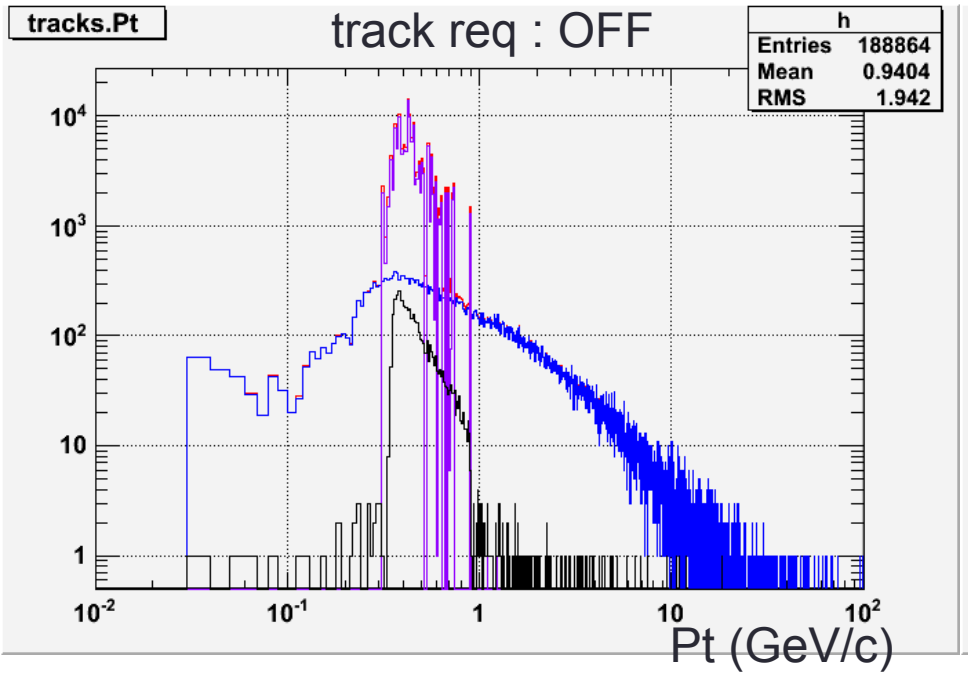
b-tag purity : 84.1% @ eff 80 %



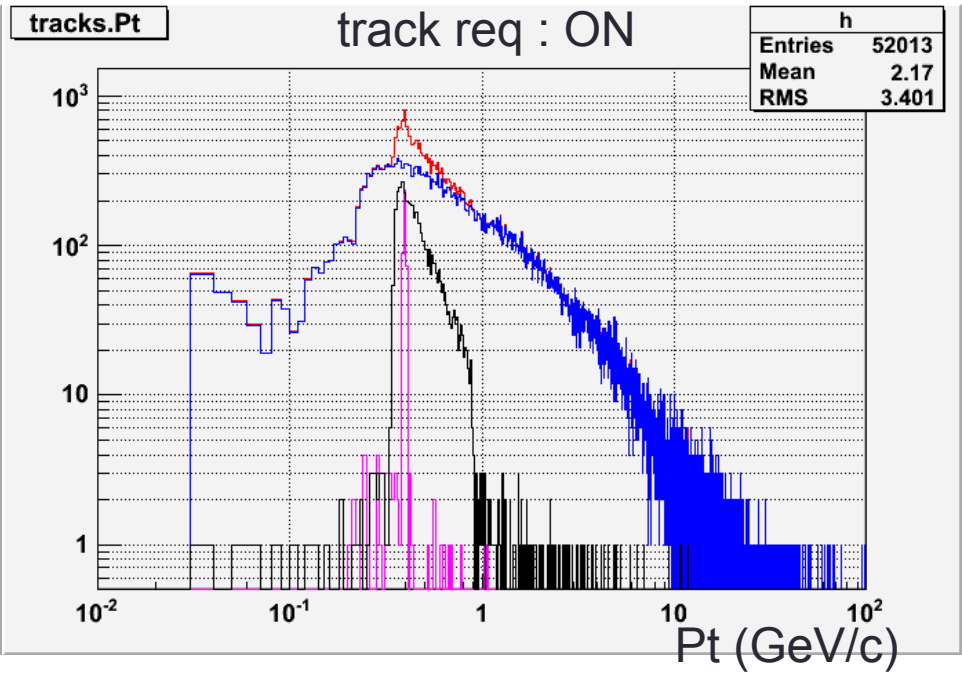
Pt distribution of tracks in b-jet (with pair-BG)

red: all tracks
blue: tracks with purity > 0.75
black: tracks with purity [0, 0.75]
purple: pair-BG tracks

b-tag purity : 21.5% @ eff 80%



b-tag purity : 67.8% @ eff 80 %

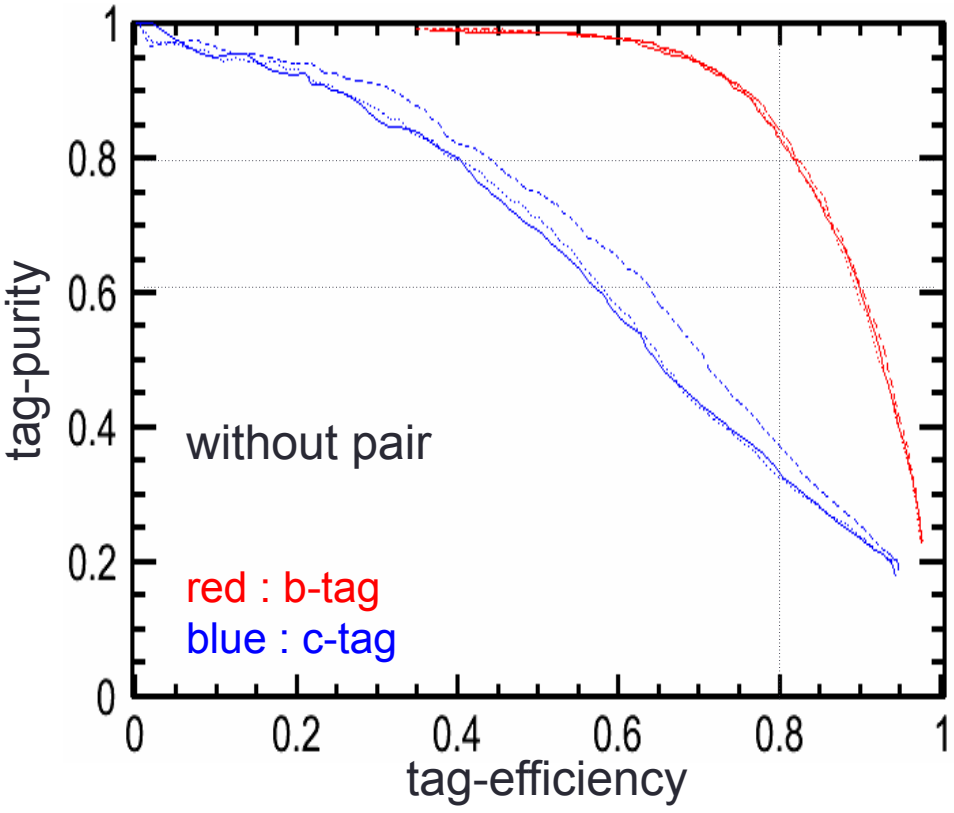


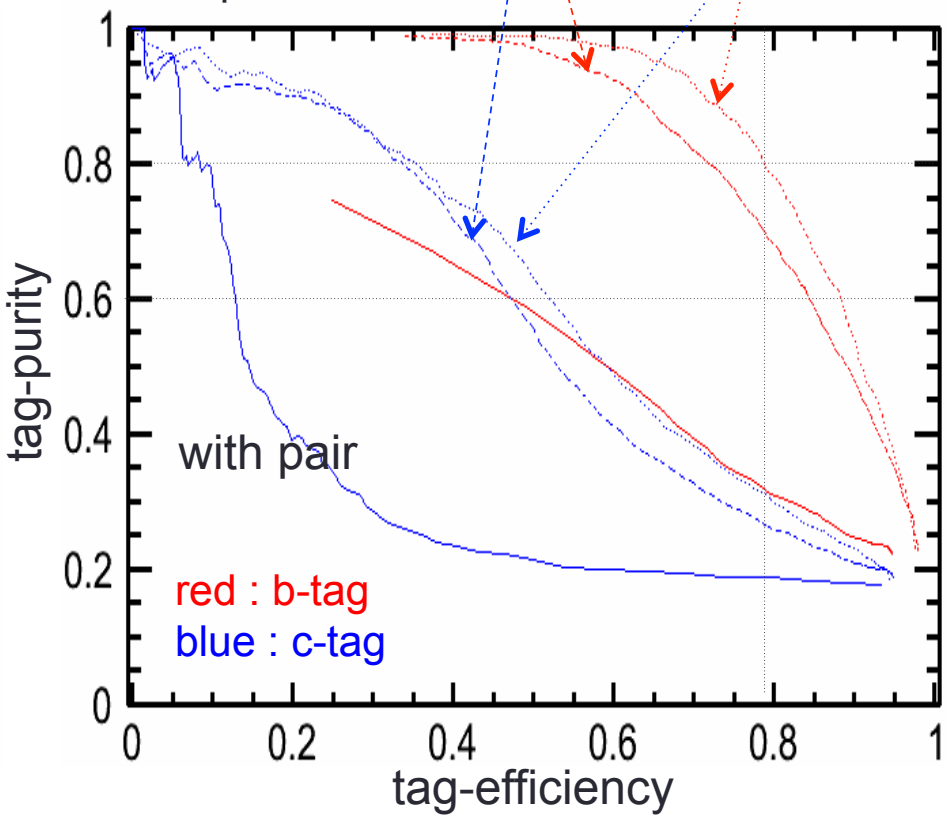
- "Track req : ON" discards most of pair-BG, but don't do tracks with purity [0, 0.75]
- We find that pair-BG makes a lot of tracks with purity [0, 0.75]
  - Track req doesn't kill these tracks → We need to find the way to discriminate them from tracks with purity > 0.75

performance without and with pairs 2

$Z^* \rightarrow b\bar{b}, c\bar{c}, q\bar{q}$ (q : u, d, s) @ 250 GeV

[———] : CMOS + current ILD tracking
[. . . . .] : CMOS + FPCCD TF + track req.
[– – – –] : FPCCD + FPCCD TF + track req.





- **We can suppress the deterioration (but lower than the case without pairs)**

Improvement by track requirement

VXD	tracking	pairs	Track Req.	b-tag purity [%] @ eff. 80 %	c-tag purity [%] @ eff. 60 %
CMOS	std	×	×	82.8	56.4
CMOS	std	○	×	30.4	20.0
CMOS	FPCCDTF	×	×	83.0	58.1
CMOS	FPCCDTF	×	○	82.9	57.4
CMOS	FPCCDTF	○	×	37% recover 40.8	27% recover 22.8
CMOS	FPCCDTF	○	○	**77.6**	**49.4**
FPCCD	FPCCDTF	×	×	85.5	63.9
FPCCD	FPCCDTF	×	○	84.1	65.5
FPCCD	FPCCDTF	○	×	46% recover 21.5	23% recover 18.7
FPCCD	FPCCDTF	○	○	**67.8**	**41.6**

- **FPCCD is also worse than CMOS with track requirement**
  - **purity 10 % down @ efficiency 80 % b-tag**
  - **purity 8 % down @ efficiency 60 % c-tag**

Comparison of CPU time and Memory

Values are the mean calculated from 2000 events of Z → bb @ 250 GeV

VTX	tracking	pairs	CPU time [sec/evt]		Max Memory [GB/evt]
			Silicon Tracking	Full Tracking	
CMOS	std	×	0.2	1.1	408.7
CMOS	std	○	**342.0**	6.8	561.5
CMOS	FPCCDTF	×	7.2	1.0	619.5
CMOS	FPCCDTF	○	**34.0**	3.0	709.6
FPCCD	FPCCDTF	×	5.6	1.0	623.0
FPCCD	FPCCDTF	○	**407.6**	27.7	2276.0

1/10

- In the case of CMOS with pairs, FPCCDTrackFinder make CPU time ~1/10.
- In the case of FPCCD, we need to reduce more CPU time.

Summary and Plan

- Summary
  - Flavor tagging performance is degraded by pair-BG tracks
  - Track requirement reduces pair-BG tracks
  - Finally, in the presence of pairs, flavor tagging performance is

VXD	tracking	pairs	Track Req.	b-tag purity [%] @ eff. 80 %	c-tag purity [%] @ eff. 60 %
CMOS	std	○	×	30.4	20.0
CMOS	FPCCDTF	○	○	**77.6**	**49.4**
FPCCD	FPCCDTF	○	○	**67.8**	**41.6**

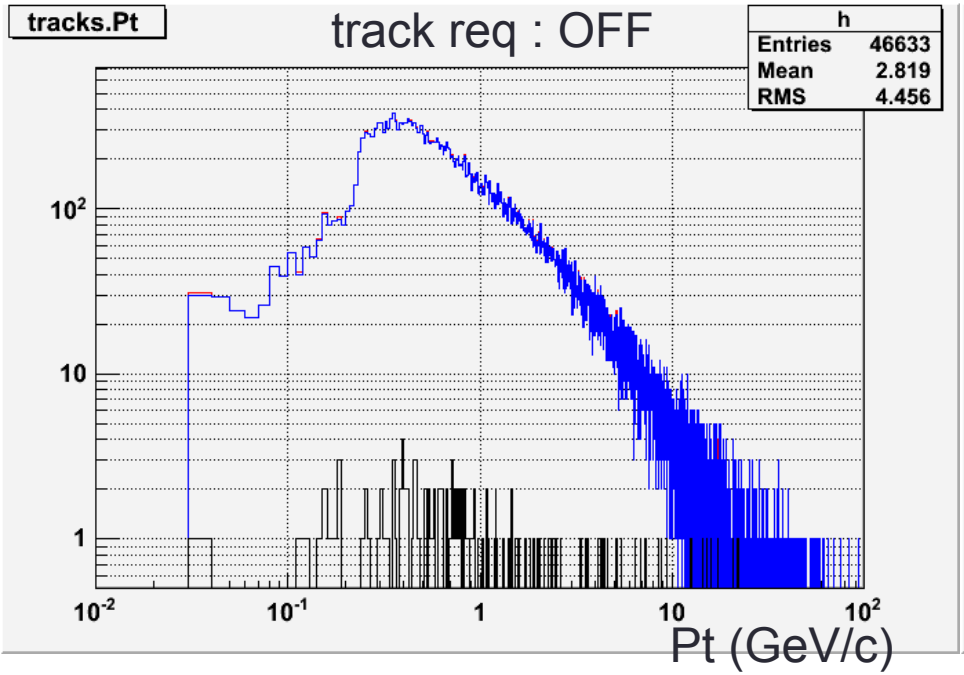
  - FPCCDTrackFinder reduces CPU time by up to 1/10 in the case using CMOS
- Plan
  - Try to reduce ghost tracks by
    - new track requirement
    - modification of tracking algorithm

Backup

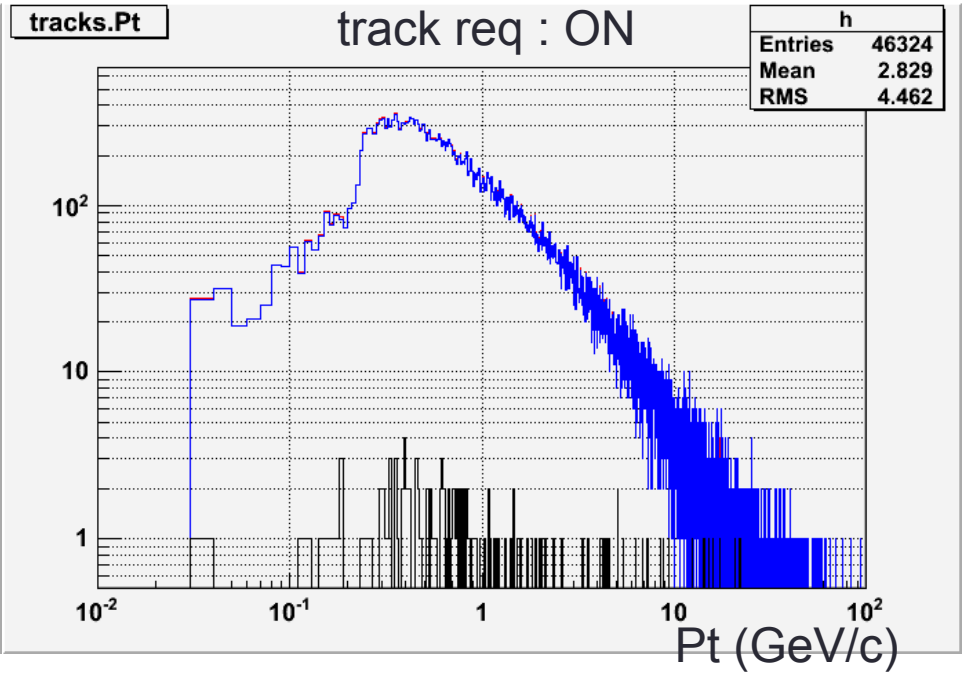
Pt distribution of tracks in c-jet (without pair-BG)

red: all tracks
blue: tracks with purity > 0.75
black: tracks with purity [0, 0.75]

c-tag purity : 63.9% @ eff 60%



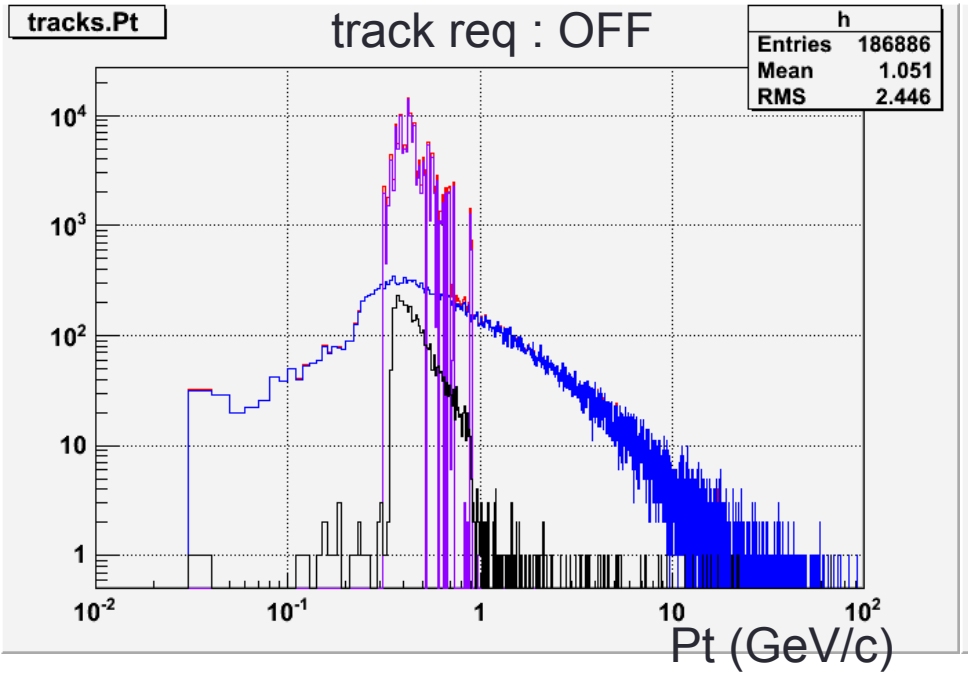
c-tag purity : 65.5% @ eff 60 %



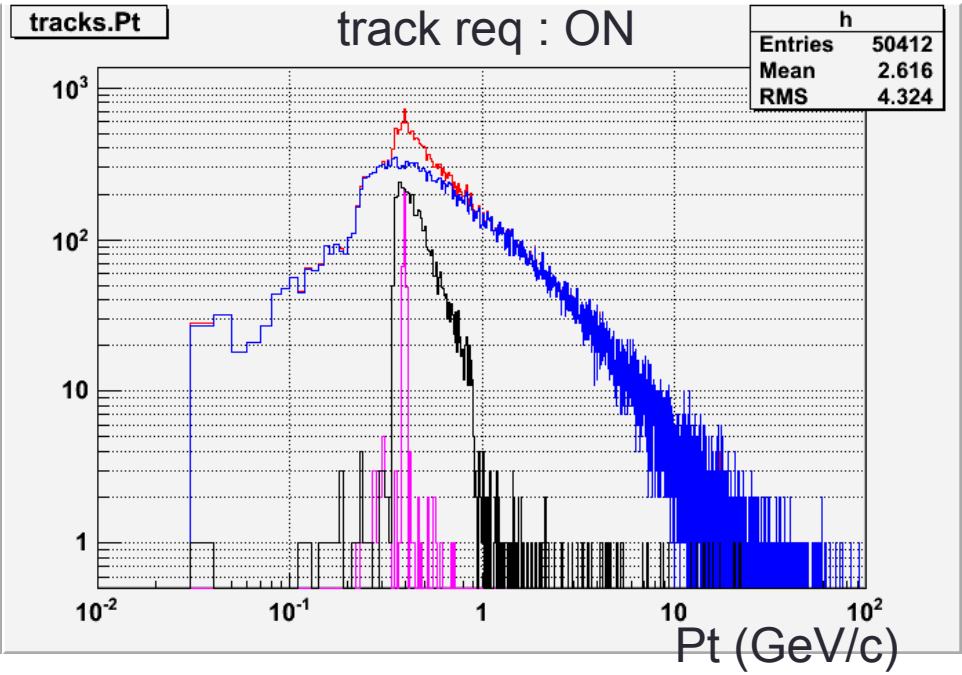
Pt distribution of tracks in c-jet (with pair-BG)

red: all tracks
blue: tracks with purity > 0.75
black: tracks with purity [0, 0.75]
purple: pair-BG tracks (purity == -1)

c-tag purity : 18.7% @ eff 60%



c-tag purity : 41.6% @ eff 60 %



almost same as the case of b-jet