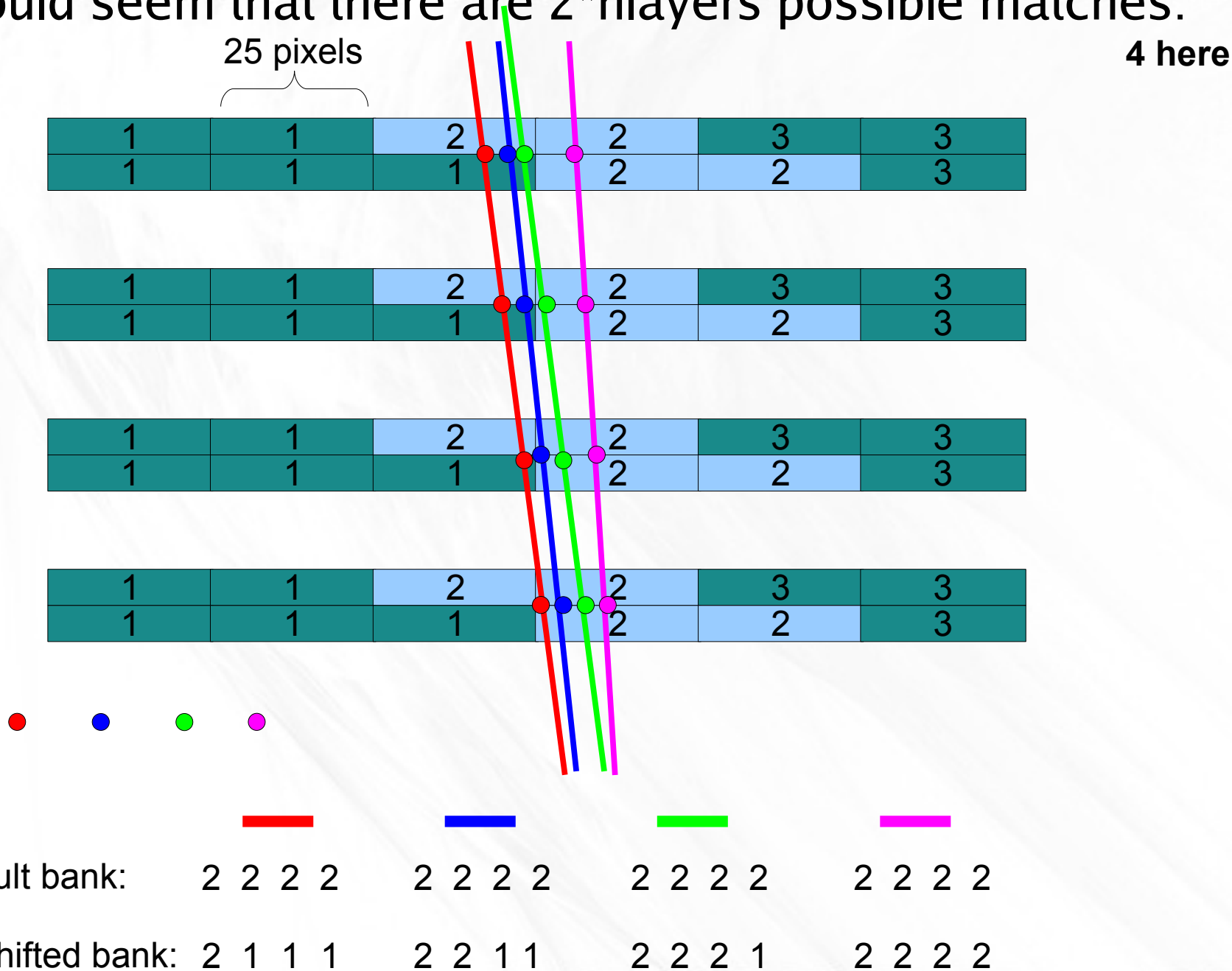
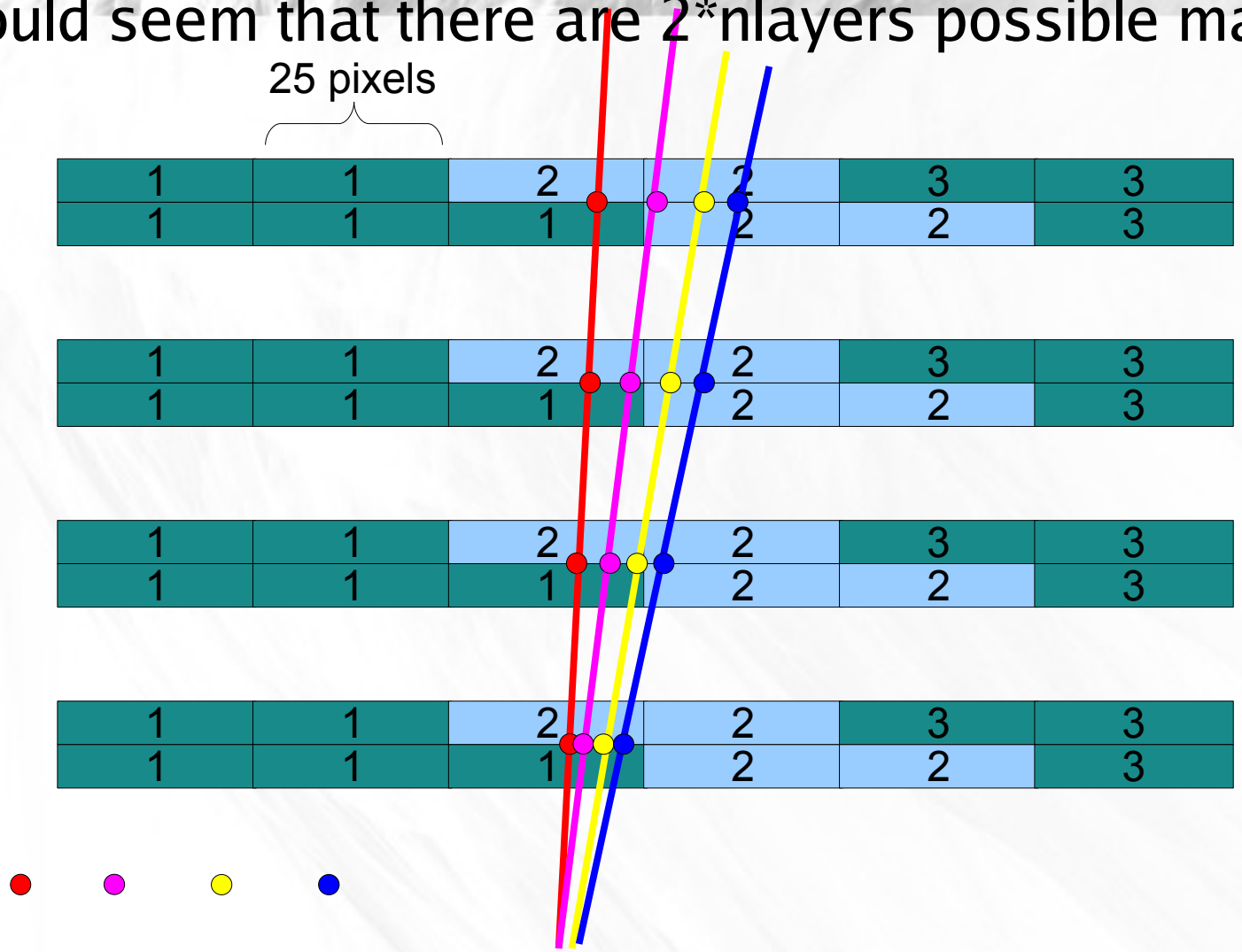
1. Topological filtering

- Attempt to filter matched roads using topological arguments (no cross-matches)
- Explanation why it failed

For a given pattern in default bank (2 2 2 2 in the example below), it would seem that there are 2*nlayers possible matches:



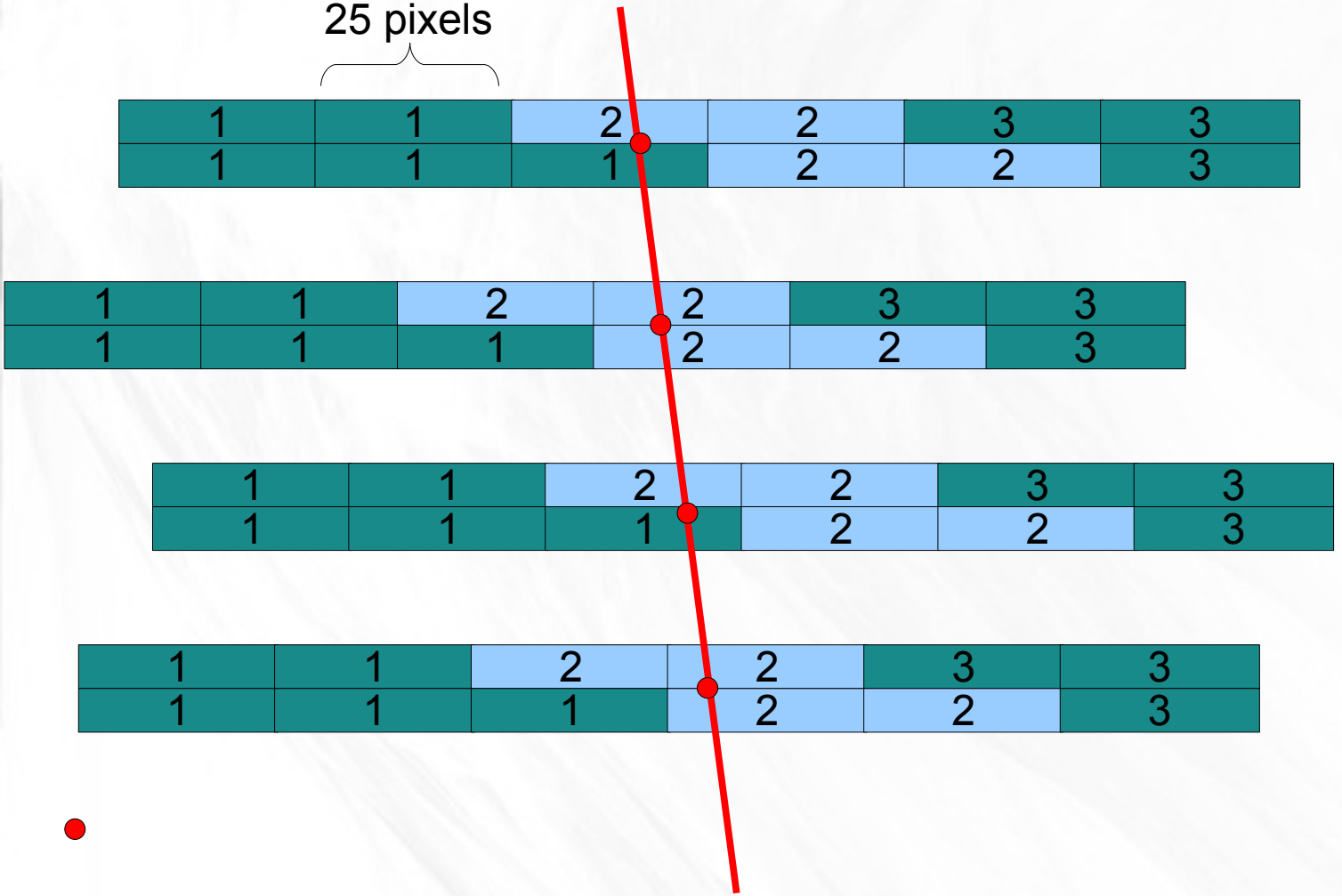
For a given pattern in default bank (2 2 2 2 in the example below), it would seem that there are 2*nlayers possible matches:



Default bank: 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

SS-shifted bank: 1 1 1 1 1 1 1 2 1 1 2 2 1 2 2 2

But: the picture was wrong! The layers are **not** aligned!



We have a “cross-match” not enumerated before!

So in general, we cannot make any requirement on matches (such as requiring no not being “crisscrossed”)

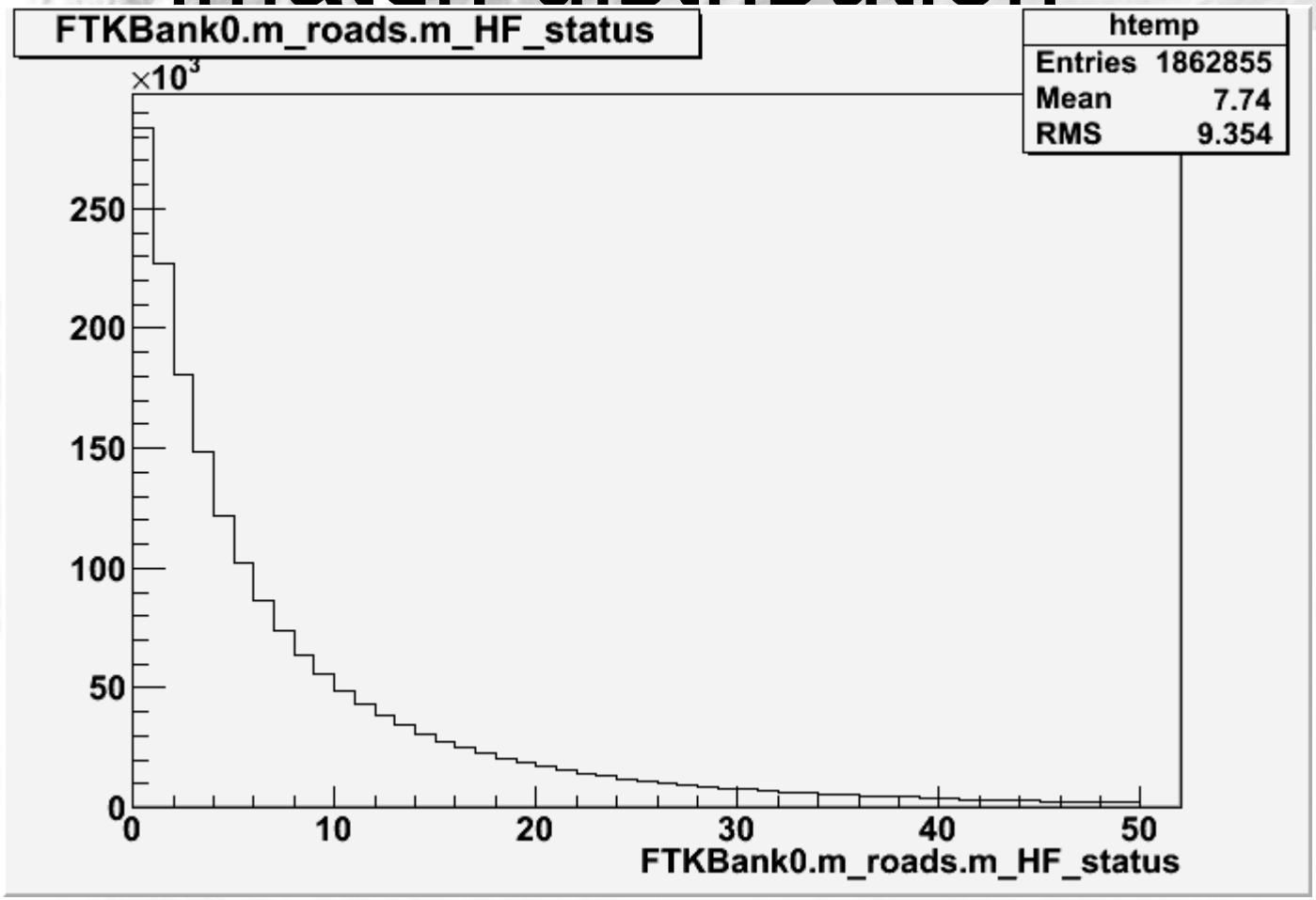
2 2 2 2

2 1 2 1

2. # of matching SS-shifted roads

- Distribution of # of matching roads
- How cutting on max matches affects efficiency

“imatch”distribution



- Given a default road, we find a list of N matching ss-shifted roads.
  - Matching here means ss1=ss2 or ss1=ss2-1
- **i=[0..N-1]** matching ss-shifted roads are sorted by coverage.
- For each match, we create a new “road”(which only has hits in the intersection of SS's)
- Plotted above is a **histogram of i values**. I.e., one entry for each final road.
- Sometimes there are 50 ss-shifted roads matching a given default road!

Efficiency vs a cut on nmatches (+ effect of Joe's 10/11 scheme)

Whbb @ 10^34
Only barrel patterns; |eta|<1.0
Efficiency with respect to all primary tracks (mostly pions) with pt>1GeV

description	#matches	track eff	#roads	#fits (tot)	
default	N/A	75.87%	10,601	10,681,364	
Ss-shift	50	74.91%	47,843	1,846,518	Factor: 5.8 Eff: -0.96%
Ss-shift	30	74.83%	45,719	1,756,362	Factor: 6.1 Eff: -1.04%
Ss-shift	20	74.70%	42,591	1,623,017	Factor: 6.6 Eff: -1.17%
Ss-shift	10	73.83%	34,333	1,285,323	Factor: 8.3 Eff: -2.04%
Ss-shift	5	72.15%	24,185	879,092	Factor: 12 Eff: -3.72%
ss+joe	50	72.10% (?)	47,843	385,062	Factor: 28 Eff: -3.77%

IPAT efficiency = 85%

Joe's reduction to majority fix is largely orthogonal to ss-shifted scheme!

Notes on efficiency

- Absolute efficiency (for all primary tracks) is still quite low. This is a general problem
  - Pions not leaving hits in outer layers?
- Relative to default case, SS-shift + joe case works well
- Additional possibilities to reduce #fits by cutting on max nmatches

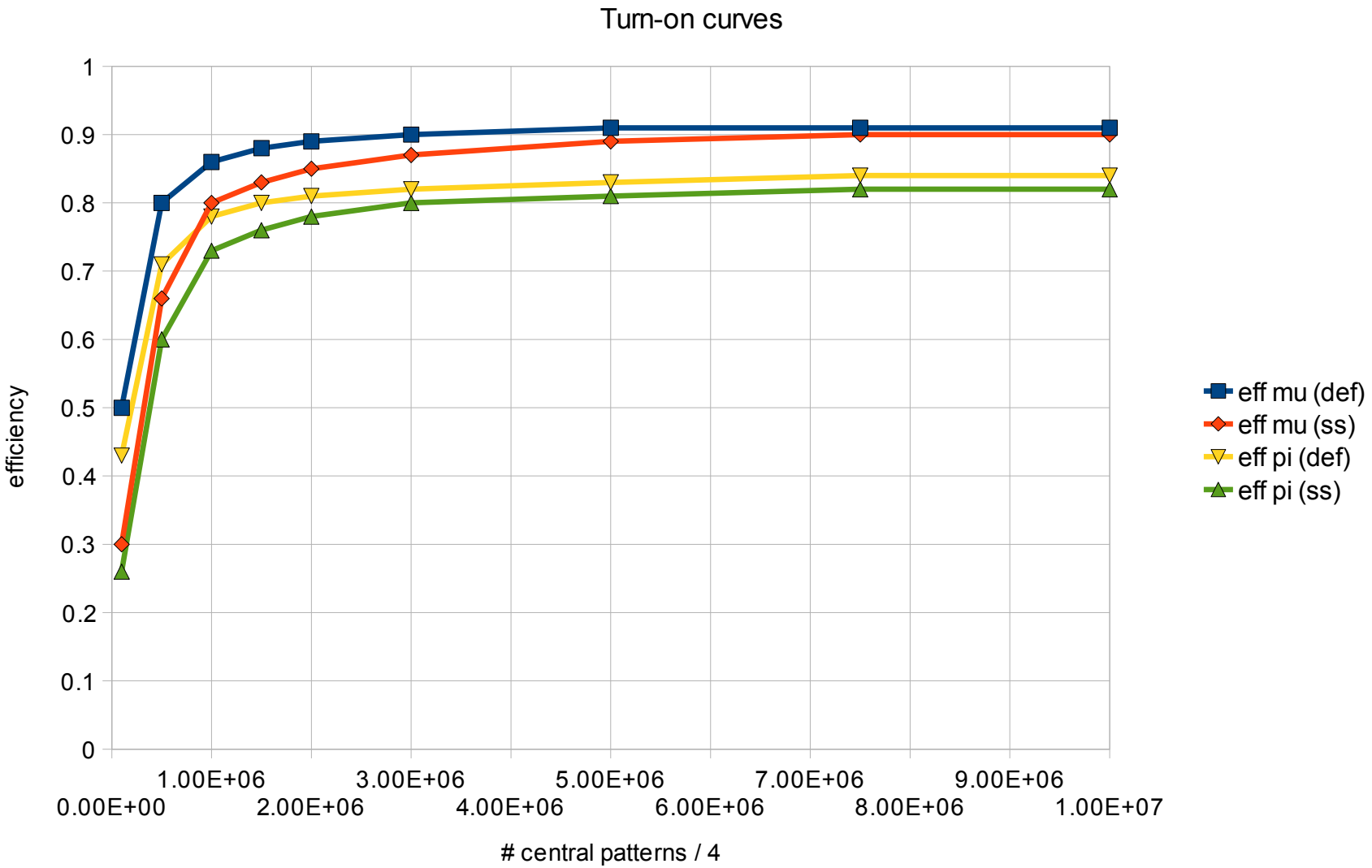
3. turn-on curves

- Efficiency turn-on curves for muons and pions
- Using single mu/pi samples in region 6
- With and without SS-shifted scheme

Turn-on curves with ss-shifted arch

single muons and pions in central eta ONLY

Only barrel patterns are loaded (!)

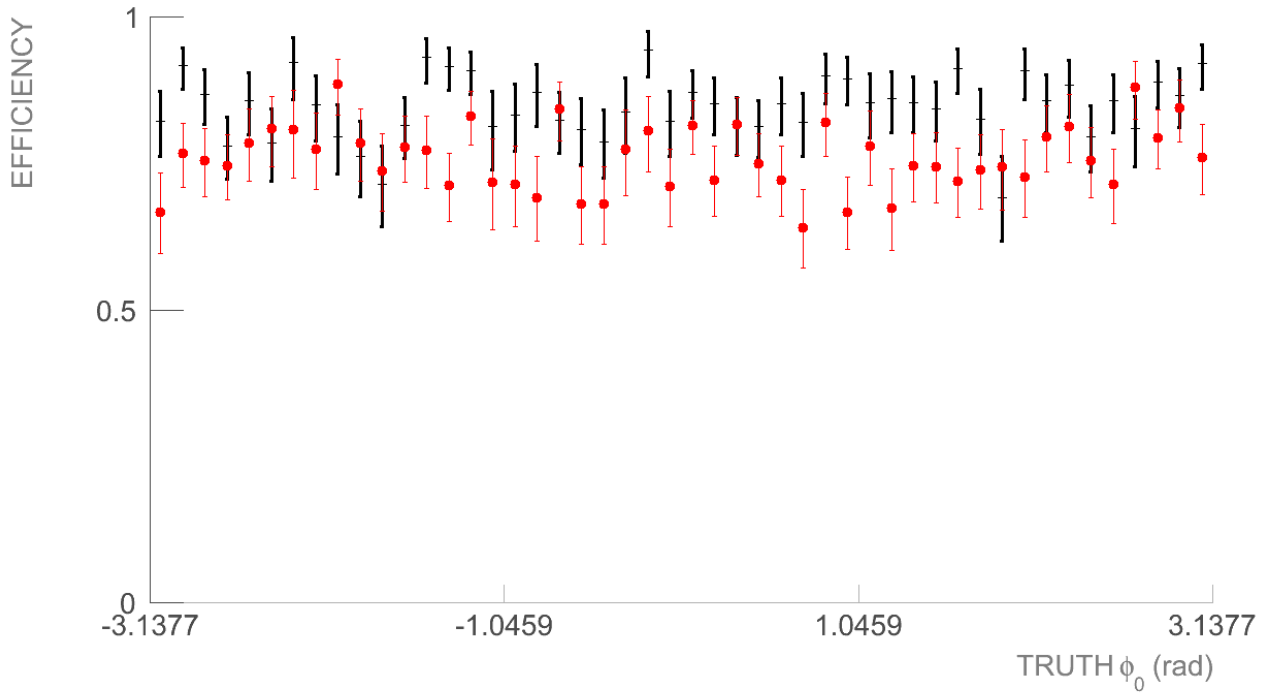


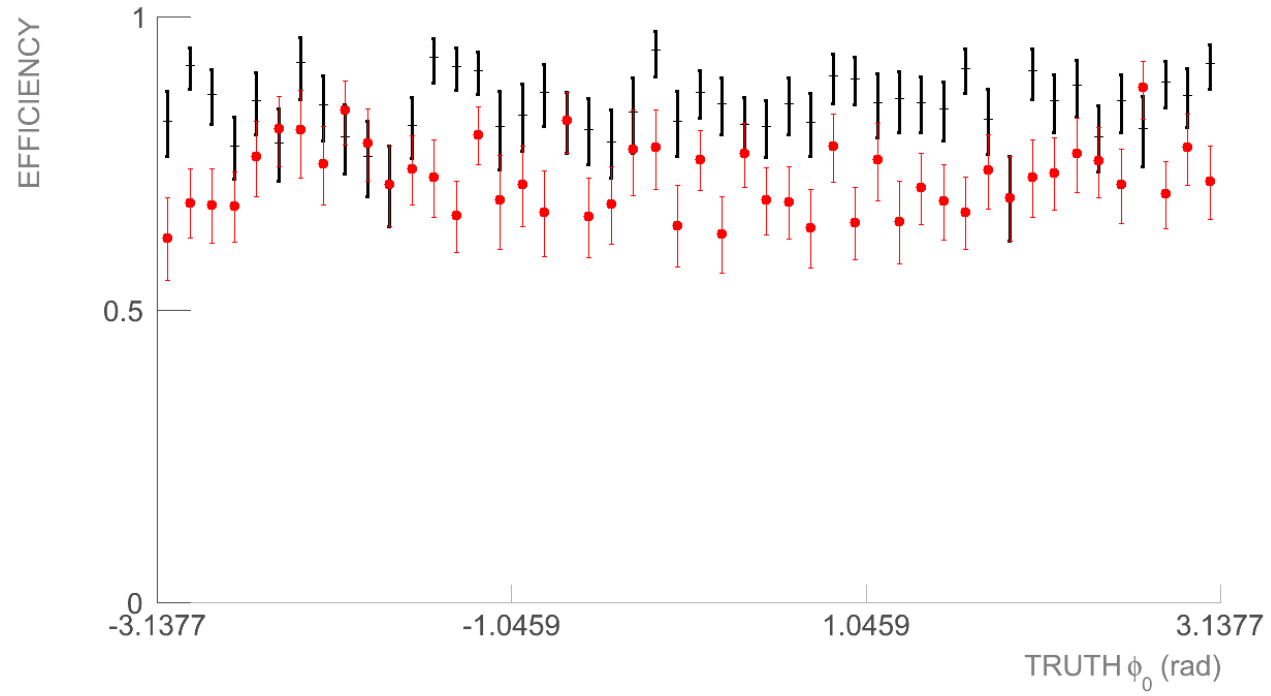
Summary table for efficiencies

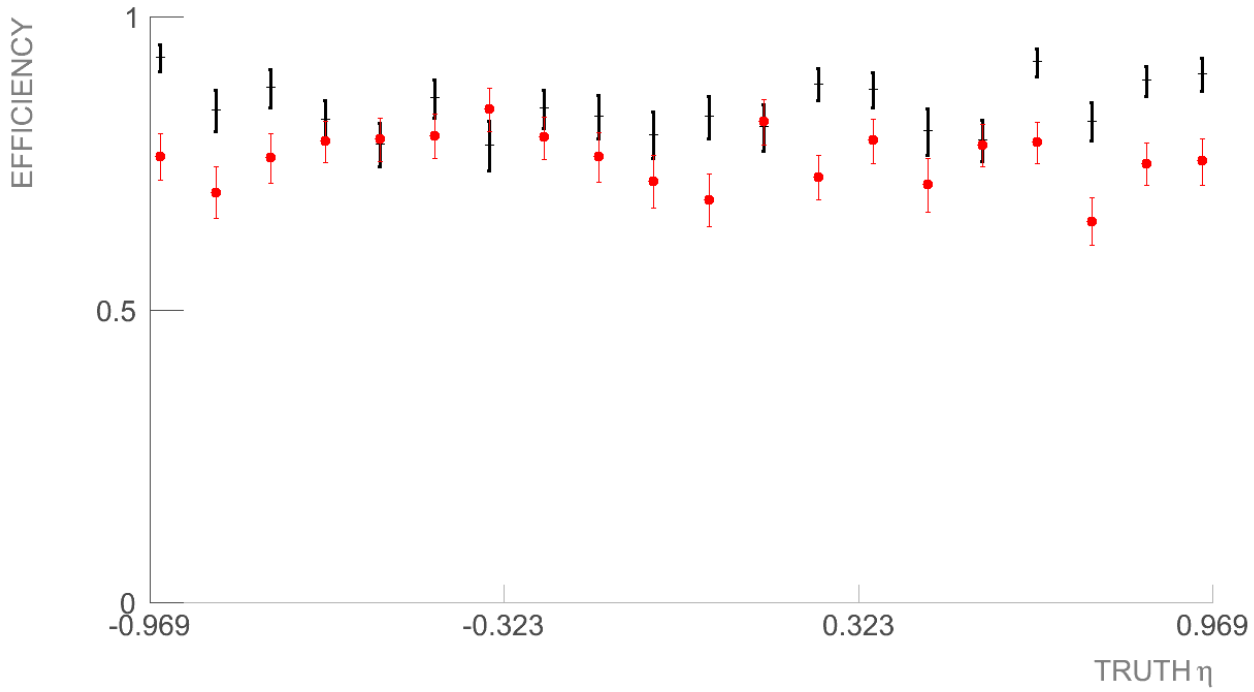
# patt / 4	eff mu (def)	eff mu (ss)	eff pi (def)	eff pi (ss)
100000	0.5	0.3	0.43	0.26
500000	0.8	0.66	0.71	0.6
1000000	0.86	0.8	0.78	0.73
1500000	0.88	0.83	0.8	0.76
2000000	0.89	0.85	0.81	0.78
3000000	0.9	0.87	0.82	0.8
5000000	0.91	0.89	0.83	0.81
7500000	0.91	0.9	0.84	0.82
10000000	0.91	0.9	0.84	0.82

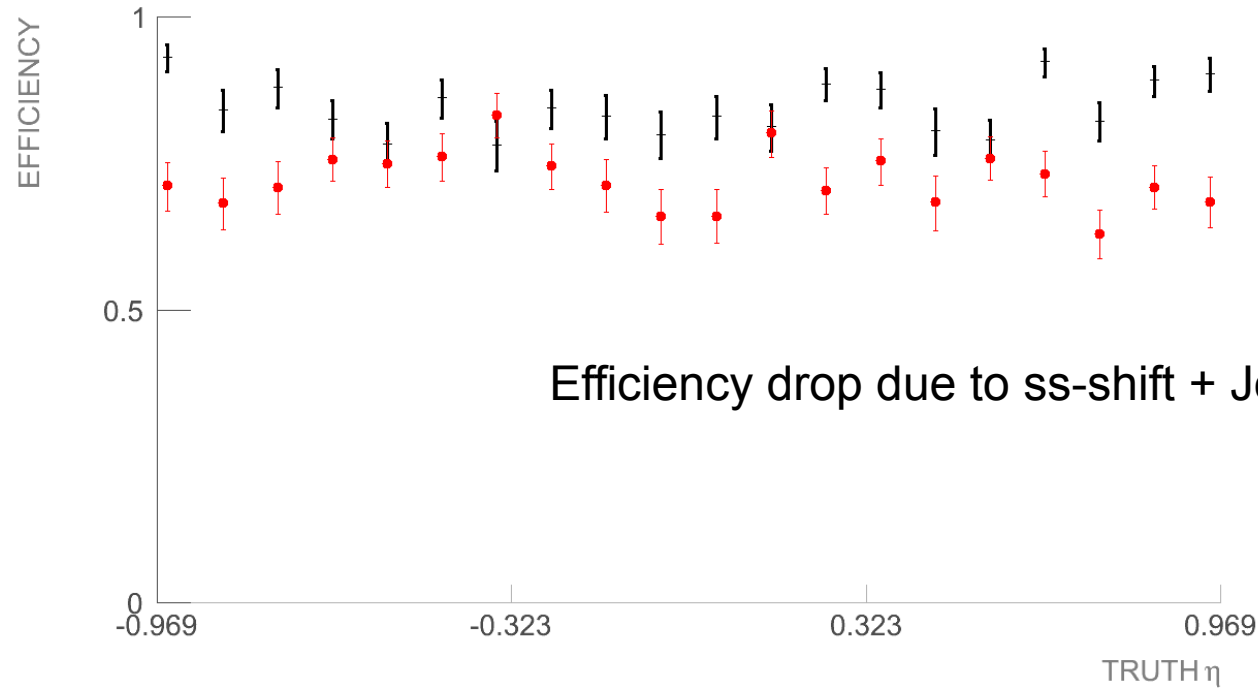
Resolution/efficiency plots

- Top plot: default TrigFTKSim vs IPAT
- Bottom plot: SS-shift (50 matches) +Joe 10/11

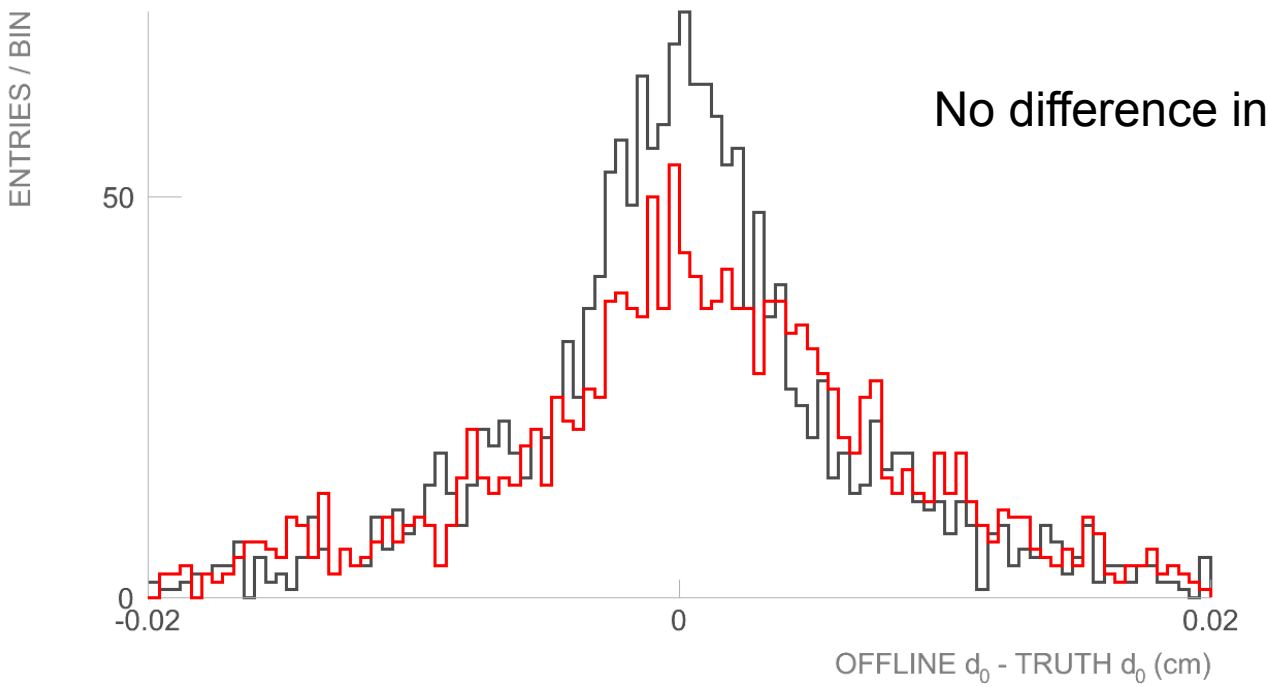

EFFICIENCY
1
0.5
0
-3.1377
-1.0459
1.0459
3.1377
TRUTH ϕ_0 (rad)


EFFICIENCY
1
0.5
0
-3.1377
-1.0459
1.0459
3.1377
TRUTH ϕ_0 (rad)

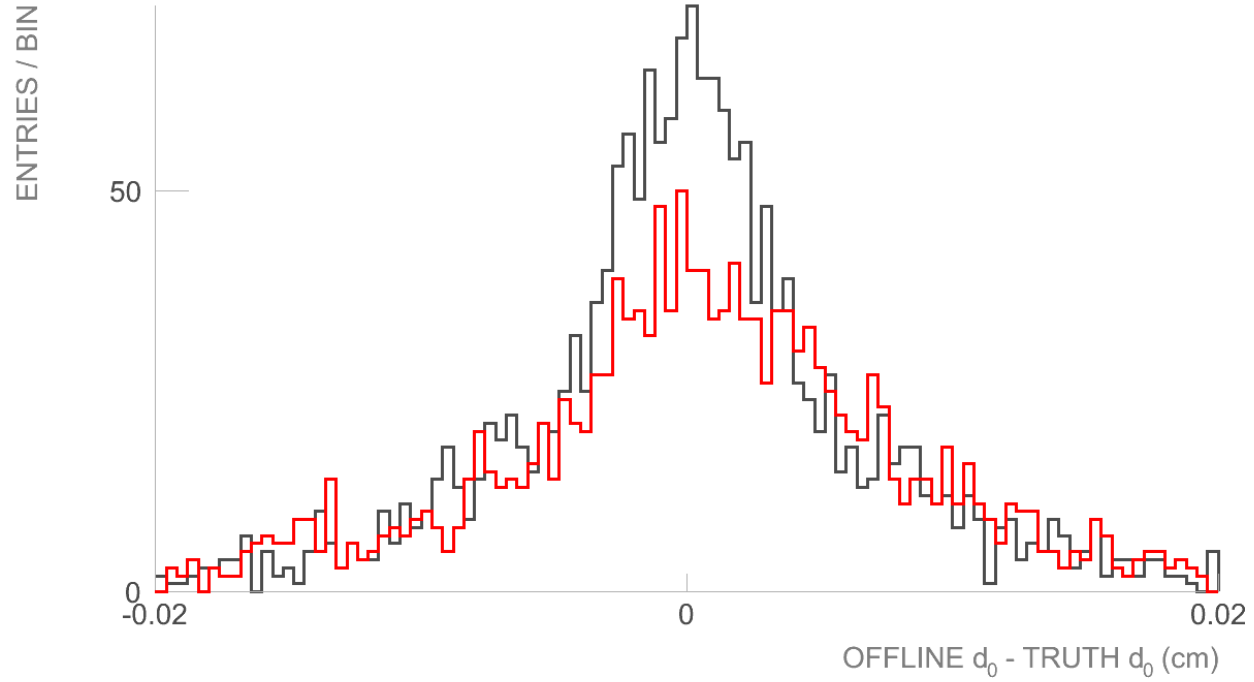


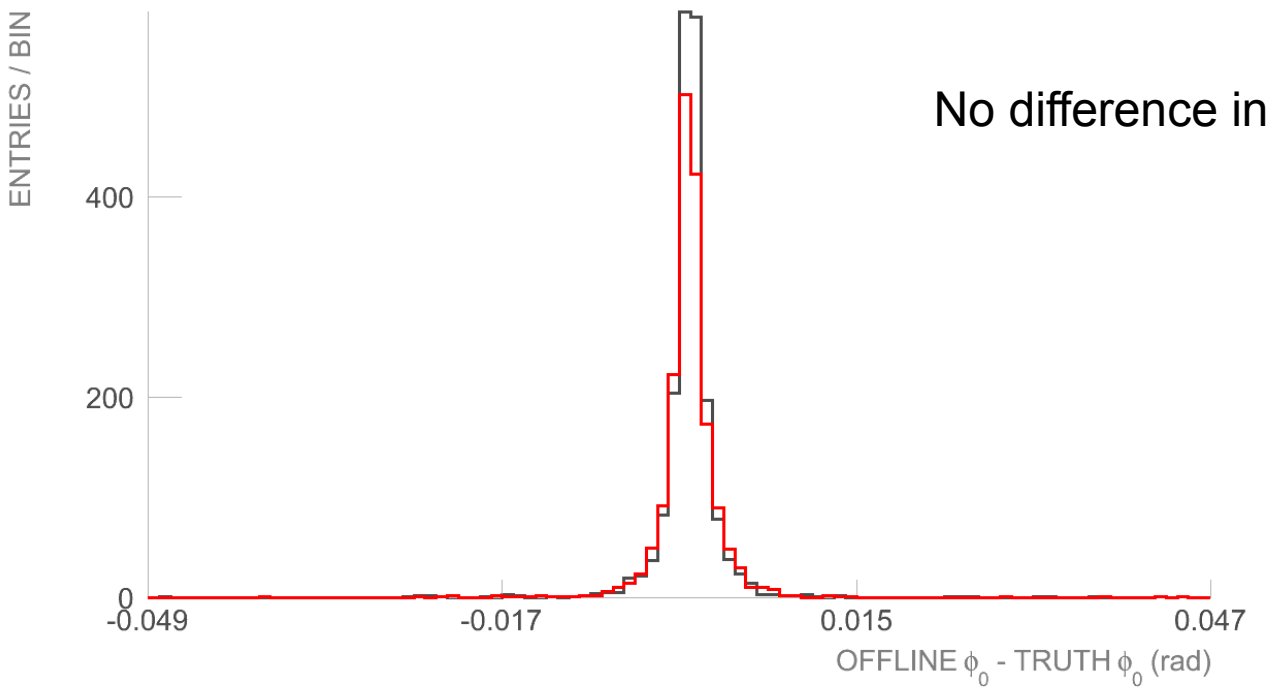


Efficiency drop due to ss-shift + Joe filtering

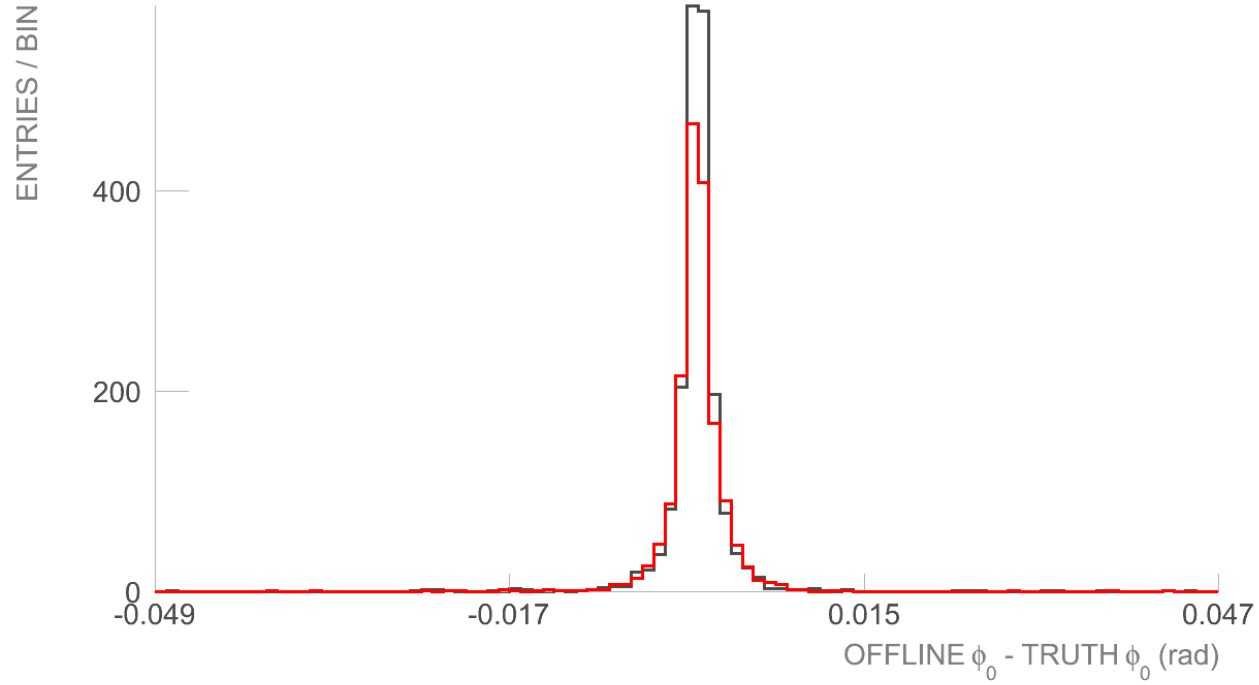


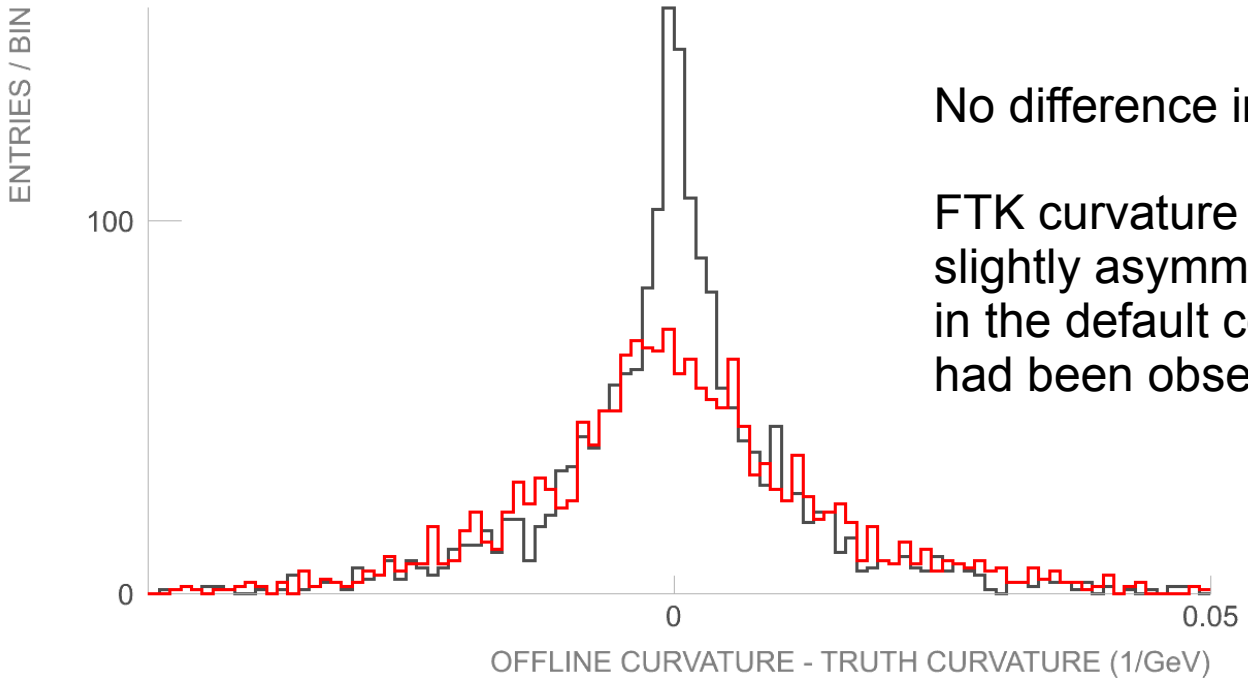
No difference in resolution!





No difference in resolution!





No difference in resolution!

FTK curvature resolution is slightly asymmetric (even in the default config). This had been observed before.