Recent updates and performance of LCFIPlus

Masakazu Kurata (KEK)
Tomohiko Tanabe (The University of Tokyo)
Taikan Suehara (Kyushu University)
Jan Strube (PNNL)
*Ryo Yonamine (Tohoku University)

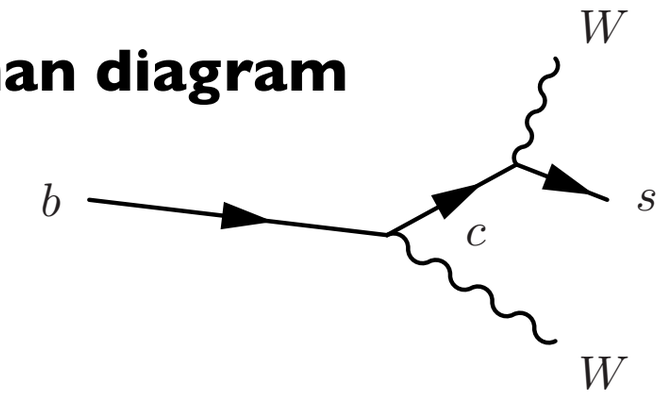
The work is based on the collaboration but RY is responsible for this presentation.

We would like to thank the ILD group, in particular the software group, for the development of the simulation framework and the data samples used in these studies.

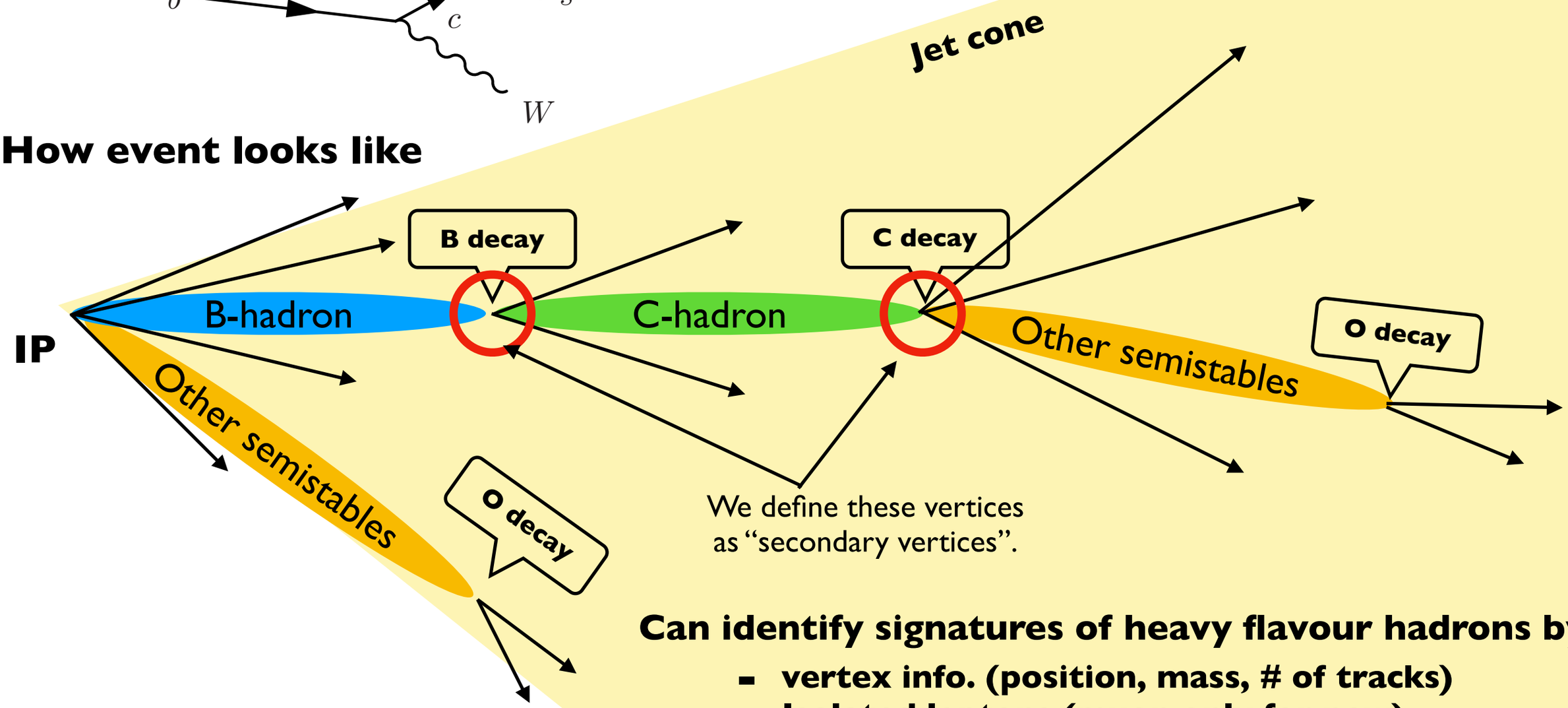
CLIC workshop 2019 @ CERN, 21-25 Jan. 2019

Our goal : b-tagging and c-tagging

Feynman diagram



How event looks like



Can identify signatures of heavy flavour hadrons by:

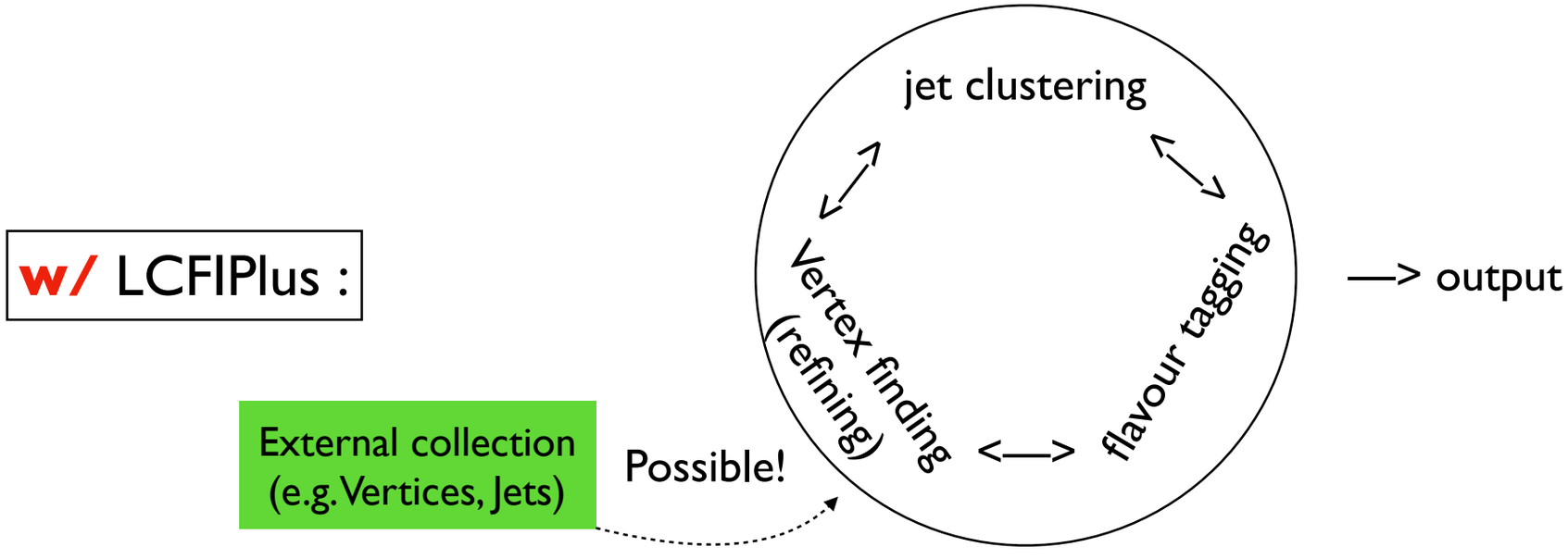
- **vertex info. (position, mass, # of tracks)**
- **Isolated leptons (muon only for now)**

Splitting secondary vertex tracks (e.g. by jet mis-clustering) would easily lose the signatures, especially in "jetty" environment.
—> Search secondary vertices first, then construct jets keeping the vertex structures.

LCFIPlus for the best b/c tagging

- **A framework for jet flavour identification.**
  - Integrates **Vertex finding**, **Jet clustering**, and **flavour tagging**.
  - Originated from LCFIVertex (https://arxiv.org/abs/0908.3019).
  - Composed of modular algorithms.
    - Gives flexibility to iterate or reverse the processes.

w/o LCFIPlus : jet clustering —> vertex finding —> flavour tagging —> output



- Typical flow with LCFIPlus (Vertexing first!) :
  "vertex finding —> (built-in) jet clustering
  —> vertex refining —>flavour tagging"

Outline :

1. Introduction of LCIFPlus processes
2. Recent updates
3. Plans & Summary

1. Introduction of LCFIPlus processes

Vertex finding

- **Starts from track selection**
  - Define unreliable tracks and will not try to associate them to any vertices.
- **Use Beam spot constraint for Primary vertex finding**
  - Beam spot constraint is powerful to distinguish non-primary vertices.
  - Beam spot size must be specified to use this constraint.
- **Use TearDown algorithm for Primary vertex finding**
  - Make a vertex using all tracks passed the track selection.
  - Compute chi2s from distances between the vertex and each track.
  - Remove tracks that give the highest contribution to the chi2.
  - Repeat until all the tracks satisfy a user-defined chi2 requirement.
- **For Secondary vertex finding, use tracks that are not associated to primary vertex.**
  - Make all possible track pairs, and requiring its invariant mass being less than 10GeV and sum of both track energies.
  - Apply V0 selection (vertex mass, vertex position etc.)
  - Attach additional tracks to the vertices if possible.

Vertex finding performance

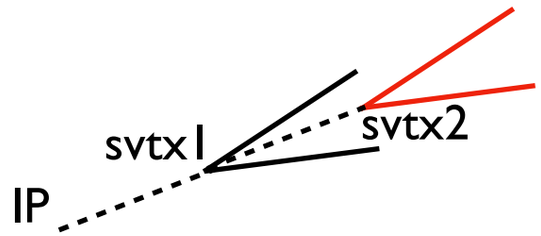
Fractions of tracks associated to three types of secondary vertices

Track origin	Primary	Bottom	Charm	Others
Total number of tracks	496897	258299	247352	56432
Tracks in secondary vertices	0.6%	57.5%	64.3%	2.5%
... from the same decay chain	—	56.6%	63.4%	1.9%
... from the same parent particle	—	32.2%	38.9%	1.2%

cat.1
cat.2
cat.3

ILD sample of bbbar events with sqrt(s)=91.2GeV.

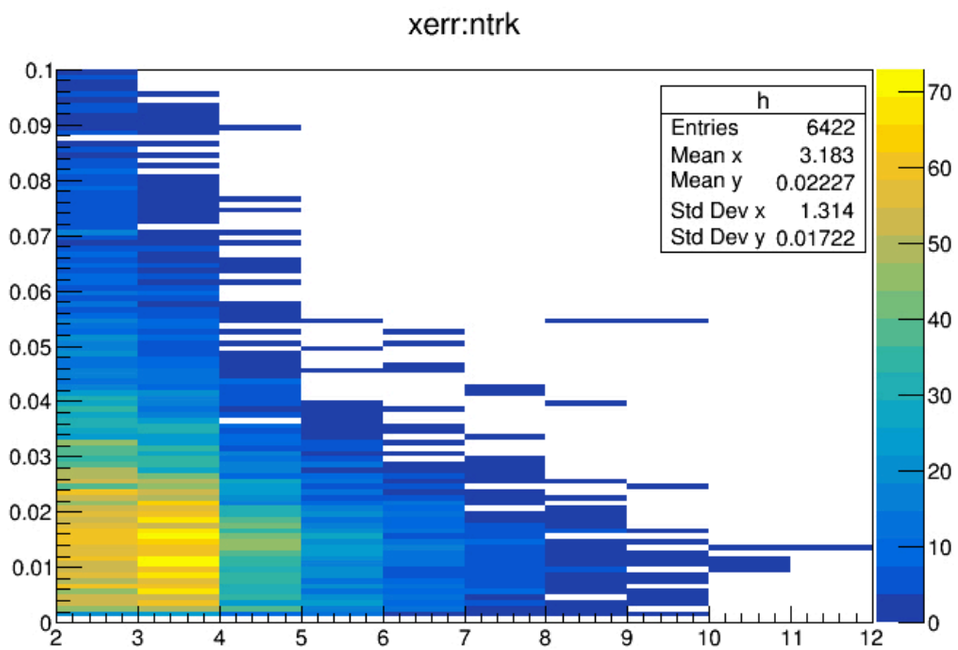
T. Suehara, T. Tanabe, "LCFIPlus: A Framework for Jet Analysis in Linear Collider Studies", NIM A 808 (2016) 109-116



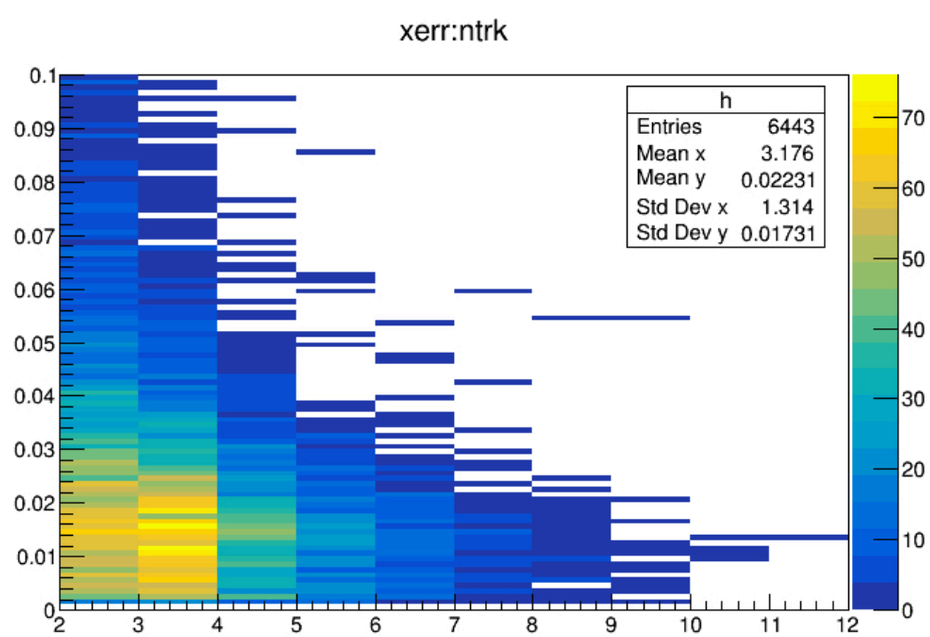
If a red track is associated to svtx1 or svtx2, this track is categorized into cat.2.
if a red track is associated to svtx2, this track is categorized into cat.3.
—> A drop from cat.2 to cat.3 indicates confusion of these two vertices.

Position (x) resolution w.r.t # of vtx tracks (Secondary vertex)

(w/o beam bkg overlay)



(w/ beam bkg overlay)

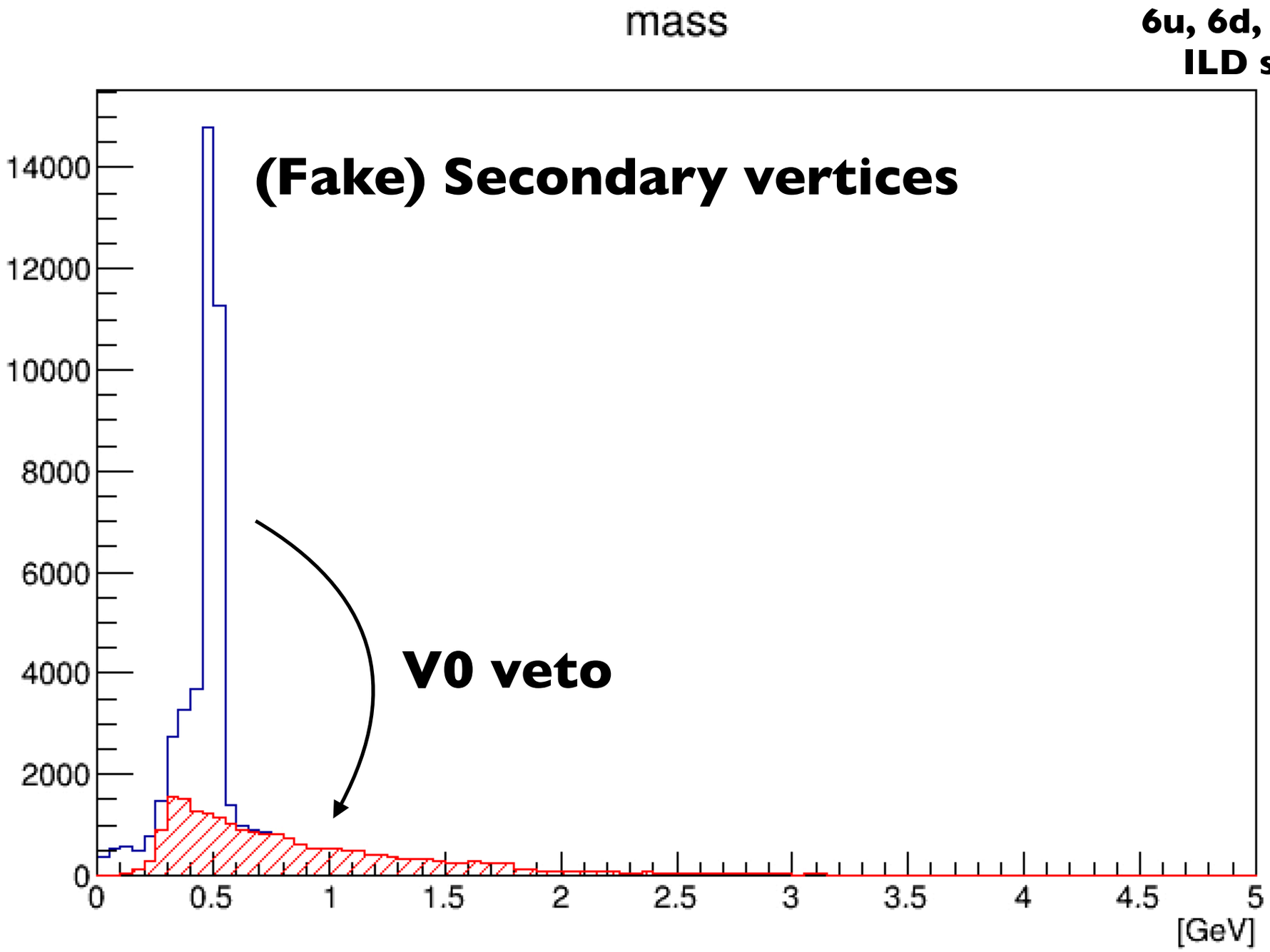


/s=500GeV, 6b sample

V0 rejection

We remove two-track vertices that are consistent with K_s, Λ^0, photon conversions (V0 vertices), because V0 vertices mimic B- or C-hadron vertices.

6u, 6d, 6s, √s=500GeV
ILD samples used



Jet clustering

- **Define jet cores by secondary vertices or leptons, and combine nearest jet cores until the required number of jets are obtained.**
  - We do not want to merge the jet cores any further. Will set α = 100 when 2 jet-cores are being combined in (modified) clustering algorithms.
- **Attach remaining tracks and neutral particles to one of the jet cores by using following jet algorithms.**

- **Built-in jet algorithms in LCFIPlus**
  - Durham
  - Kt
  - Valencia

  } **standard version**

  - DurhamVertex
  - KtVertex
  - ValenciaVertex

  } **modified version**

Intend to protect jet core structures —> Effective for multi-jet events

Modified y value (DurhamVertex) :

$$Y(i,j) = \frac{2\min(E_i, E_j)^2(1-\cos\theta_{ij})}{Q^2} + \alpha$$

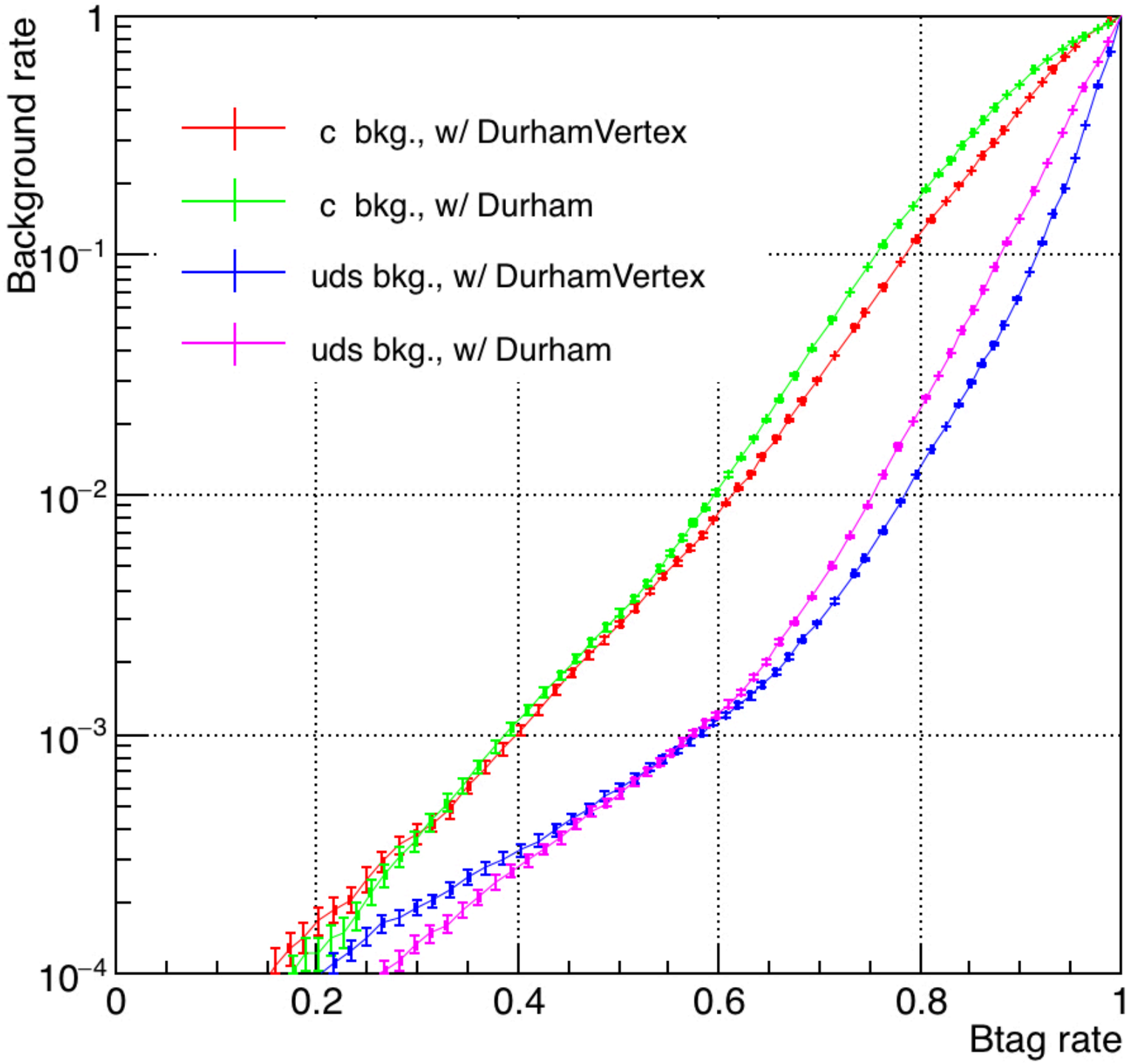
E_i, E_j : Jet energy
θ_{ij} : angle b/w Jets
Q : $\sqrt{s}$
α=0, 100

- **Jet collections produced by external packages can also be used instead of using jet clustering in LCFIPlus.**

However built-in jet algorithms that use vertex information are recommended.

Jet clustering effect on flavour tagging

Comparison between Durham and DurhamVertex

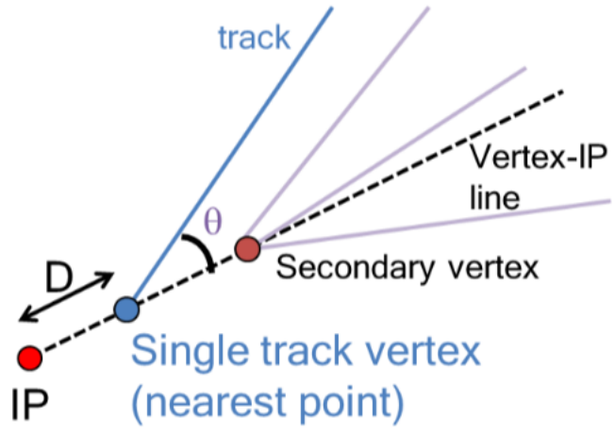


6b, 6c, 6q
√s=500GeV
ILD(l5) sample used

Jet vertex refiner

- **Re-vertexing but now using jet information**
  - More than one secondary vertex in a jet implies a b-jet.
  - Useful for b-c separation.
  - Try to improve the efficiency of secondary vertex reconstruction.

- **Pseudo vertex : Single track vertex**
  - If one secondary vertex is found in a jet and if there is a track whose trajectory comes near a point collinear to the primary and secondary vertices, it is defined as pseudo vertex, unless the track is tagged as a primary track.



- **For each vertex in a jet, compute chi2 again to all tracks and check if there is any possibility to refine vertex reconstruction.**

Jet vertex refiner performance

w/o vertex refiner
(w/ vertex refiner)

	(# of vtx , # of pseudo-vtx)	6 b	6 c	6 q
b-jet signature	(2 , 0)	22.35 % (23.92%)	0.45% (0.42%)	0.05% (0.06%)
	(1 , 1)	2.18% (17.78%)	0.15% (1.42%)	0.00% (0.06%)
c-jet signature	(1 , 0)	53.09% (36.84%)	42.67% (41.94%)	0.97% (1.20%)

6b, 6c, 6q √s=500GeV
ILD(l5) sample used

The main effect of vertex refining is the recovery of vertices and pseudo-vertices: as a result, b jets migrate into the b-like category (2,0) and (1,1) from the c-like category (1,0), which helps with b/c separation.

Flavour tagging

- **Based on multi-variate analysis ("BoostedDecisionTrees")**
  - input variables : impact parameters, track multiplicity, vertex mass, etc.

- **For efficient training, 4 jet-categories are used.**
  - 0 vertex jet —> light flavour like
  - 1 vertex jet —> c like
  - 2 vertex jet (pseudo vertex = 1) —> b like
  - 2+ vertex jet —> b like

- **We typically offer training samples for different energies and different jet multiplicities. For the best performance, the analyst should compare the different weight files.**

  examples:
  - 91GeV, 2b, 2c, 2q sample
  - 500GeV, 6b, 6c, 6q samples
  - 1TeV, 6b, 6c, 6q samples

2. Recent updates

Update list

1. **Migrated to ROOT6. ROOT >= 6.08.00 required.**

2. **Adaptive Vertex Fitting** Under testing
   - currently relatively strict track selection is applied to prevent spoiling vertex reconstruction with fake tracks.
   - try to loosen the track selection while keeping fake track rate low by introducing a weight.
3. **BNess for better track selection** Under testing
   - identifying tracks from B-hadron using MVA.
4. **Vertex Mass Recovery for better B/C separation** Under testing
   - Pi0 reconstruction
5. **Fix related to the IP smearing.**
   - Some MVA variables assumed IP=(0,0,0).
6. **Position errors on primary vertex.**
   - Fit parameters for primary vertex slightly modified (—> fraction of fitting failures on primary vertex was reduced. No changes for secondary vertex finding.).

Adaptive Vertex Fitting in LCFIPlus

(Implemented as an option in secondary vertex finding)

- **Concern is fake track**
  - e.g mis-measured tracks, mis-assigned tracks to a vertex
  - Usually remove these tracks requiring chi2 to be small.
  - A tight selection to reduce fake tracks in "jetty" environment reduces vertex finding efficiency.
- **Compute following weight(W) for track 'k' and each vertex 'n'.**
  - χ_{cut} and T are control parameters to be defined by users.
  - When W > 0.5, the track 'k' will be associated to the vertex 'n'
  - Will reduce track mis-assignment probability even in "jetty" events.

$$w_{nk} = \frac{e^{-\chi^2_{nk}/2T}}{e^{-\chi^2_{\rm cut}/2T} + \sum_{i=1}^{N} e^{-\chi^2_{ik}/2T}}$$

Let's see rough idea :

- T—> 0 case, W—>1 (identical to standard way.)
- T—> ∞ case, W —> 1 / (1+N) with N being total # of vertex candidates.
- Computed χ^2_{nk} >> χ^2_{cut} case, W => always small
- Computed χ^2_{nk} << χ^2_{cut} case, W => large only when no other good candidates.
- **Roughly speaking, a track has several good vertex candidate to be associated, we put small weights, which means we recognise the track as "ambiguous".**

Impact of Adaptive Vertex Fitting

- Common parameters are set at same values for comparison
- Same event sample(qqHH sample@500GeV) 19889 events
- 6 jet clustering, jet matching with MCtruth is performed
- Num. of jets with vertex:

method	bjet with 2vtx	bjet with 1+1vtx	bjet with 1vtx	total
DBD LCFIPlus	10581	9104	12847	32532
AVF	13190	6576	13233	32999

- Total jets with vtx: ~1.4% increased
  - Jets with 2vtx: ~22% increased → good for bjet ID!
  - Jets with 1vtx: ~3% increased → good for uds jet separartion!
- Fake track rate per vtx: how many fake tracks contaminate on

method	bjet with 2vtx	bjet with 1+1vtx	bjet with 1vtx
DBD LCFIPlus	0.029 ± 0.001	0.013 ± 0.0012	0.055 ± 0.002
AVF	0.025 ± 0.001	0.012 ± 0.0013	0.055 ± 0.002

Vertex Mass Recovery

(Implemented as an algorithm like vertex finding, jet clustering, etc.)

- **Vertex mass is one of variables that distinguish B-hadron and C-hadron.**

- **Vertex mass can be computed by charged tracks only, and thus is typically smaller than its original mass.**

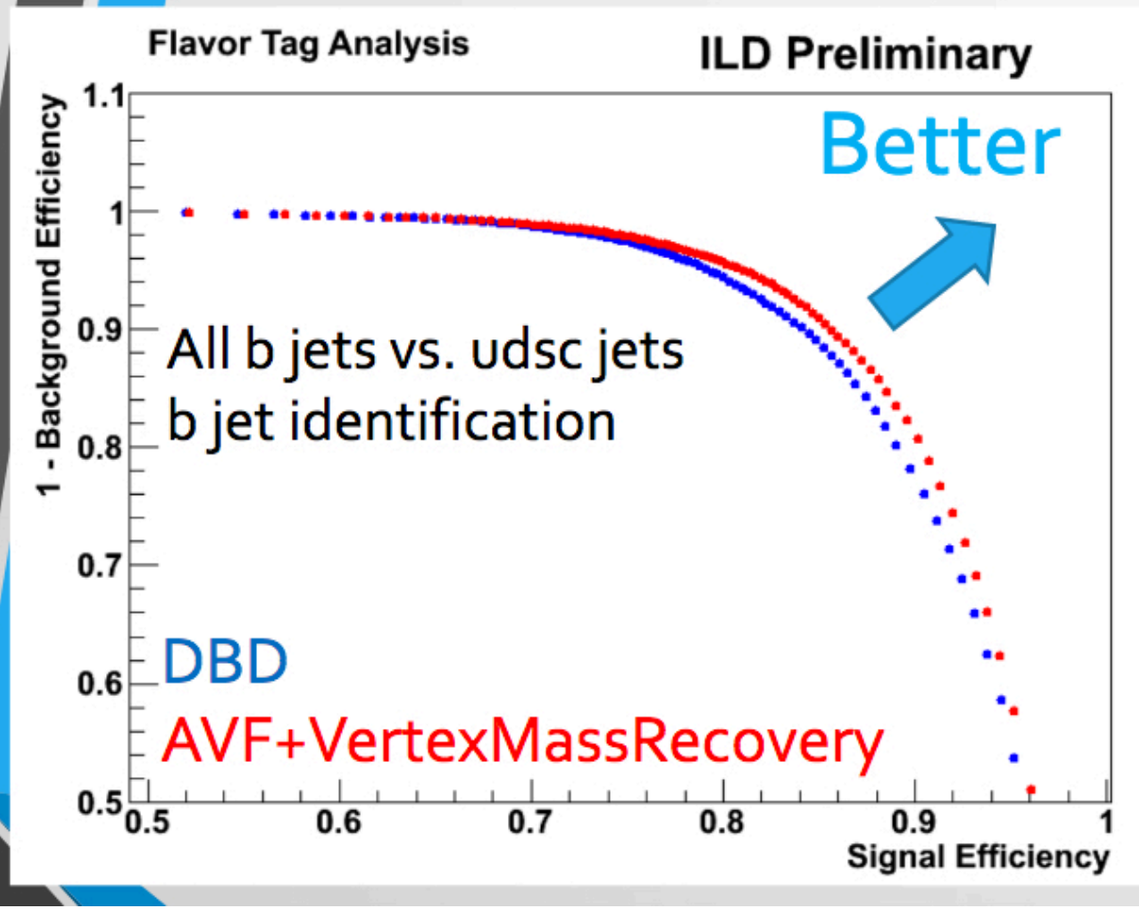
- **If Pi0 is reconstructed as a part of vertex, adding the mass helps to recover the mass. —> Try to find a best assignment to a vertex using multivariate analysis (vertex mass, vertex track PIDs)**

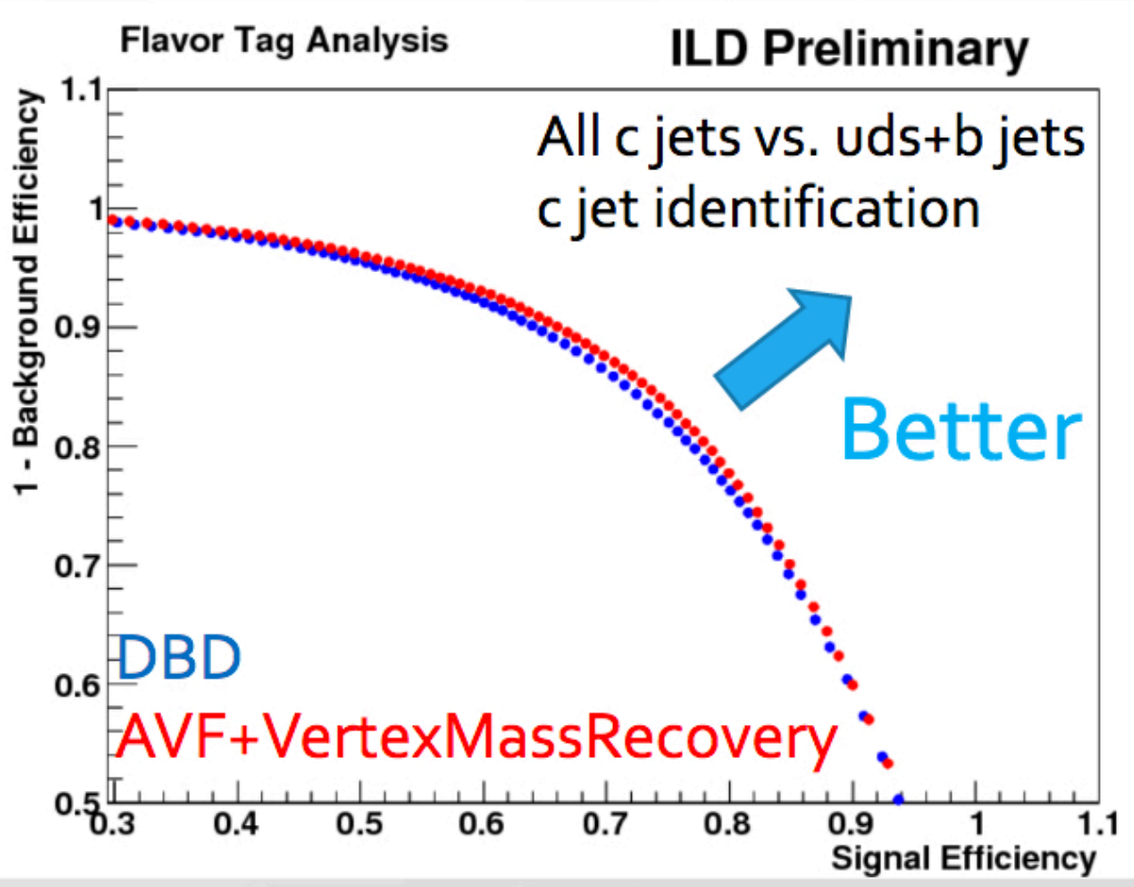
Adaptive Vertex Fitting, VertexMassRecovery

M. Kurata, LCWS2017

Impact on Flavor Tagging Efficiency

- 6f samples coming from ZZZ events@500GeV
- Compare with ROC curve

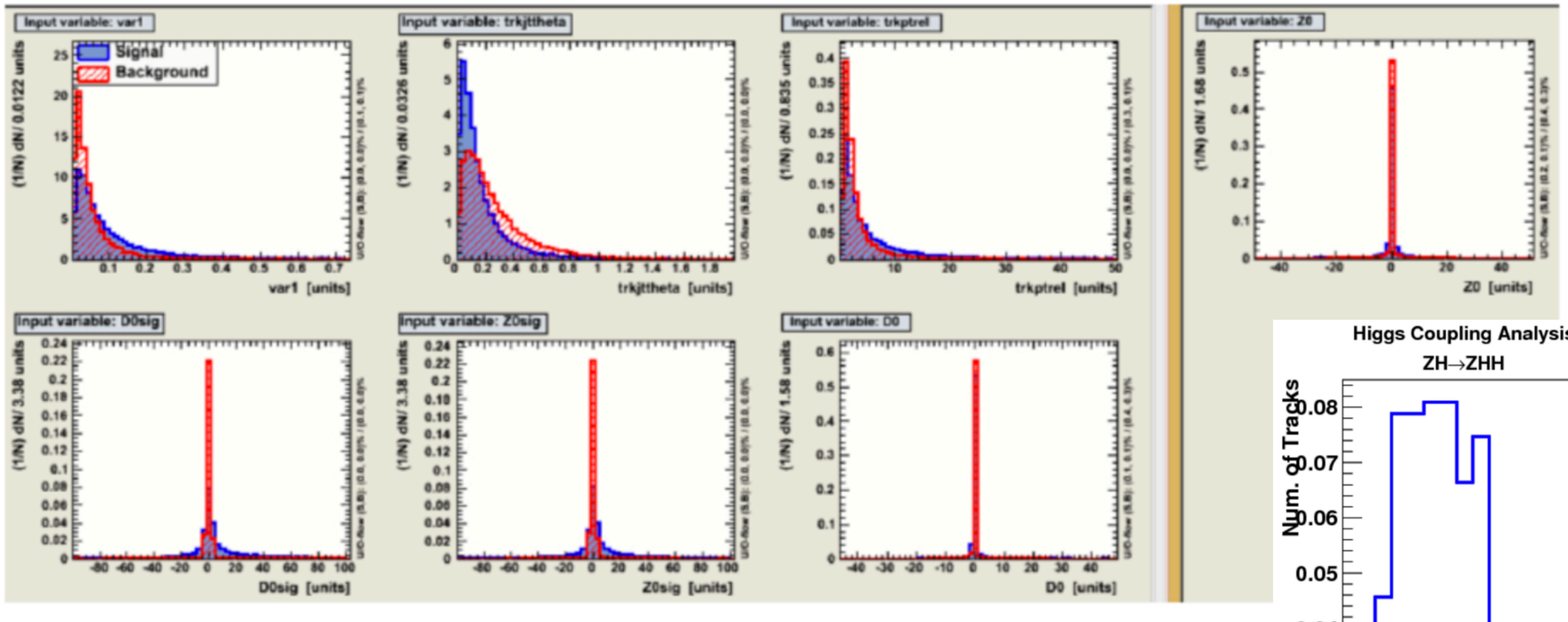


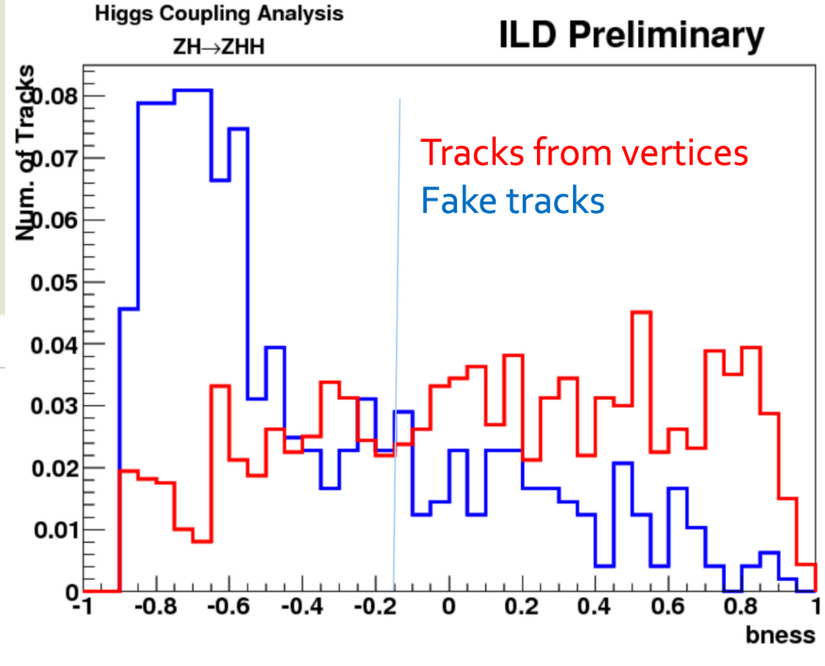


TRACK MVA(BNESS) Experimental feature in VertexRefiner

M. Kurata, LCWS2017

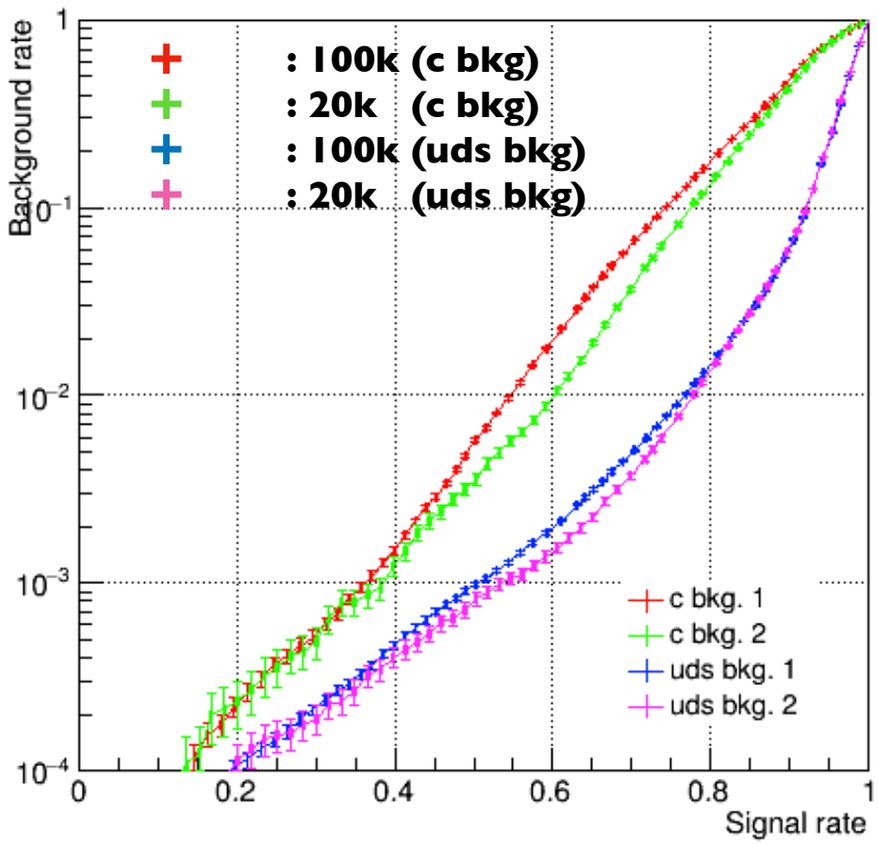
- To identify track which comes from heavy flavor particle
  →using MVA
  - Signal: tracks which come from B mesons or B baryons
  - Background: tracks produced in hadronization process
- Most significant tracks with both plus and minus signed impact parameters in a jet are collected
  - Significance: $sig = \sqrt{(\frac{d_0}{\sigma})^2+(\frac{z_0}{\sigma})^2}$

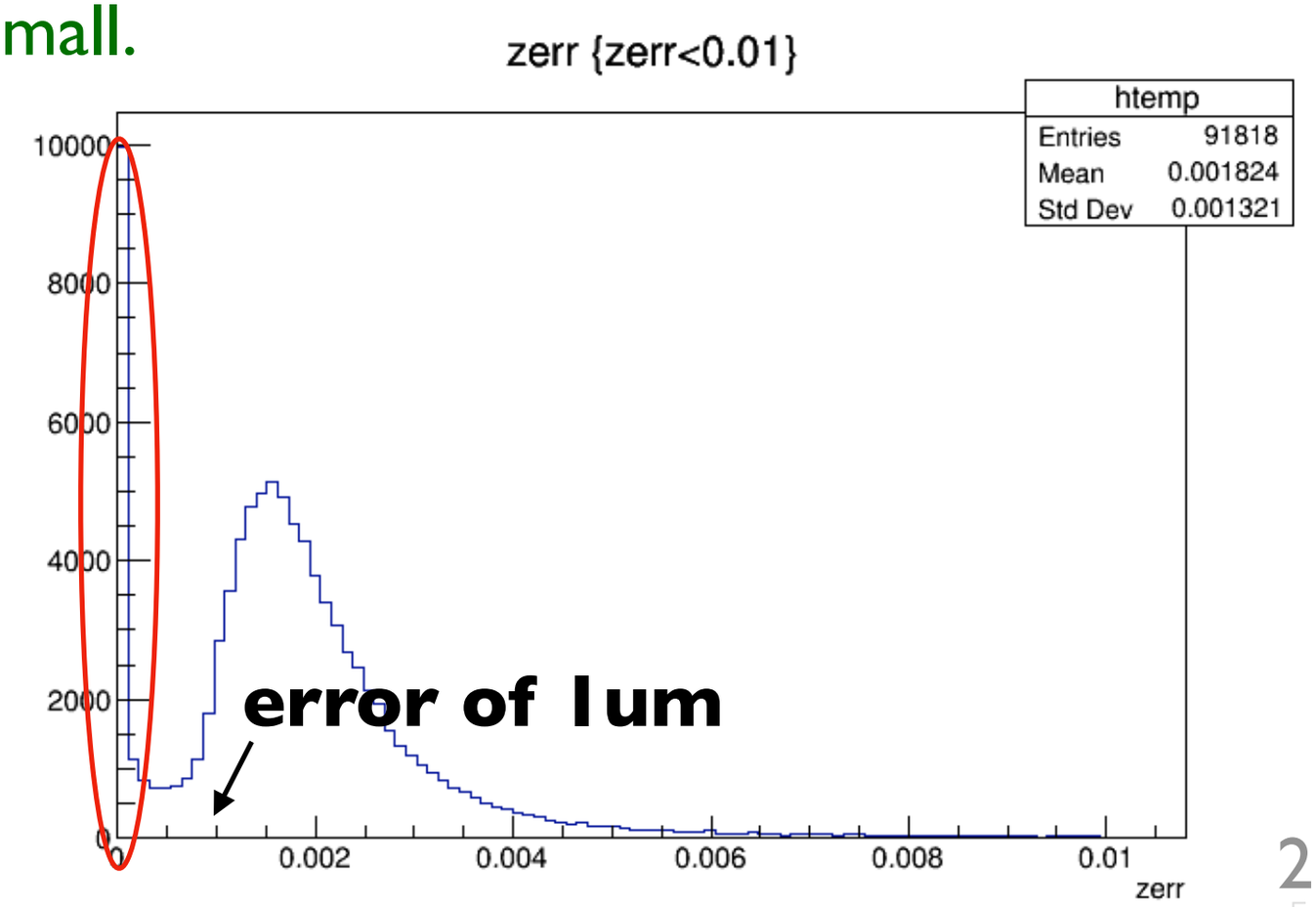




Issues in 2018

- **Strange dependency on statistics of training data: Large statistics gives worse performance.**
  - No overtaining was seen.
  - This was finally turned out that the problem came from a bug where IP=(0,0,0) was assumed, while IP smearing has been introduced in ILD simulation.
- **Failures on primary vertex fitting**
  - Originally primary vertex fitting was not well cared about (The highest priority was the secondary vertex finding!)
  - We got a feedback from a user that the error on primary vertex position was sometimes too small.

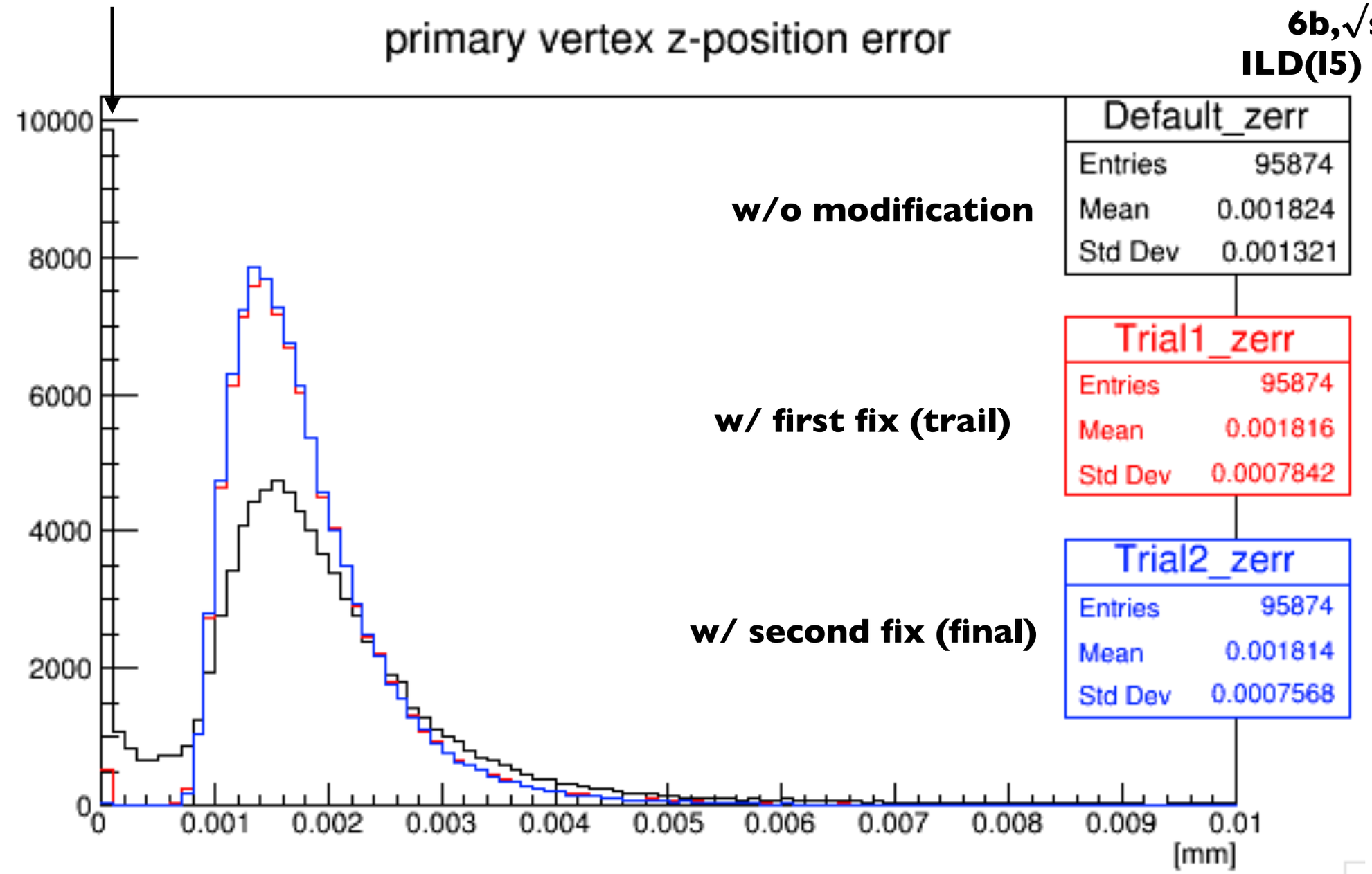




Primary Vertex Fitting

Fitting range has been increased (only for primary vertex) so that the minimization more likely completes.

fitting failures



Looking at the entires at 0,

- **Blue (second fix) is better than red (first fix).**

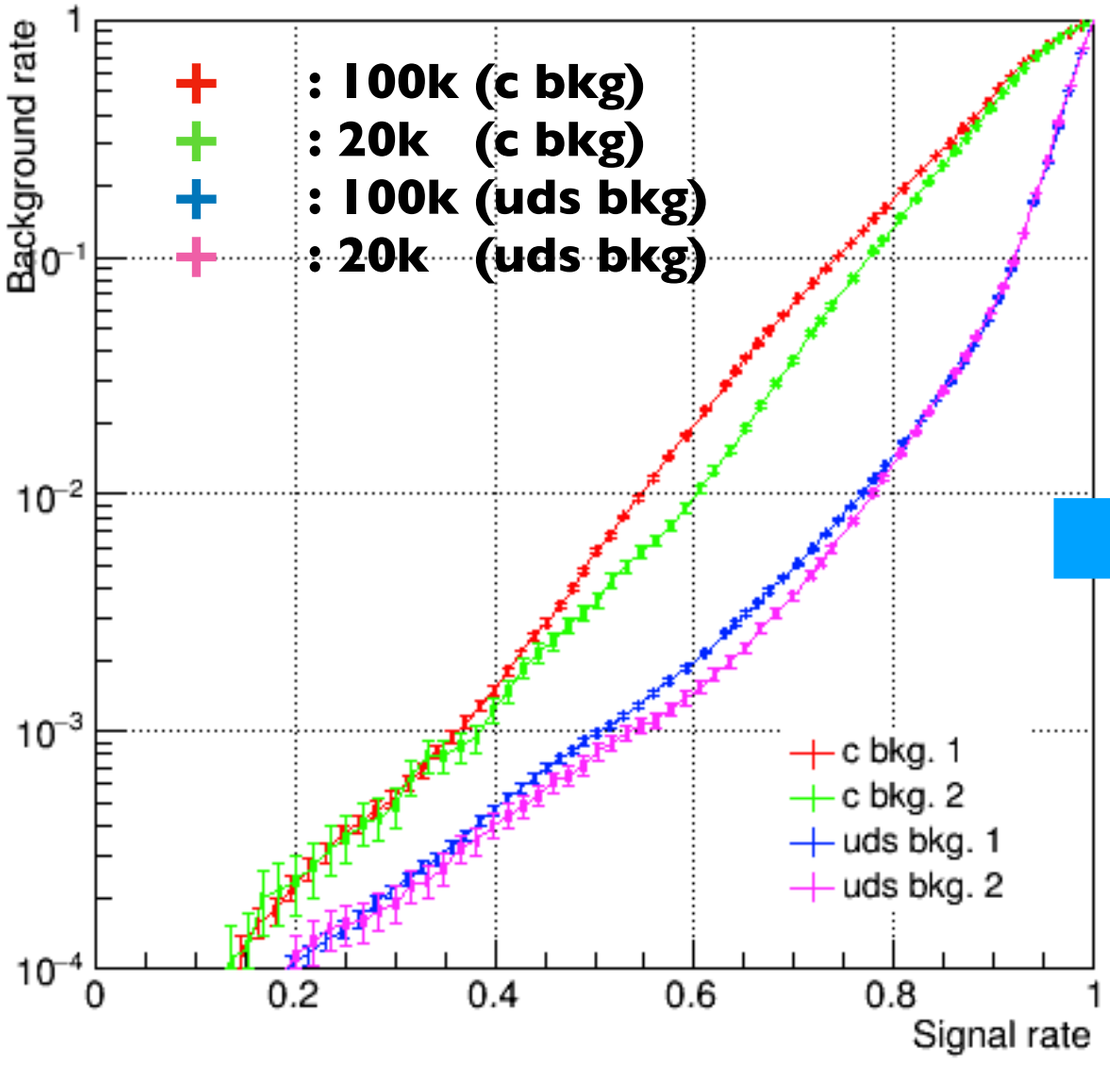
Accommodating IP smearing

Previously a training with 100k sample gave worse performance than the one with 20k sample.

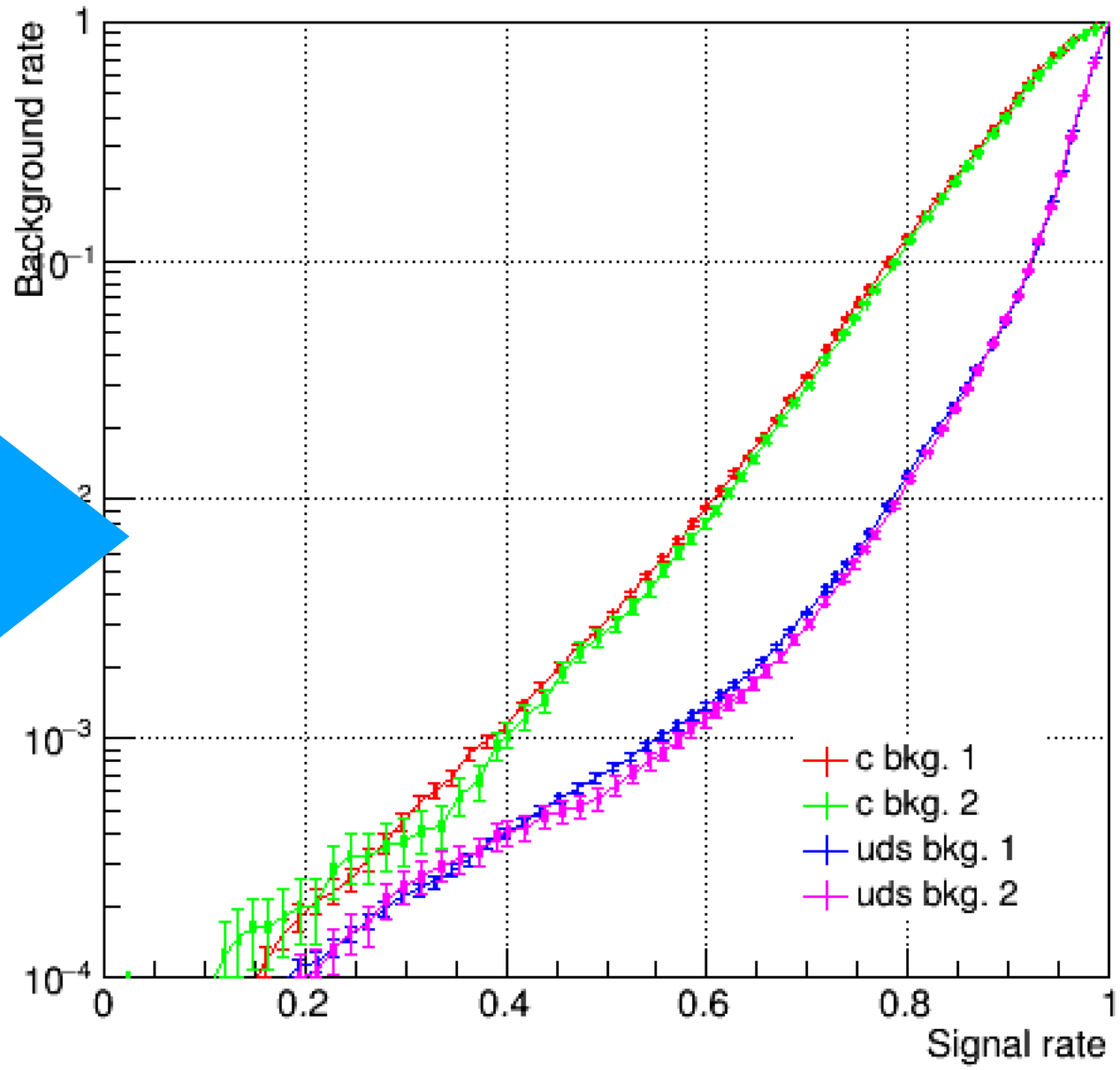
b-tagging performance

6b, 6c, 6q √s=500GeV
ILD(I5) sample used

w/o modification



w/ new fix



Still 100k is slightly worse than 20k but almost consistent now.

(These results were produced with NNodeMax=8, which has been deprecated in TMVA and recommended to replace to MaxDepth.
When MaxDepth=6 is used, the difference become negligible.)

3. Plan & Summary

ToDo/Ideas

Our plans to improve the performance in 2019!

- **Try additional information into MVA**
  - dEdx, TOF, etc
- **Try NN in flavour tagging**
  - BDT has been used because it works decently without much effort.
- **Optimization for vertex charge measurement**
  - Loosen the vertex quality cut would improve the performance.
  - Comprehensive study done by S. Bilokin (PhD thesis:https://tel.archives-ouvertes.fr/tel-01826535/document)

Notes from the developers

- **Use LCFIPlus v00-07**
  - [d0/z0][b/c/q]prob can be replaced to [d0/z0][b/c/q]prob2 as an input variable for flavour tagging.
  - MaxDepth=6 should be used instead of previous NNodeMax.

- **Consideration for "jetty" events**
  - built-in jet clustering (e.g. DurhamVertex) is recommended.
  - Re-optimization of the track quality selection and the vertex quality selection may help to obtain better performance.
  - New features (Adaptive Vertex Fitting, Vertex Mass Recovery) also may help.

Summary

- **iLCFIPlus is under active development.**
- **Our post-DBD improvemnts include:**
  - Adaptive vertex fitting
  - Vertex mass recovery
- **More recently, we are working to adapt with new ilcsoft software.**
- **Plans to improve the flavour tagging performance by incorporating more reconstruction information.**
- **We welcome your feedback and look forward to discussing with you!**

Backup

Reference

- **Paper**
  - T. Suehara, T. Tanabe, "LCFIPlus: A Framework for Jet Analysis in Linear Collider Studies", NIM A 808 (2016) 109-116

- **Presentations (for developments before 2018)**
  - By T. Suehara, https://agenda.linearcollider.org/event/7520/sessions/4400/#20170425
  - By M. Kurata, https://agenda.linearcollider.org/event/7645/contributions/40125/

- **Git repository**
  - https://github.com/lcfiplus/LCFIPlus

Jet Clustering: algorithms

- Durham with beam rejection

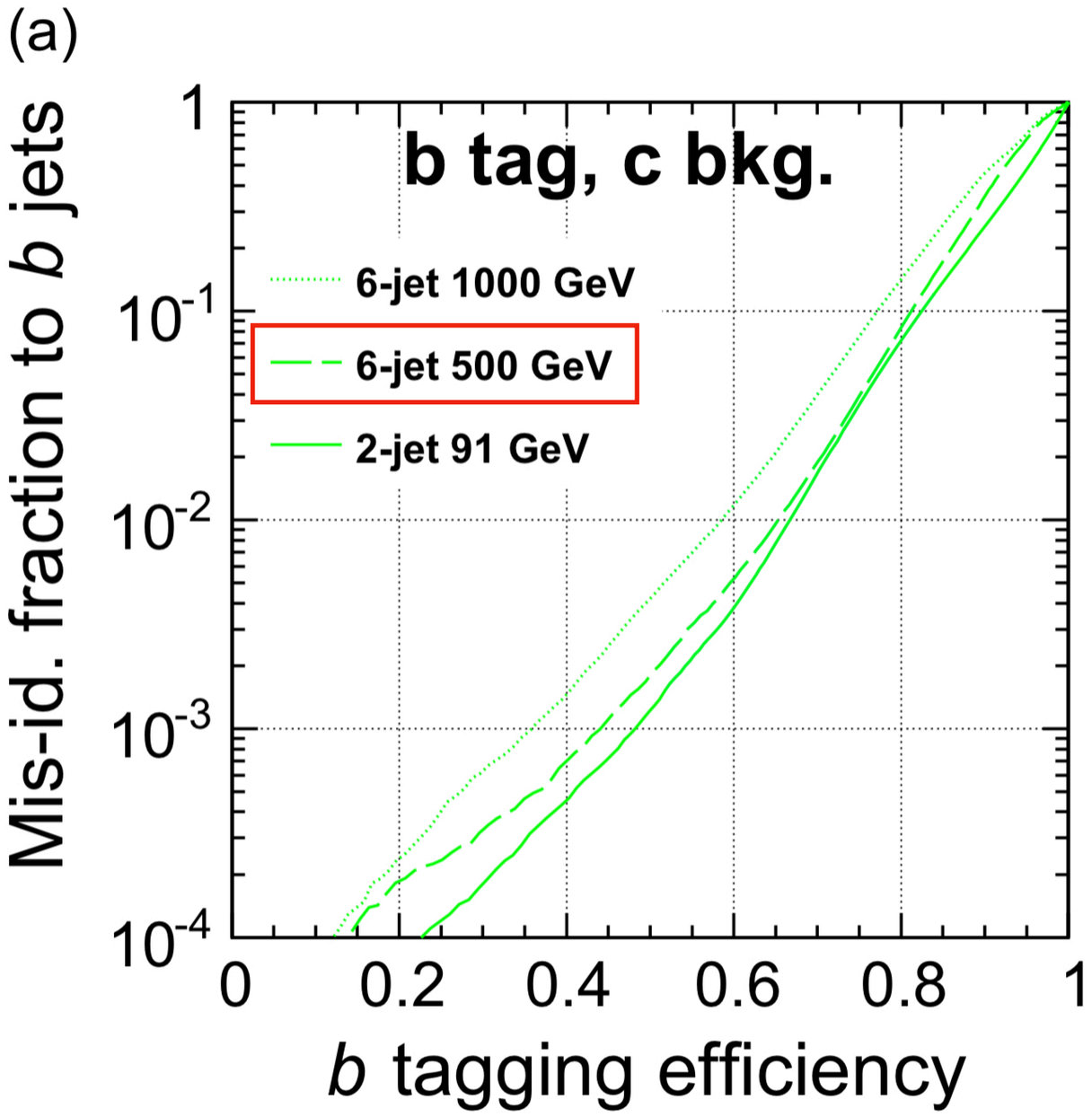
$$y_{beam} = 2E^2\alpha^2(1-\cos\theta)/E_{vis}^2$$

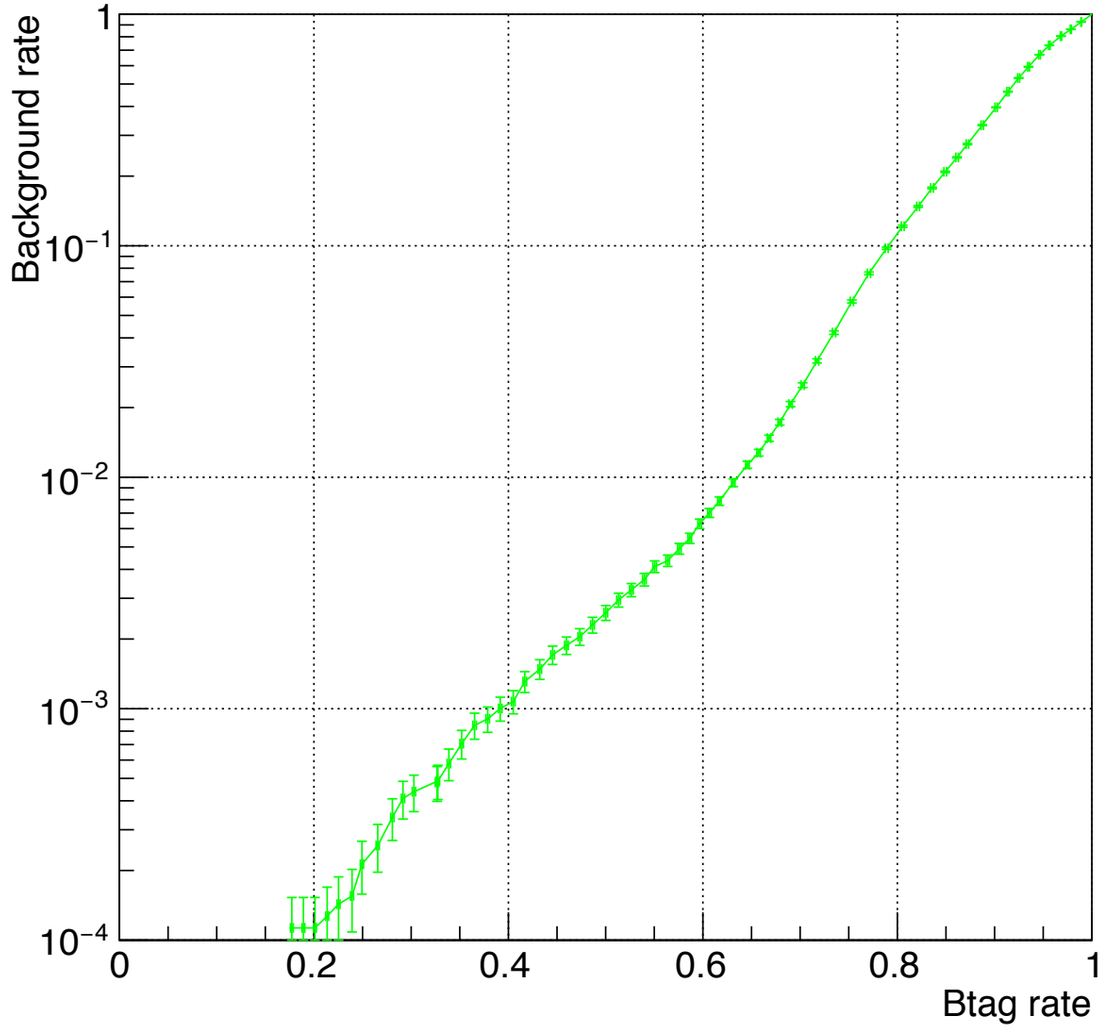
- Plain Durham (still available, of course)
- kT algorithm
  - No need to run it outside any more
- Valencia algorithm
  - Intermediate algorithm of Durham ant kT

$$d_{ij} = min(E_i^{2\beta}, E_j^{2\beta})(1-\cos\theta_{ij})/R^2$$

$$d_{iB} = p_T^{2\beta}$$

Comparison with previous result [1]



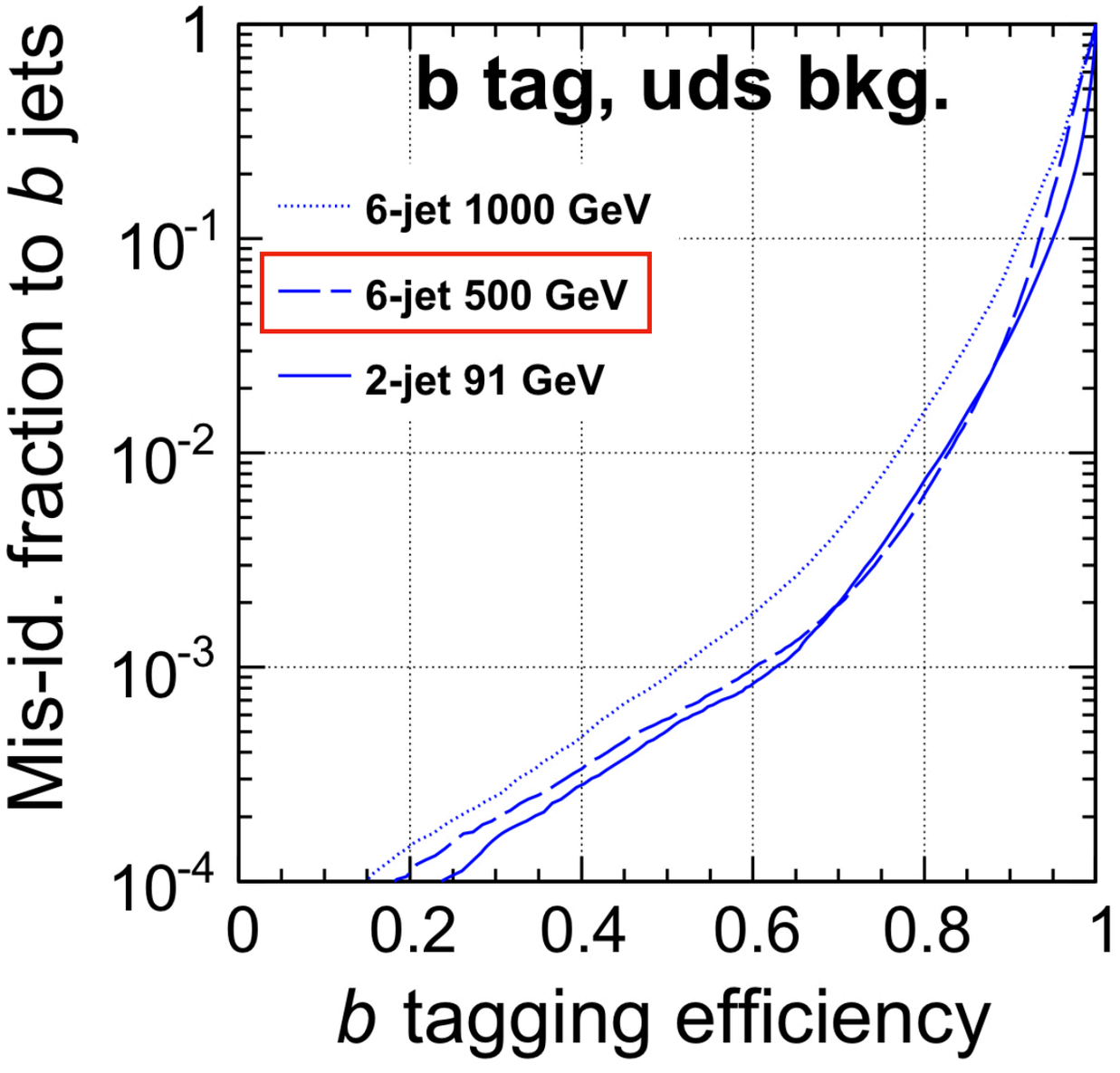


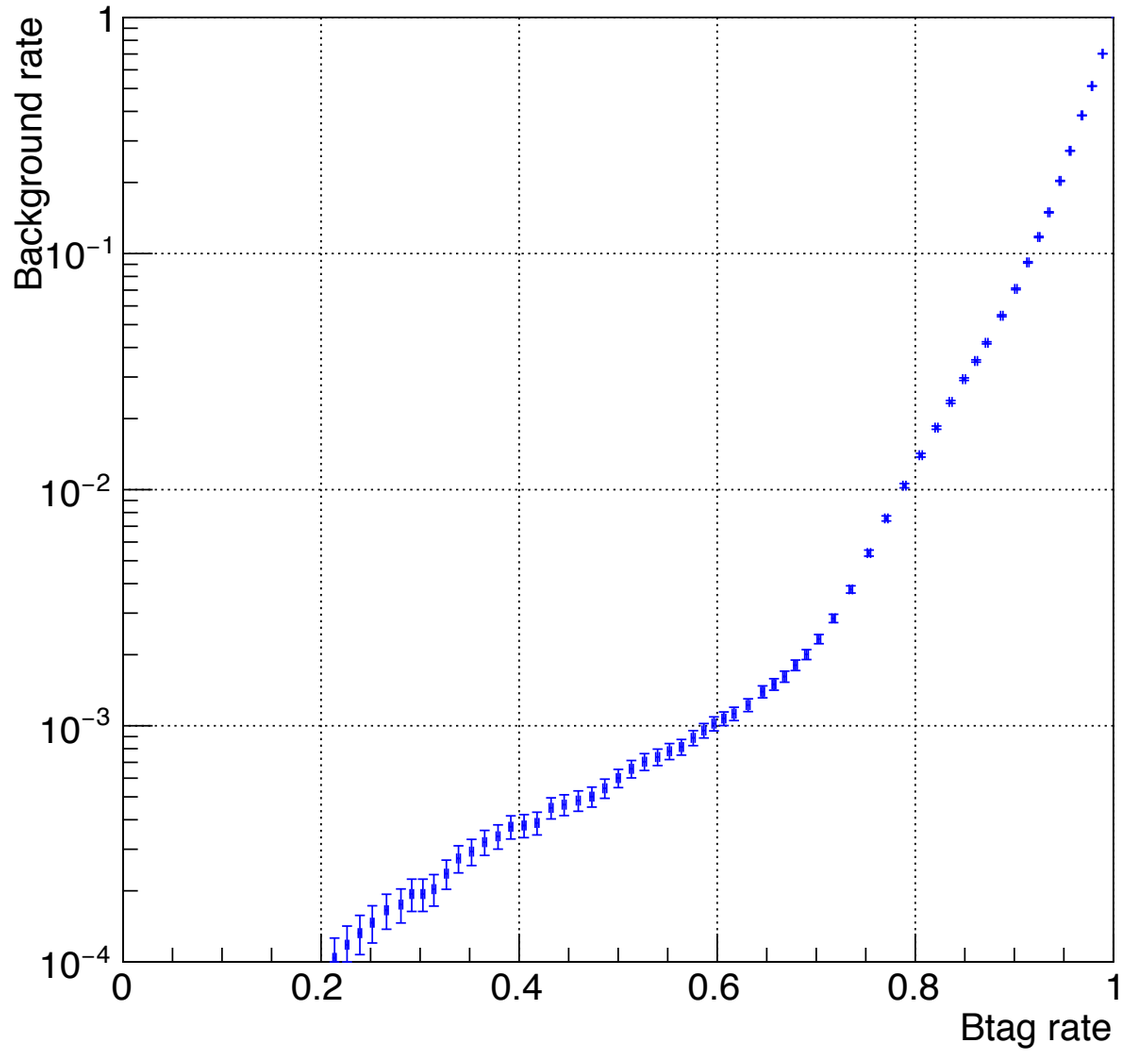
[1]: LCFIPlus: A framework for jet analysis in linear collider studies, Taikan Suehara, Tomohiko Tanabe

- ILD DBD sample (no IP smearing, but emulated in LCFIPlus) used. ~20k for each sample.
- Beam spot constraint (639nm,5.7nm,91.3um) as written in [1].
- v00-07 used. MVA param. Maxdepth=6

Consistent result.

Comparison with previous result [1]



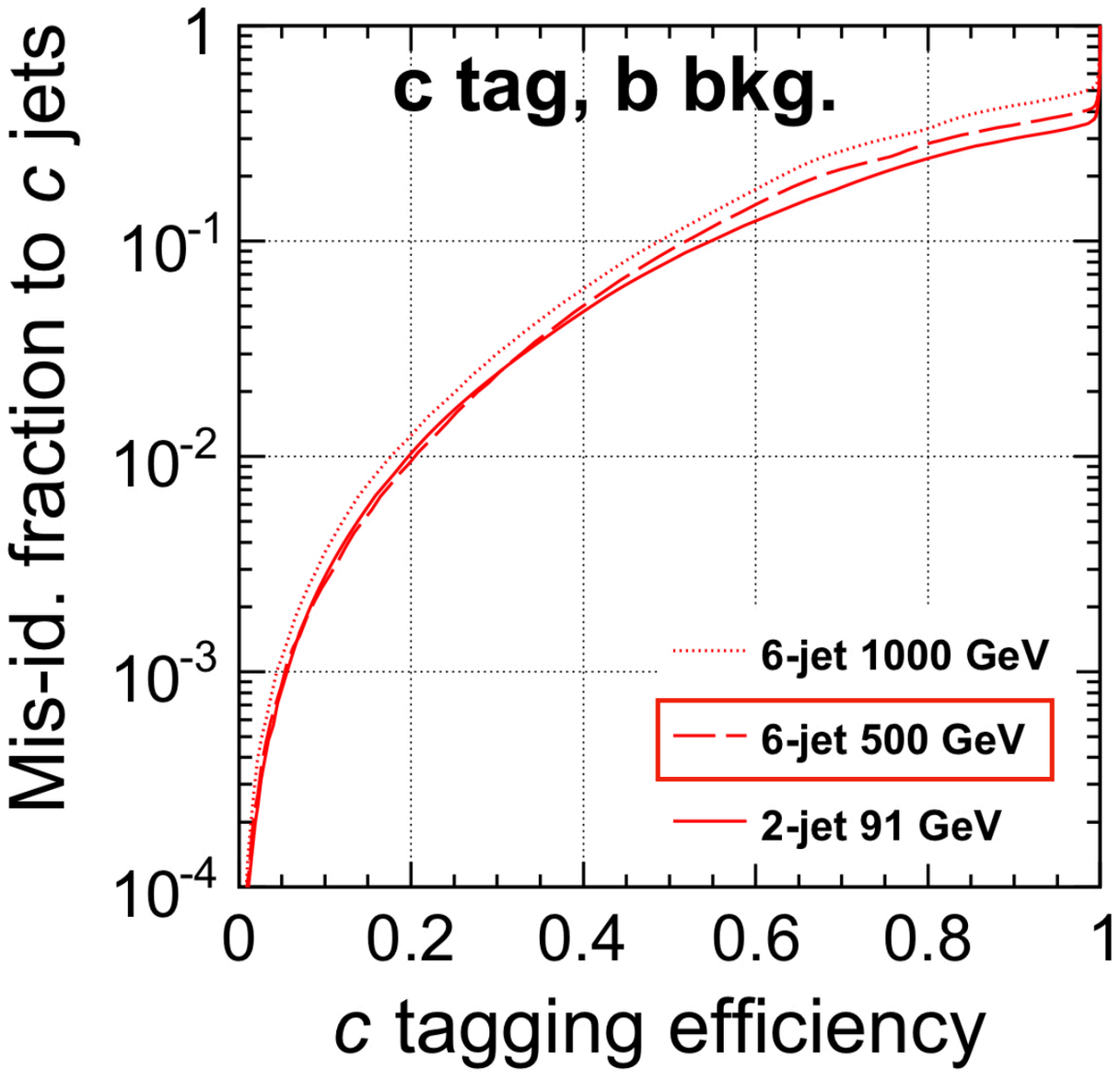


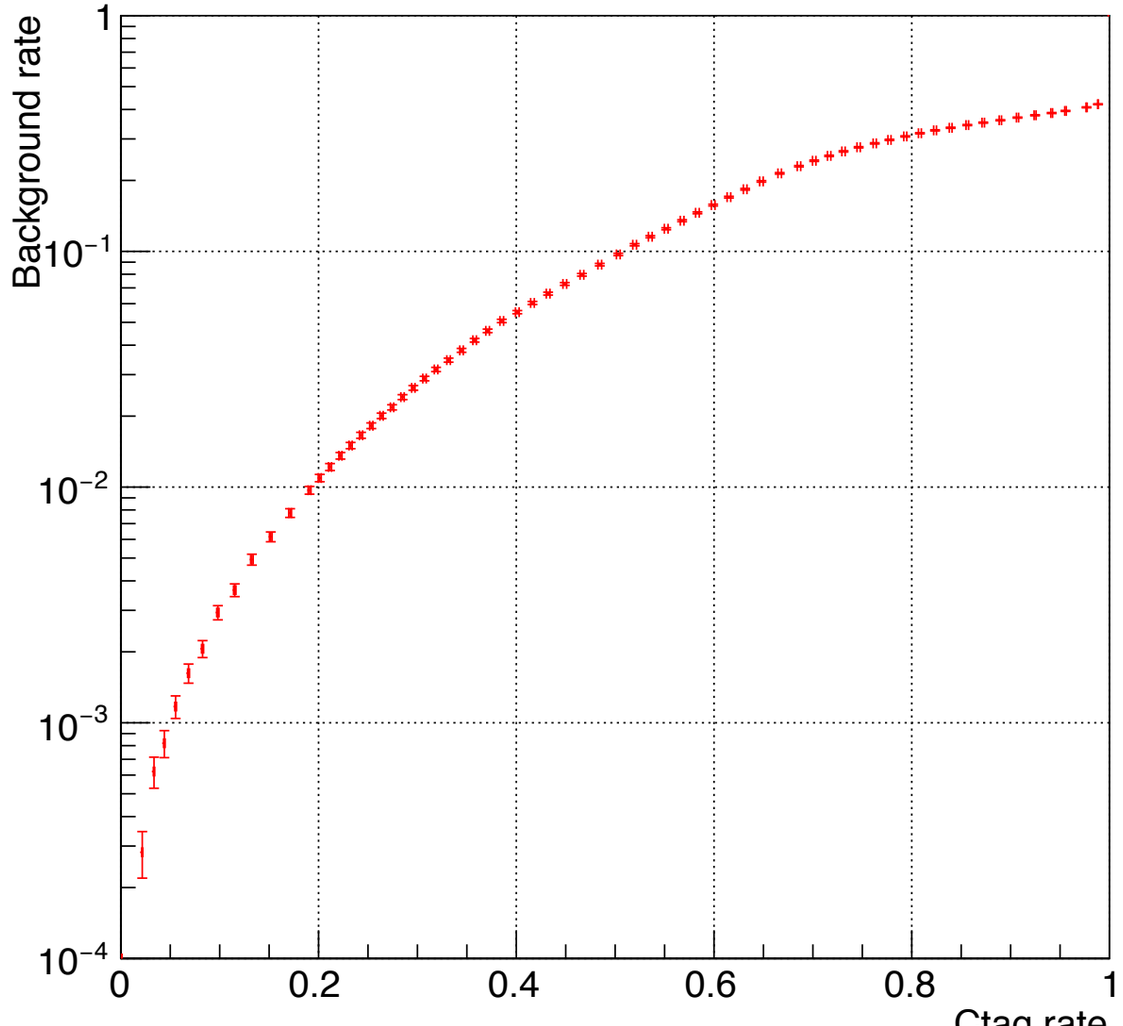
[1]: LCFIPlus: A framework for jet analysis in linear collider studies, Taikan Suehara, Tomohiko Tanabe

- ILD DBD sample (no IP smearing, but emulated in LCFIPlus) used. ~20k for each sample.
- Beam spot constraint (639nm,5.7nm,91.3um) as written in [1].
- v00-07 used. MVA param. Maxdepth=6

Consistent result.

Comparison with previous result [1]



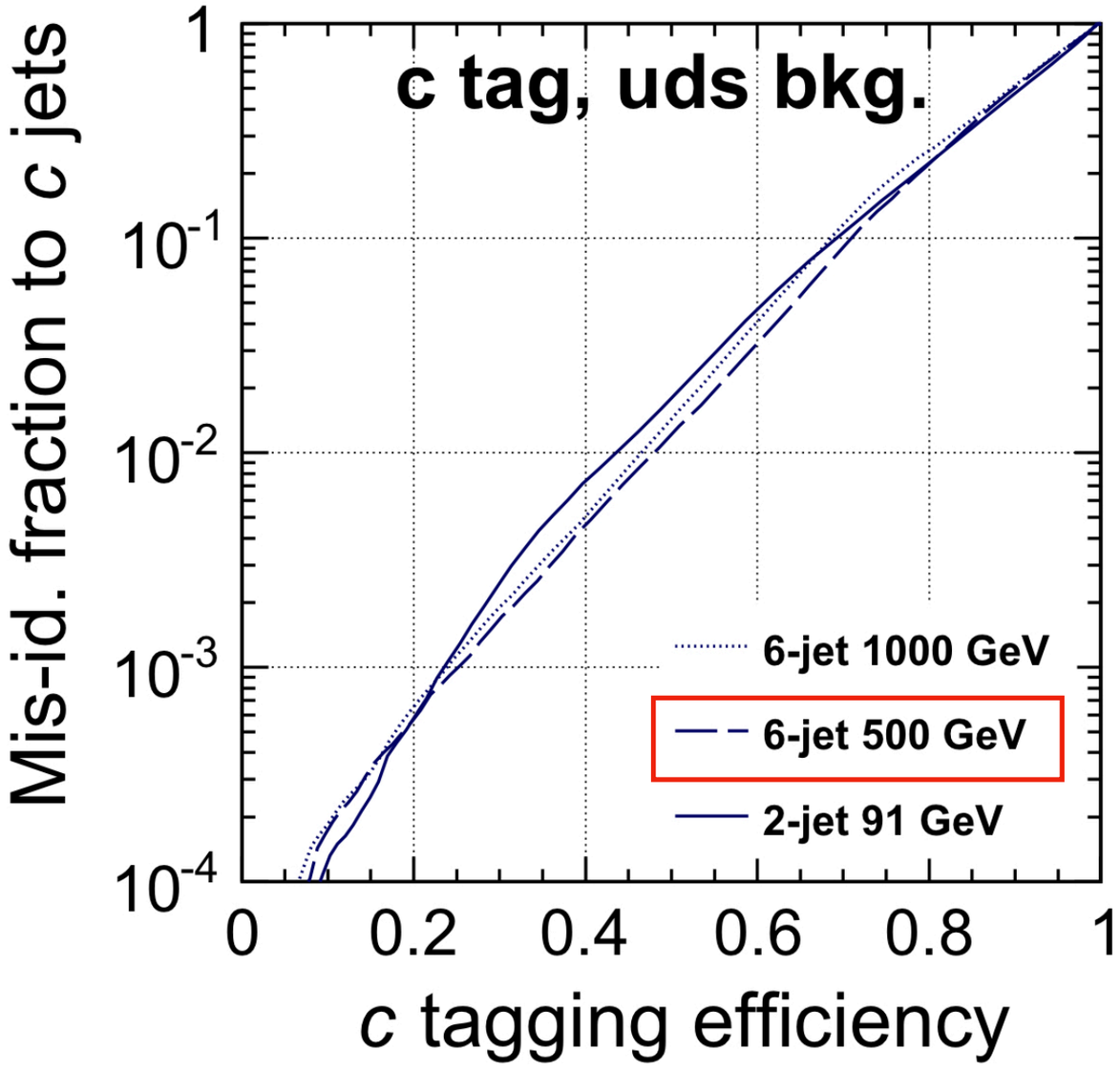


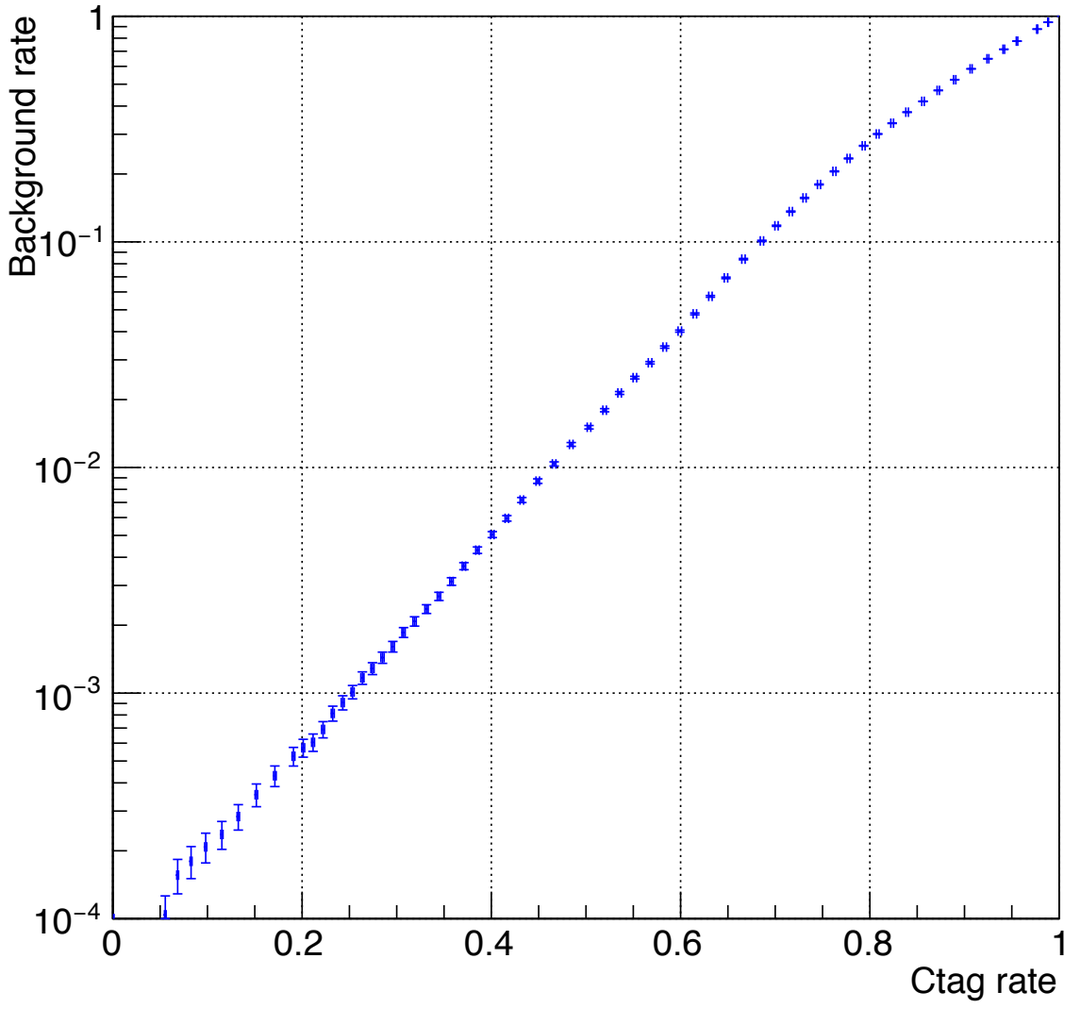
- ILD DBD sample (no IP smearing, but emulated in LCFIPlus) used. ~20k for each sample.
- Beam spot constraint (639nm,5.7nm,91.3um) as written in [1].
- v00-07 used. MVA param. Maxdepth=6

[1]: LCFIPlus: A framework for jet analysis in linear collider studies, Taikan Suehara, Tomohiko Tanabe

Consistent result.

Comparison with previous result [1]





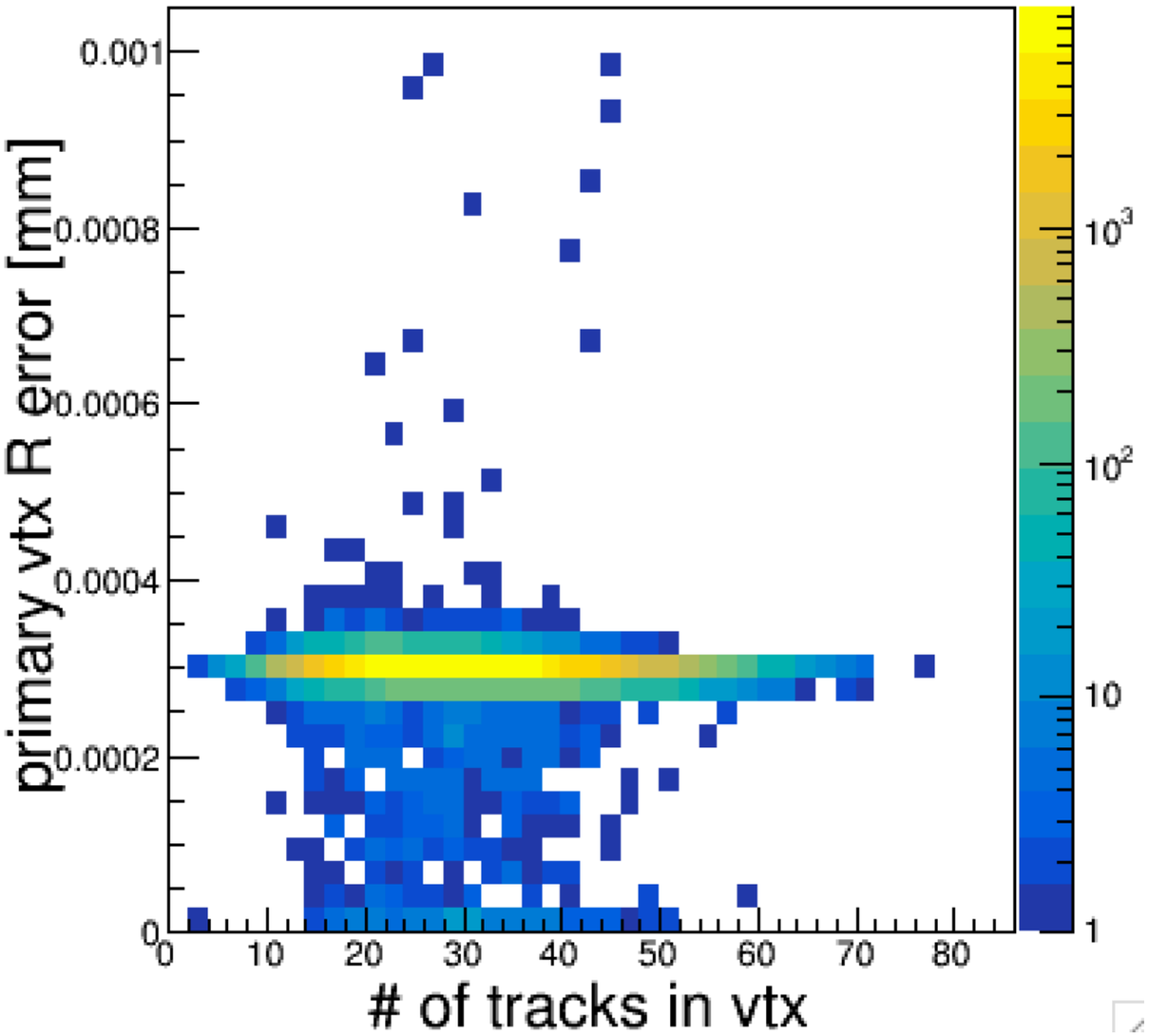
[1]: LCFIPlus: A framework for jet analysis in linear collider studies, Taikan Suehara, Tomohiko Tanabe

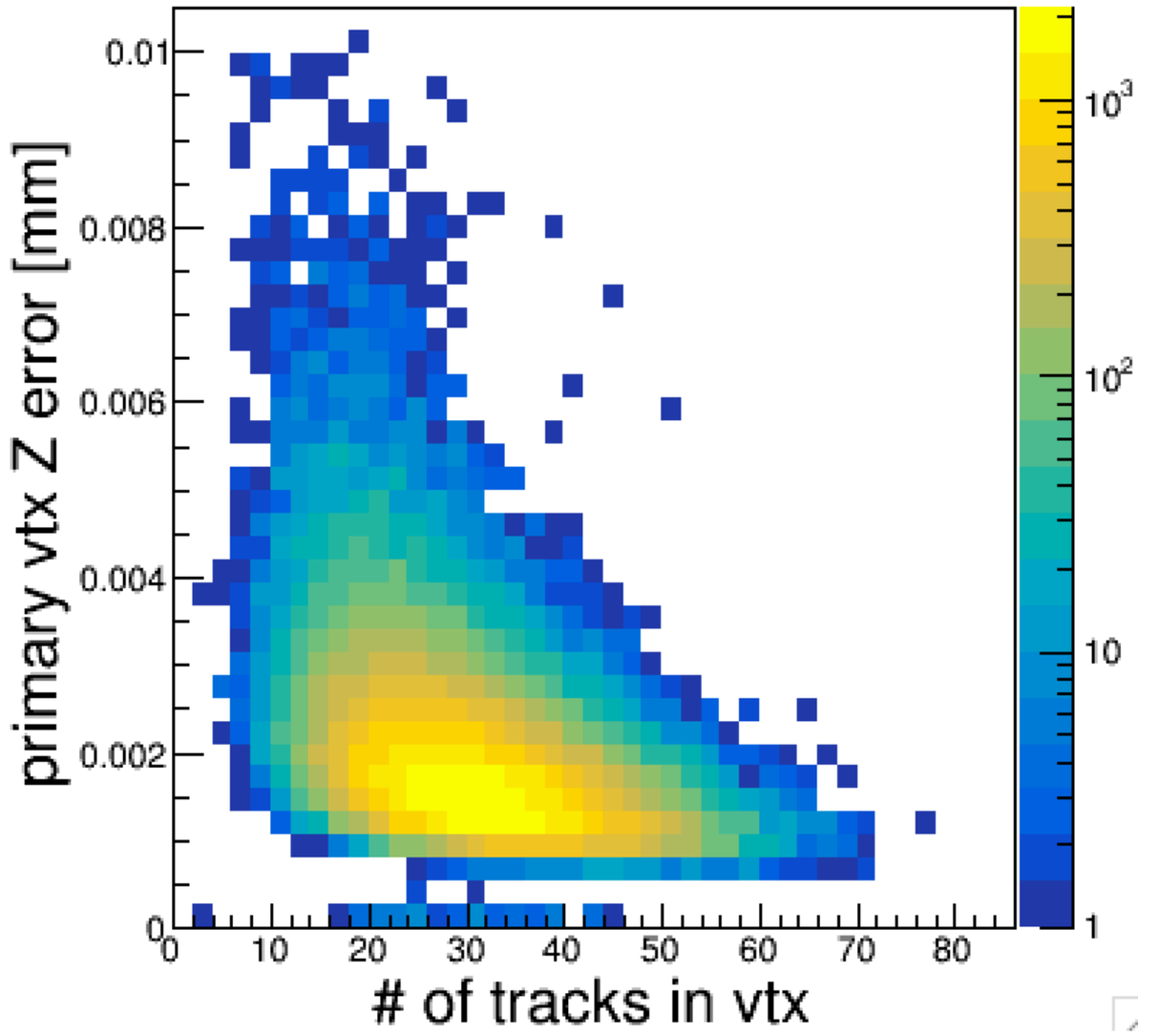
- ILD DBD sample (no IP smearing, but emulated in LCFIPlus) used. ~20k for each sample.
- Beam spot constraint (639nm,5.7nm,91.3um) as written in [1].
- v00-07 used. MVA param. Maxdepth=6

Consistent result.

ILD(l5) Vertexing performance
Primary vertex position resolution vs number of tracks
(w/o beam bkg)

6b,√s=500GeV
ILD(l5) sample used



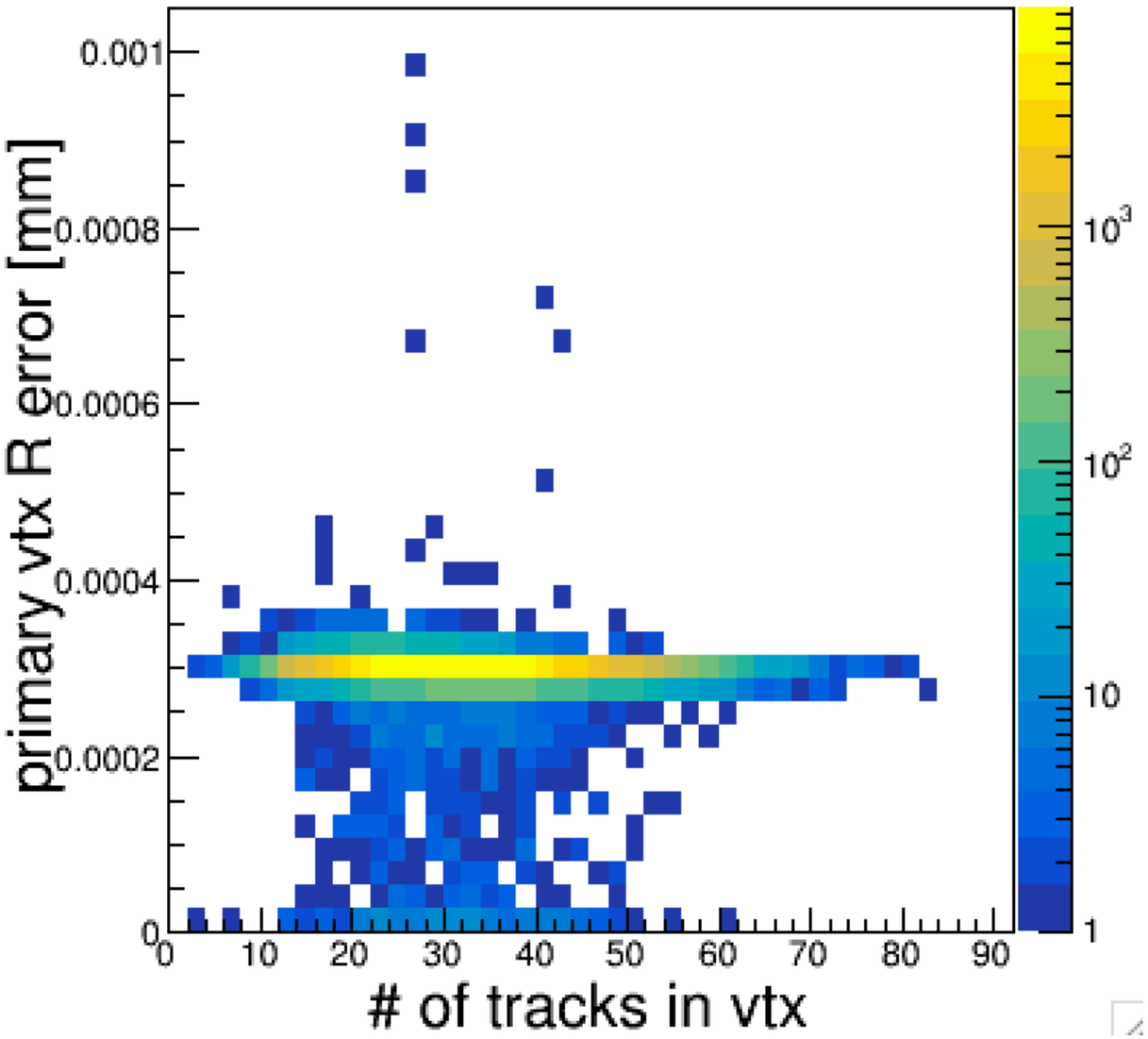


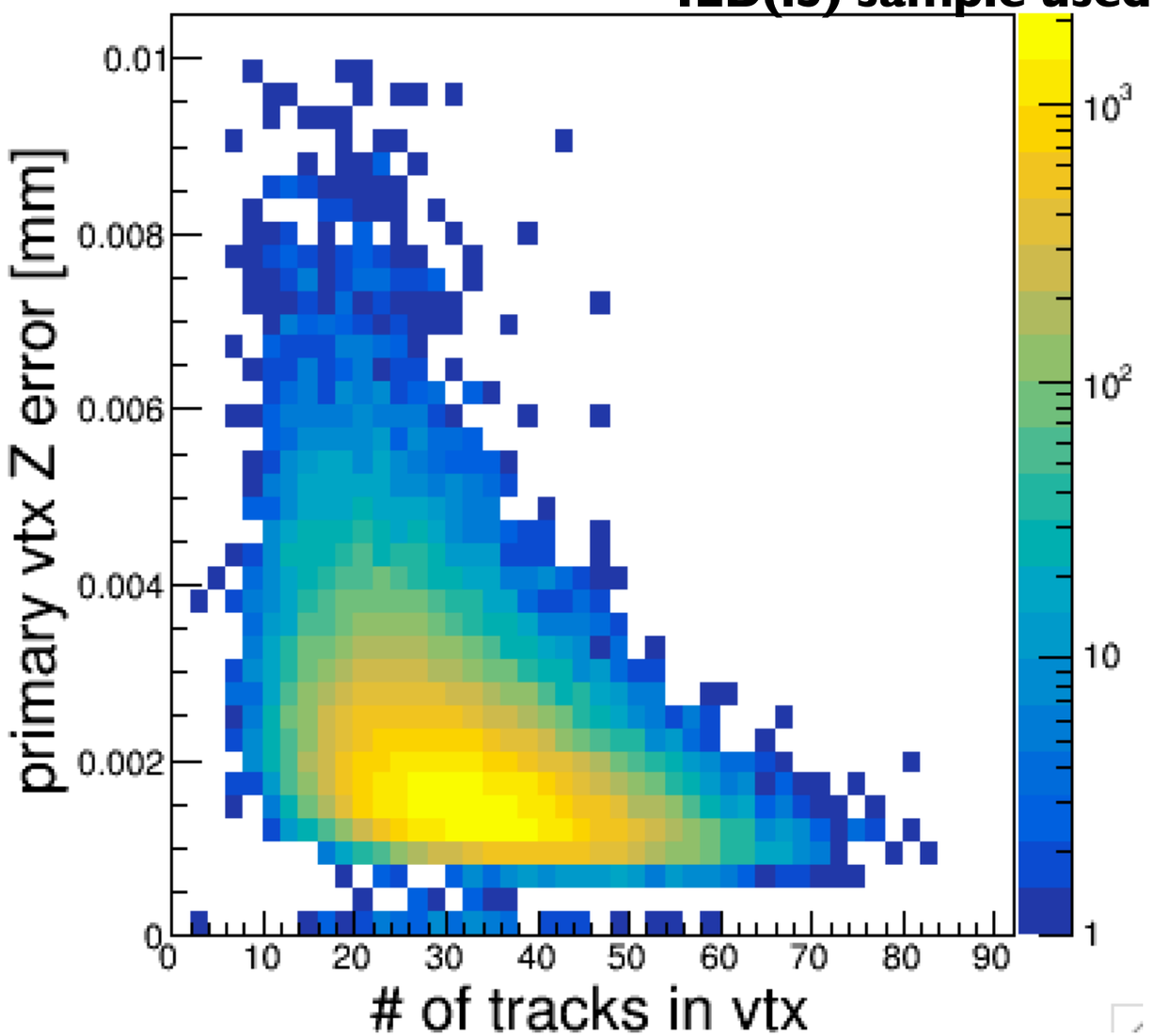
w/ beam constraint

beam size x ~ 300nm
beam size y ~ 2nm
beam size z ~ 200um

ILD(l5) Vertexing performance
Primary vertex position resolution vs number of tracks (w/beam bkg)

6b,√s=500GeV
ILD(l5) sample used



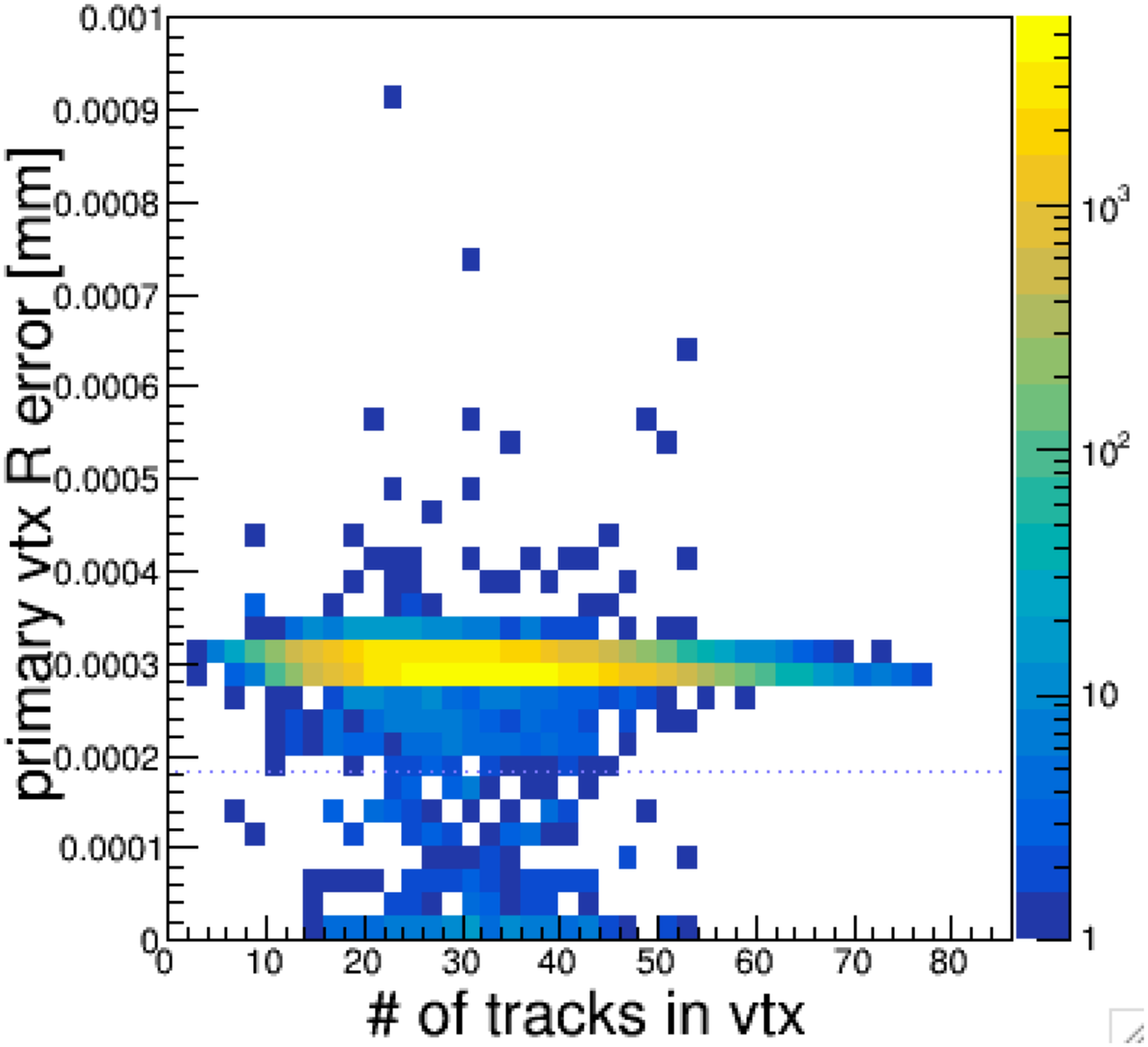


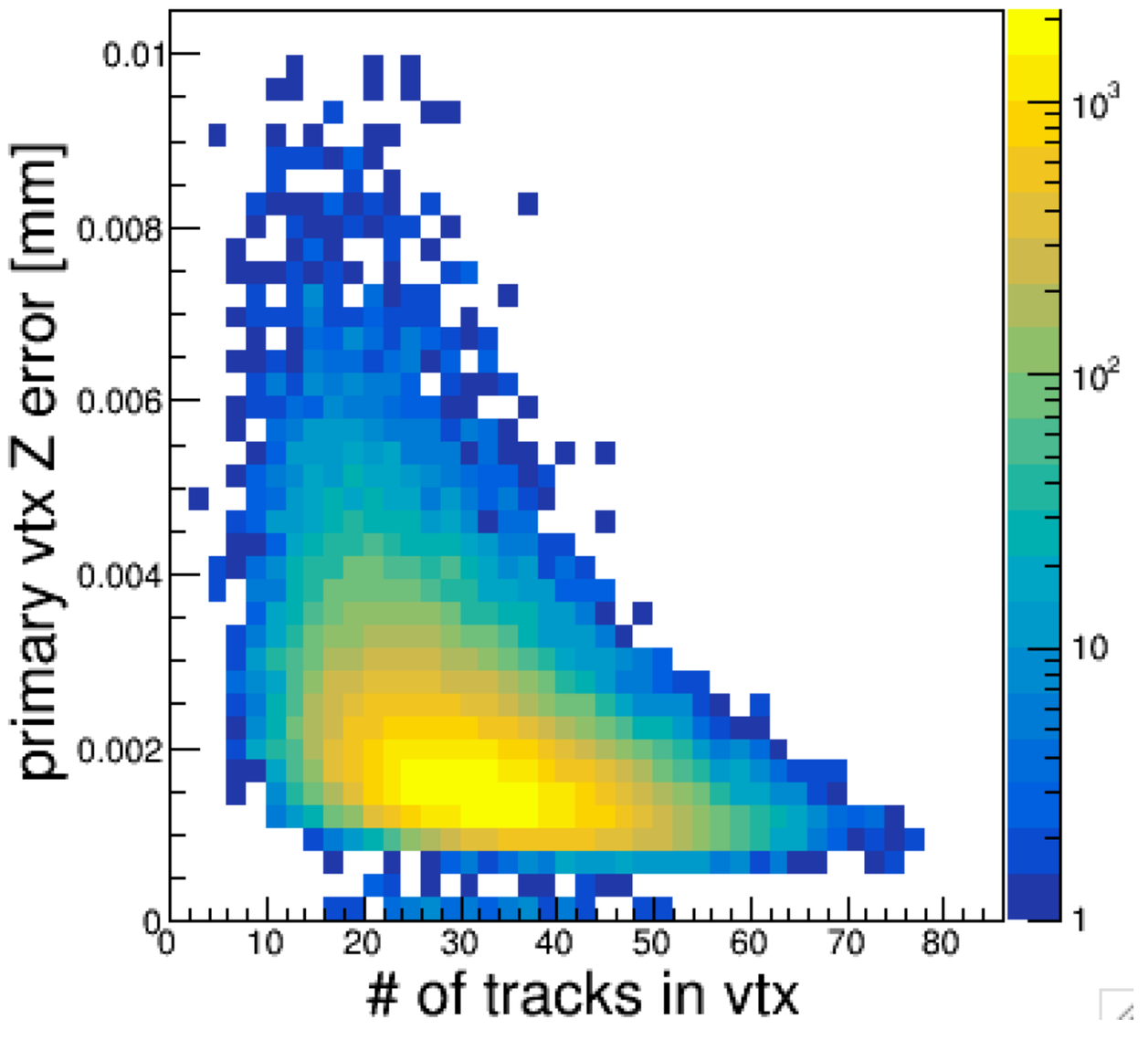
w/ beam constraint

beam size x ~ 300nm
beam size y ~ 2nm
beam size z ~ 200um

ILD(s5) Vertexing performance
Primary vertex position resolution vs number of tracks (w/beam bkg)

6b,√s=500GeV
ILD(l5) sample used



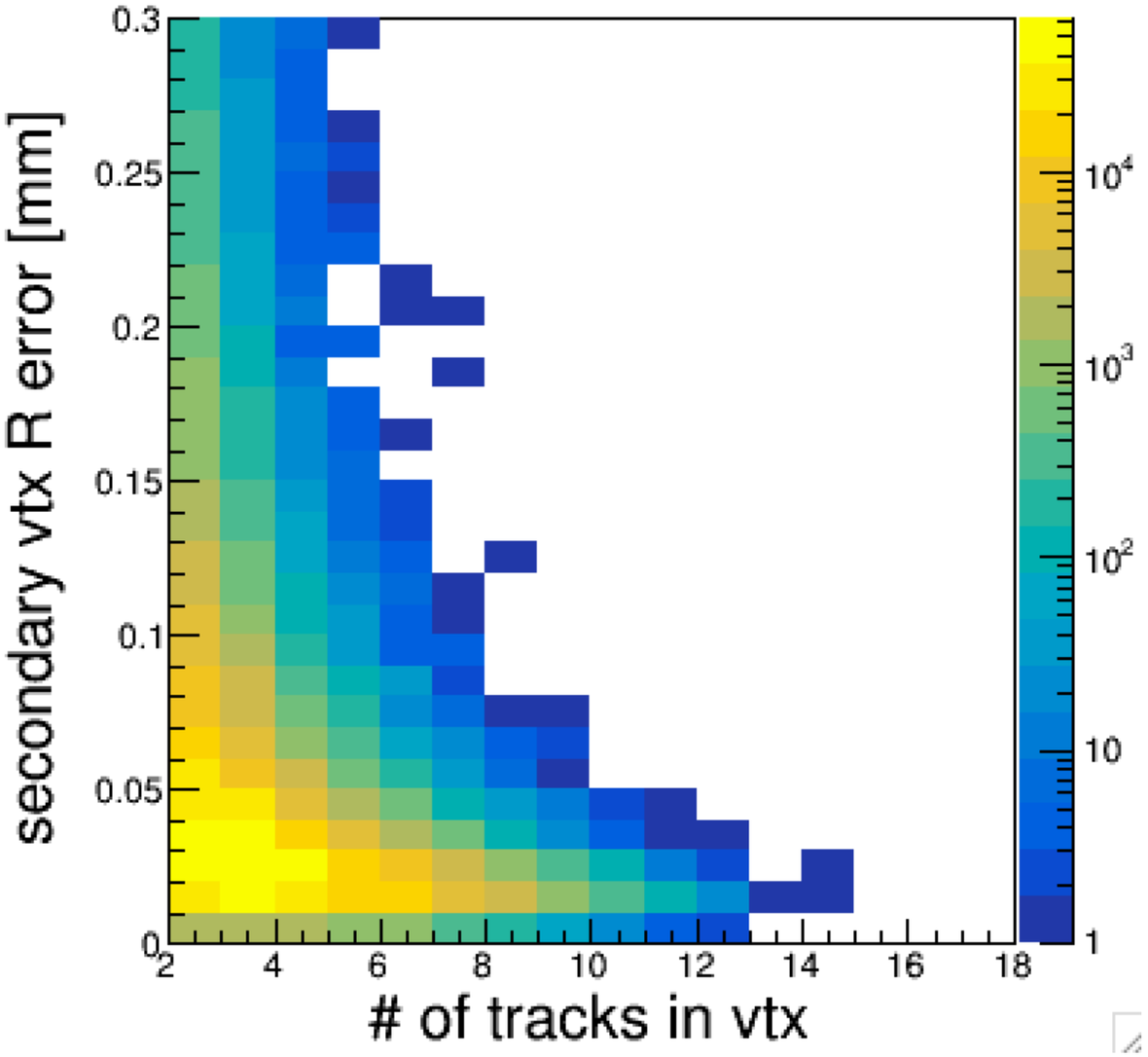


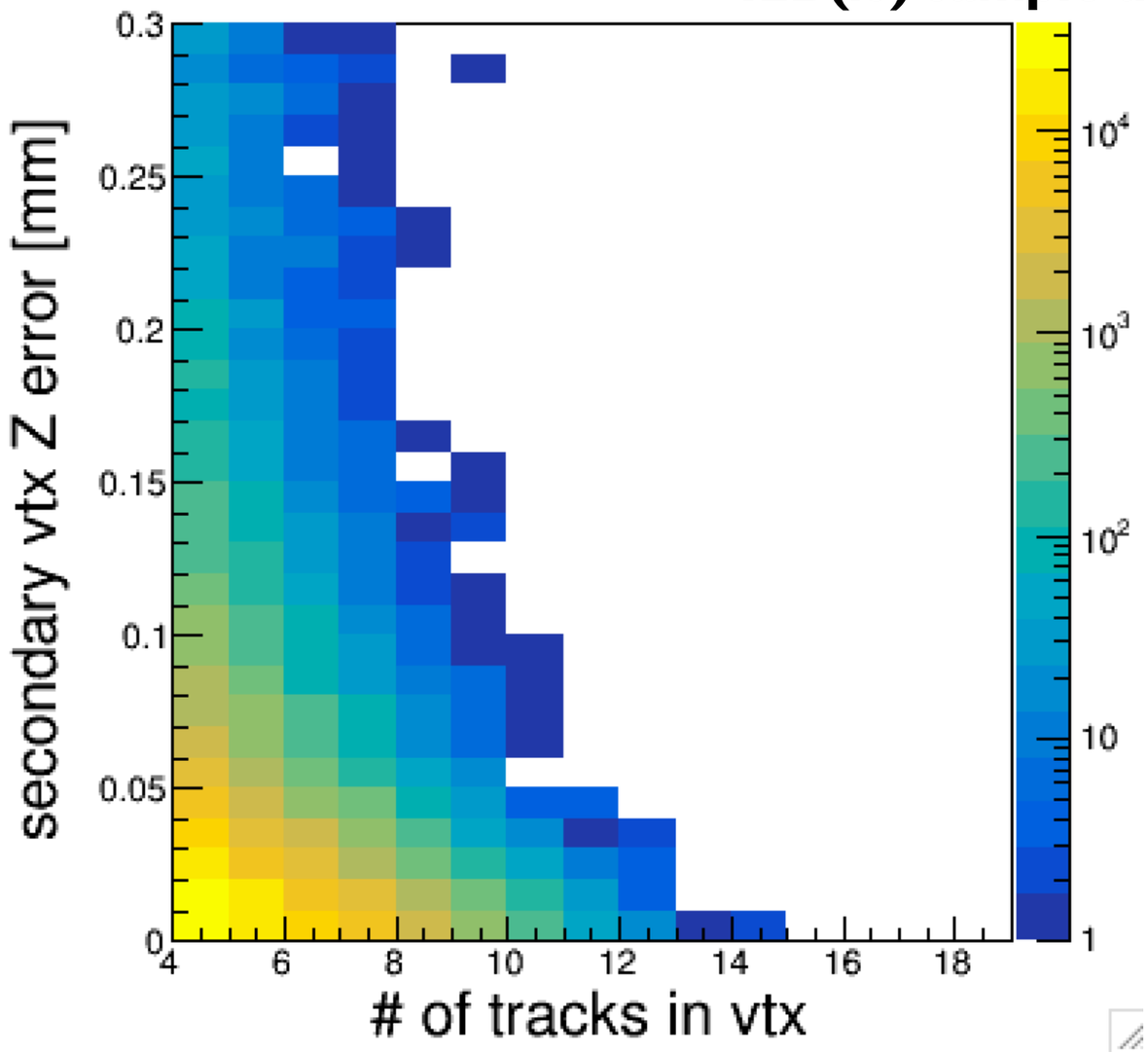
w/ beam constraint

beam size x ~ 300nm
beam size y ~ 2nm
beam size z ~ 200um

ILD(s5) Vertexing performance
Primary vertex position resolution vs number of tracks
(w/o beam bkg)

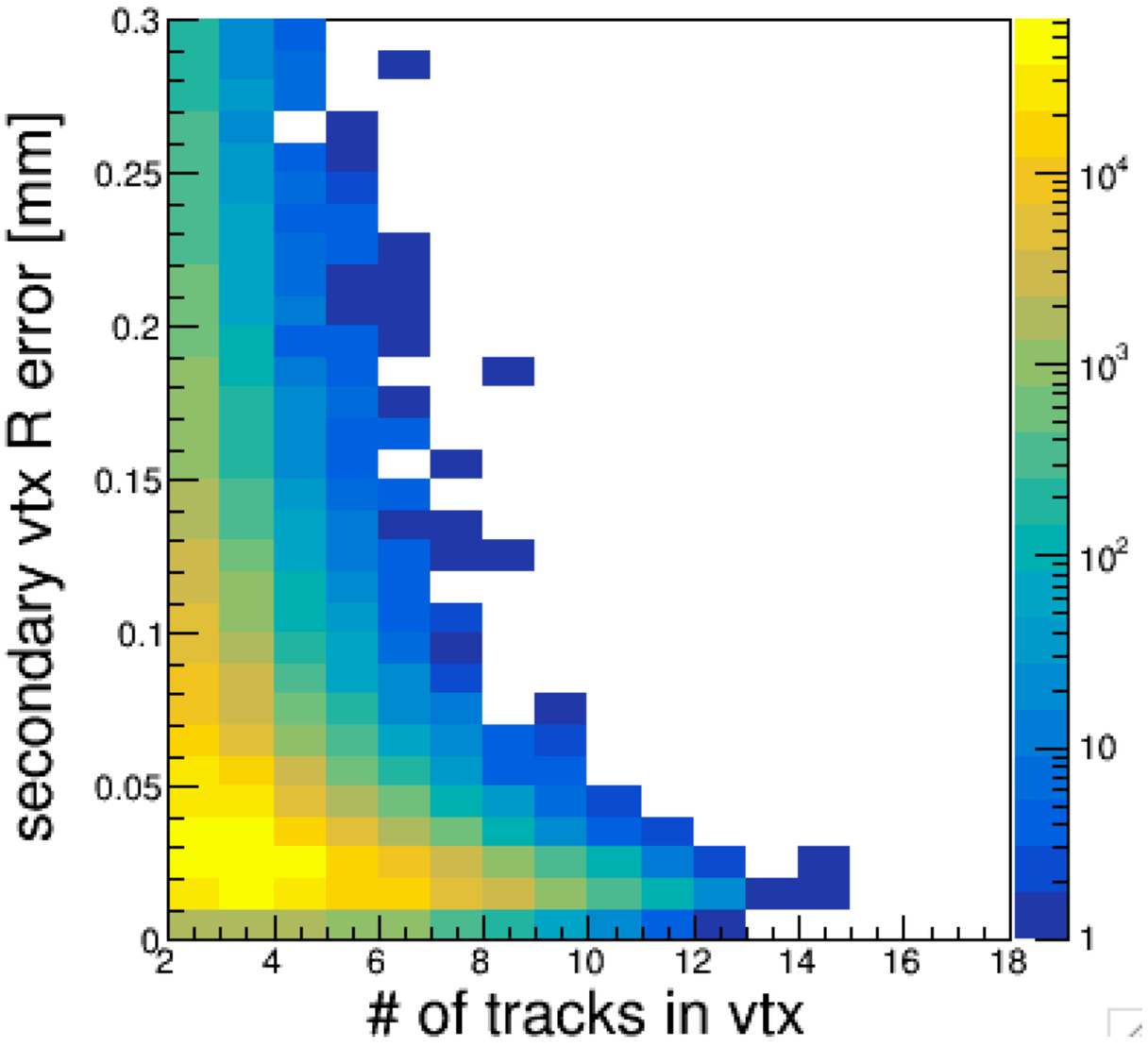
6b,√s=500GeV
ILD(l5) sample used

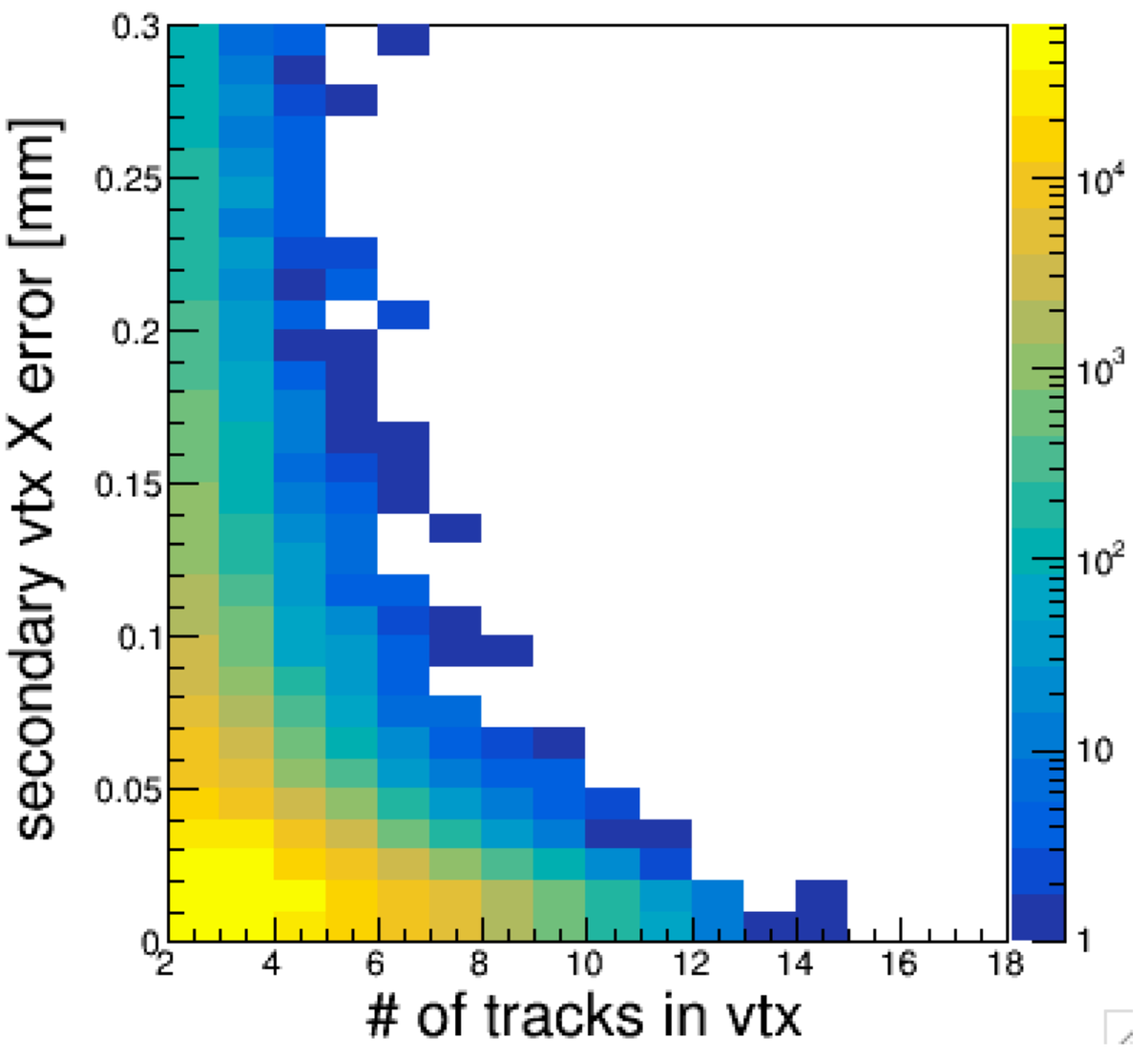




ILD(l5) Vertexing performance
Secondary vertex position resolution vs number of tracks (w/beam bkg)

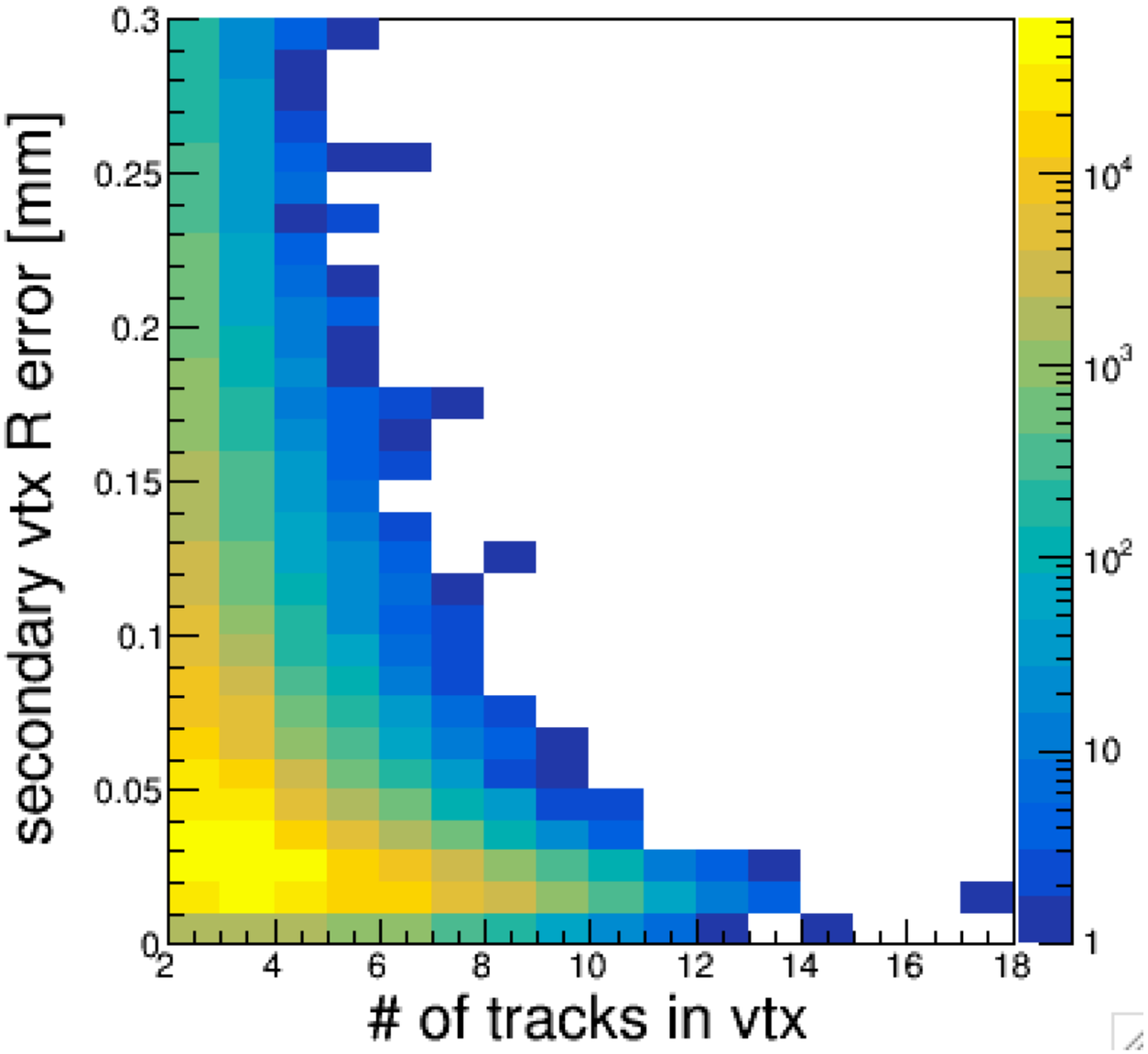
6b,√s=500GeV
ILD(l5) sample used

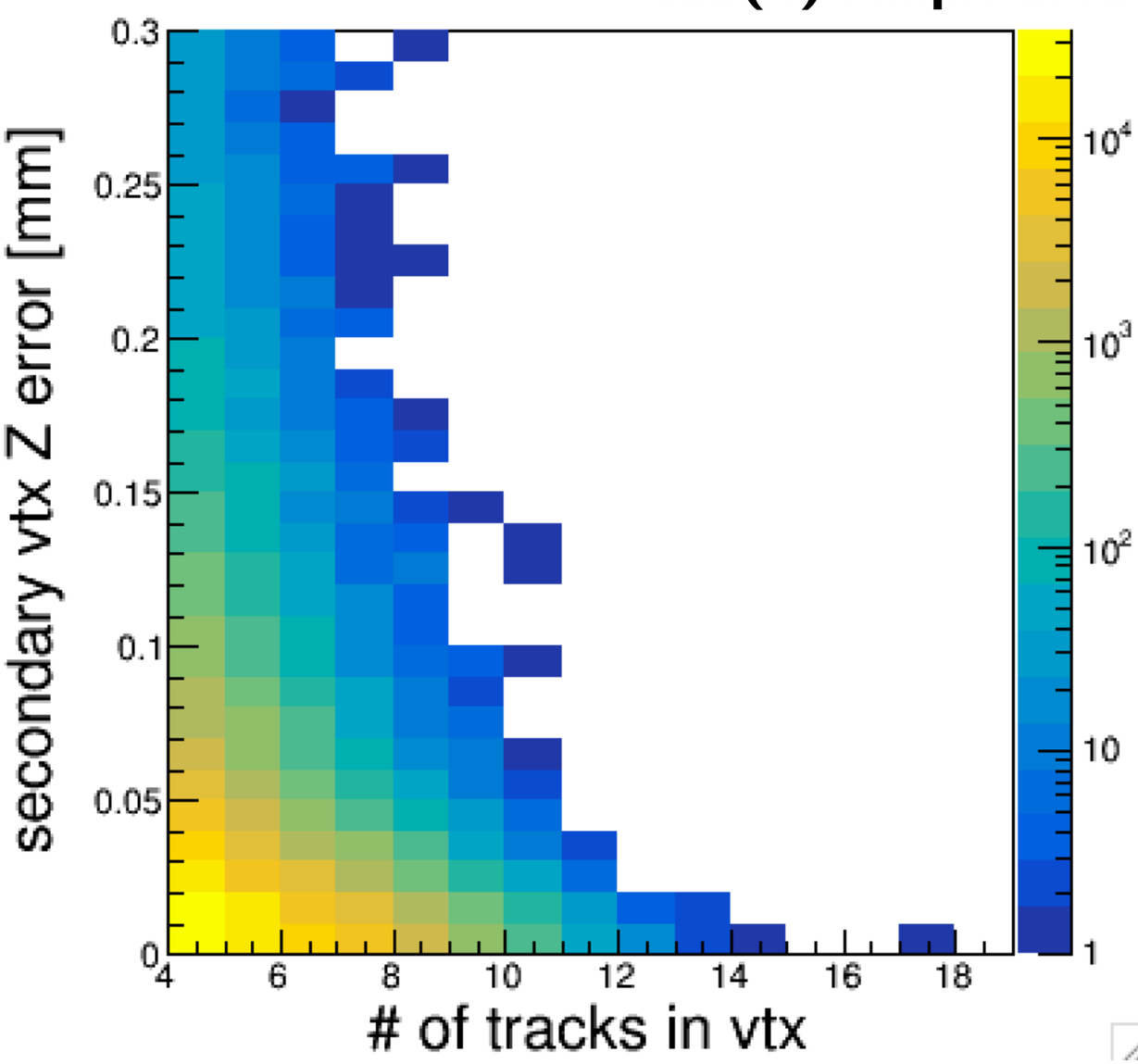




ILD(s5) Vertexing performance
Secondary vertex position resolution vs number of tracks (w/beam bkg)

6b,√s=500GeV
ILD(l5) sample used

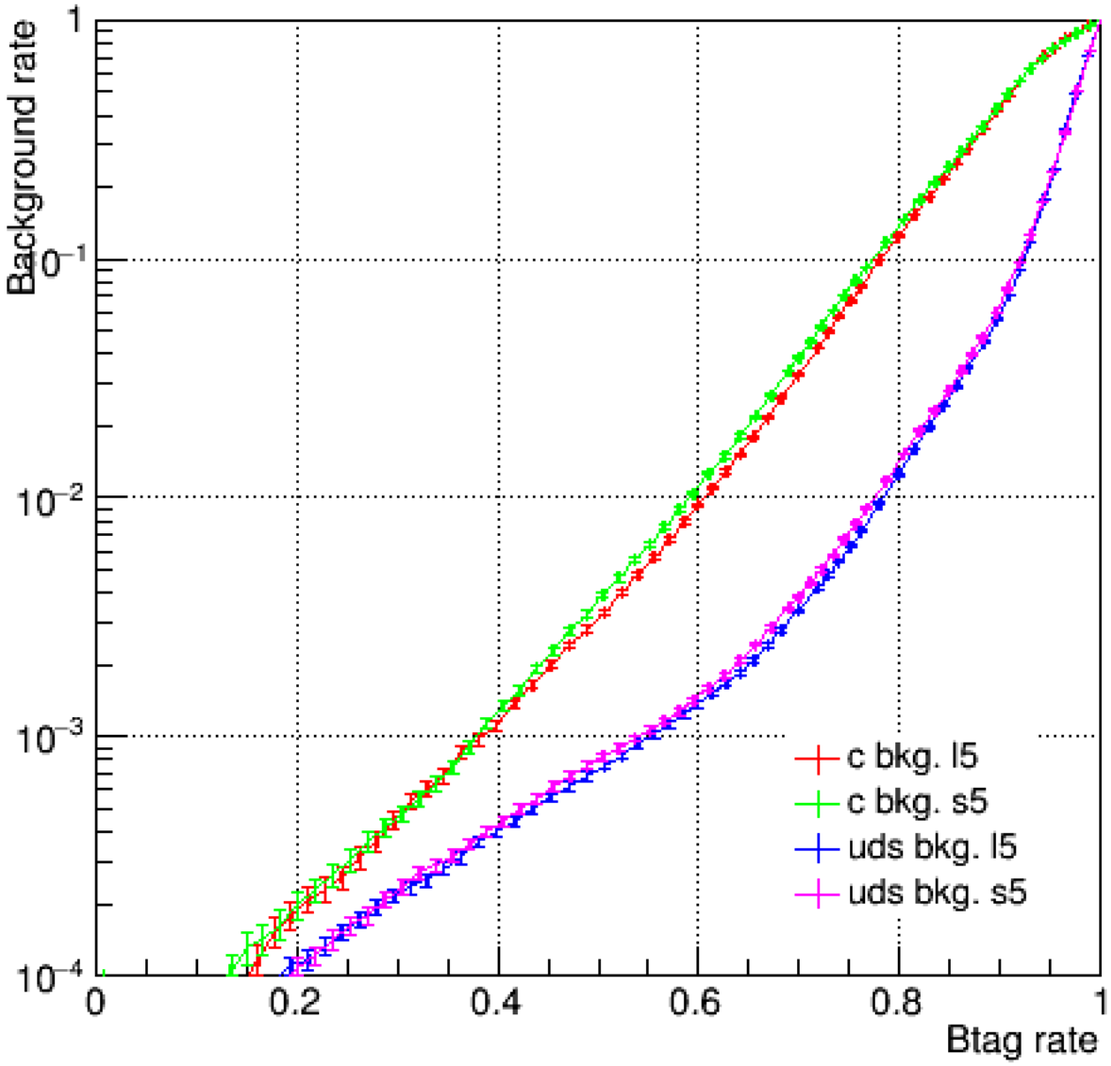


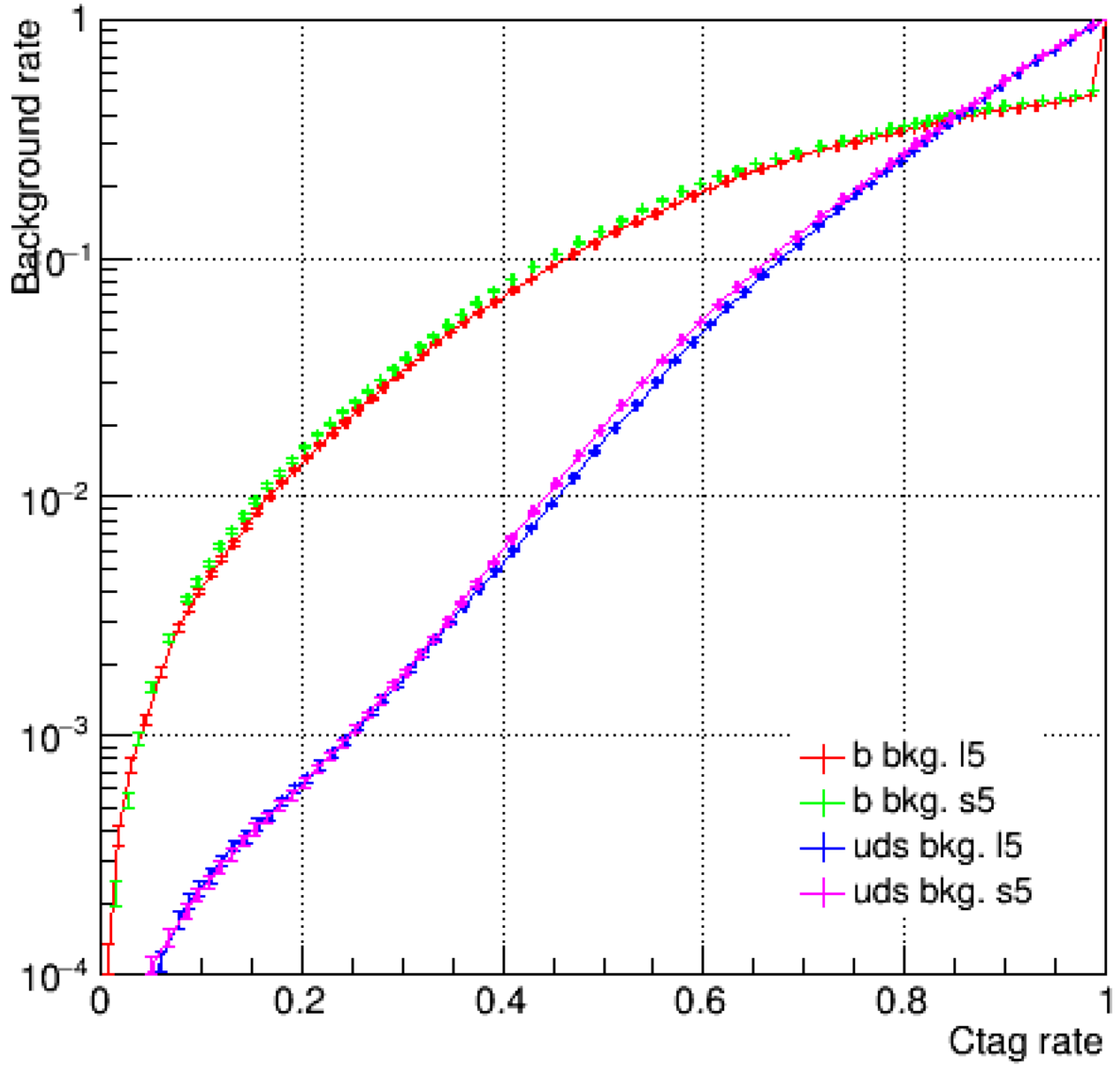


Flavour Tagging performance

w/o beam bkg sample (test and training)

6b, 6c, 6q,√s=500GeV
ILD(l5) sample used

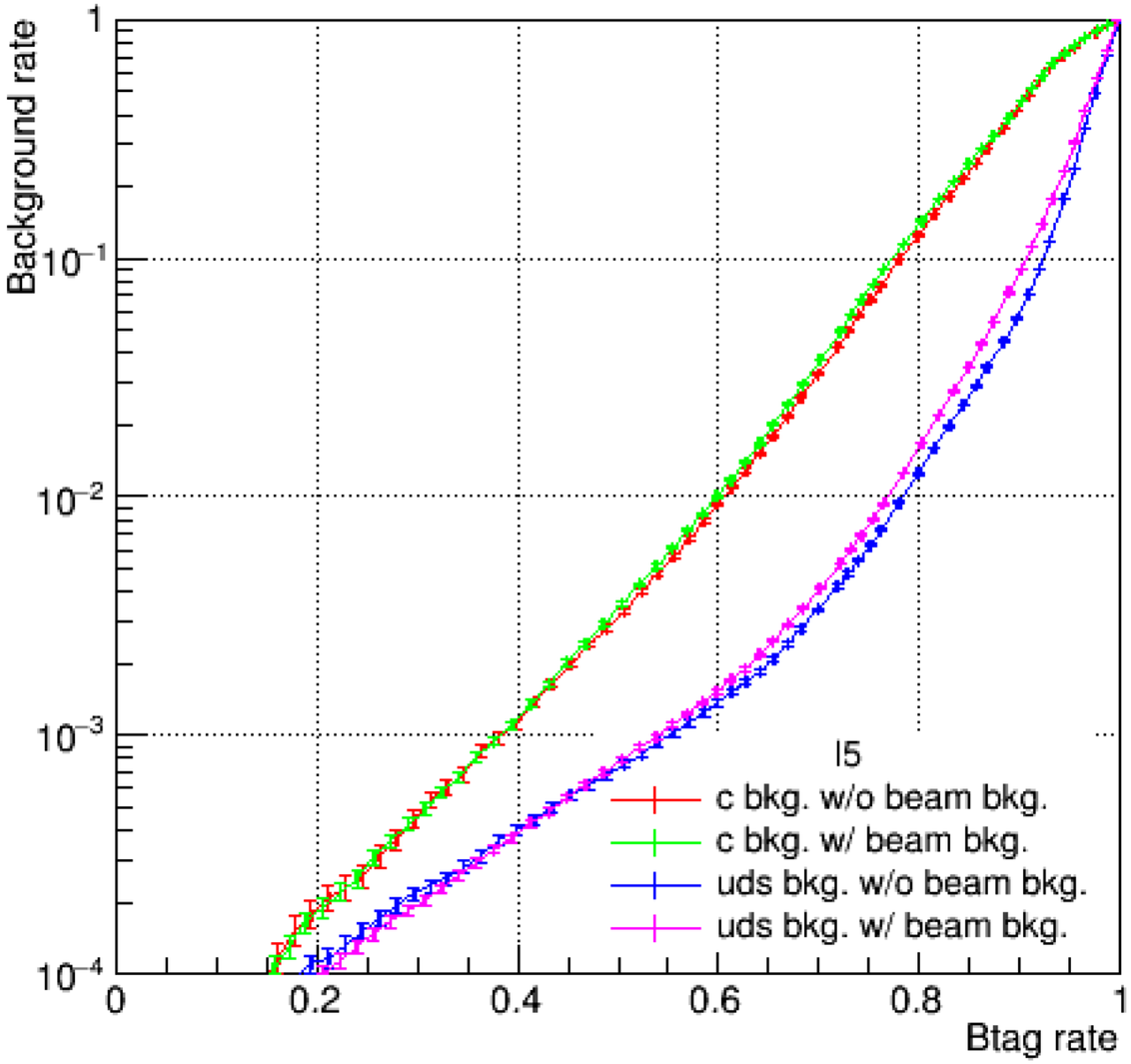


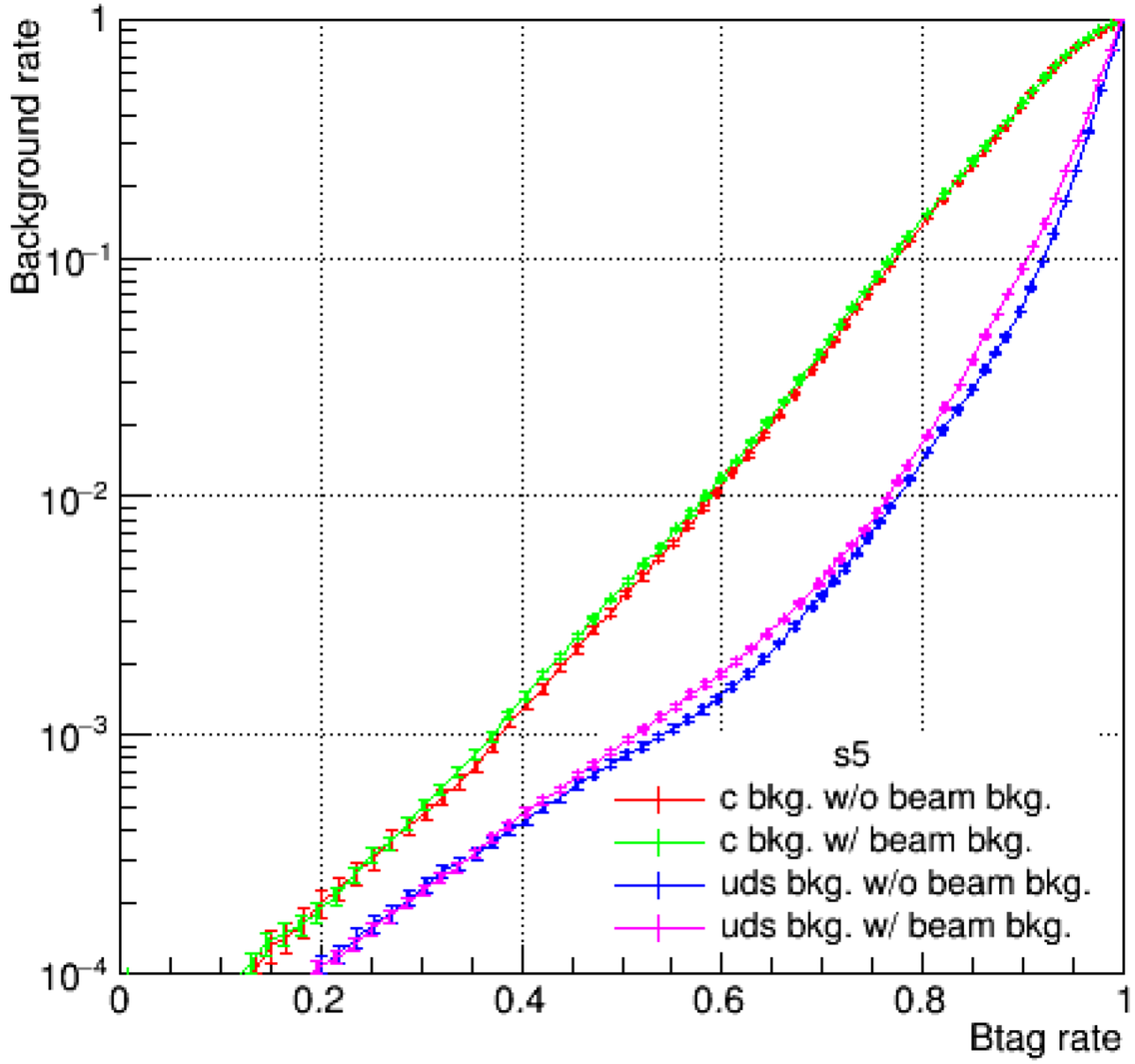


Flavour Tagging Performance (b tagging)

w/beam bkg sample as test sample
w/o beam bkg sample for training

6b, 6c, 6q,√s=500GeV
ILD(l5) sample used

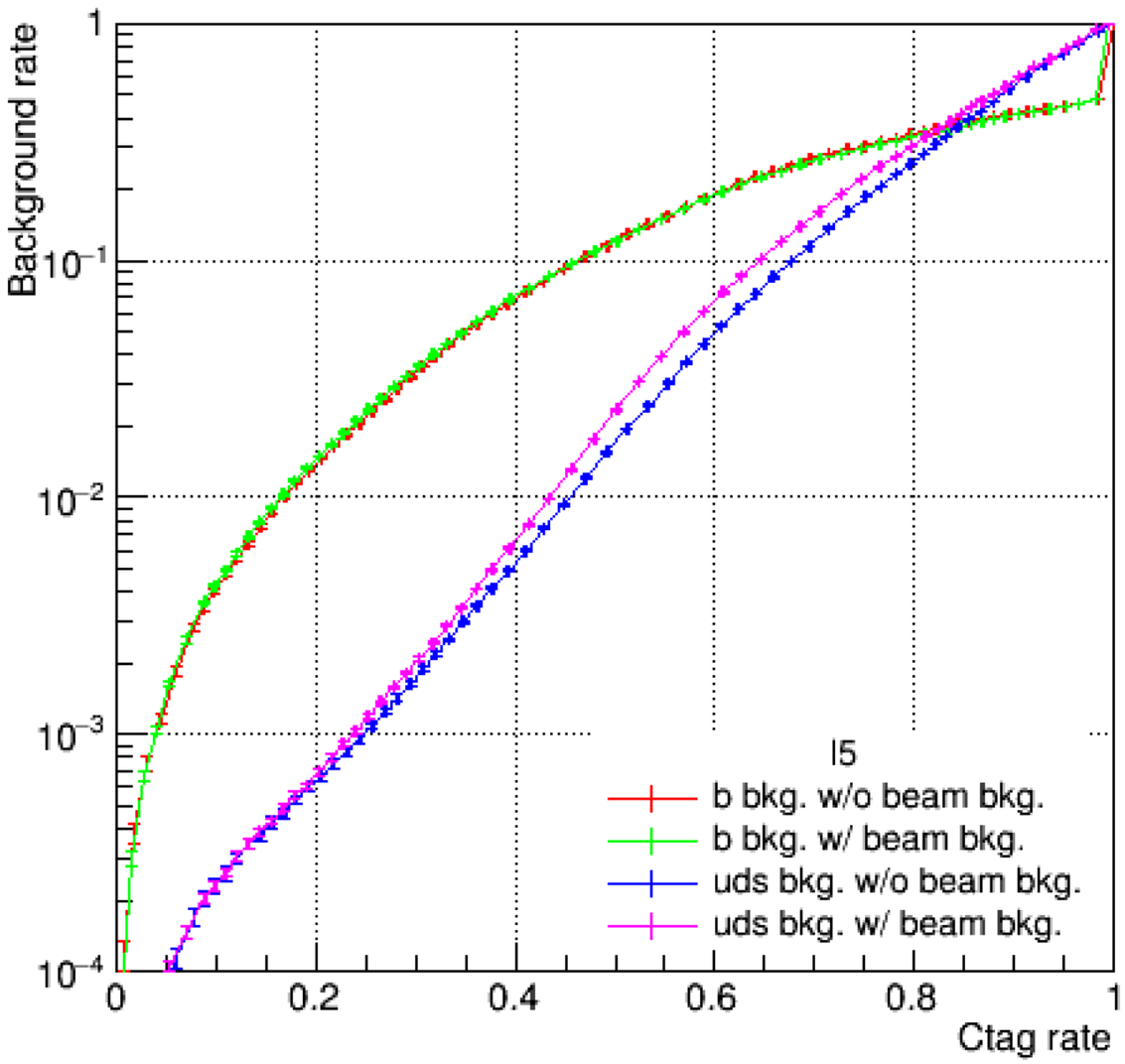


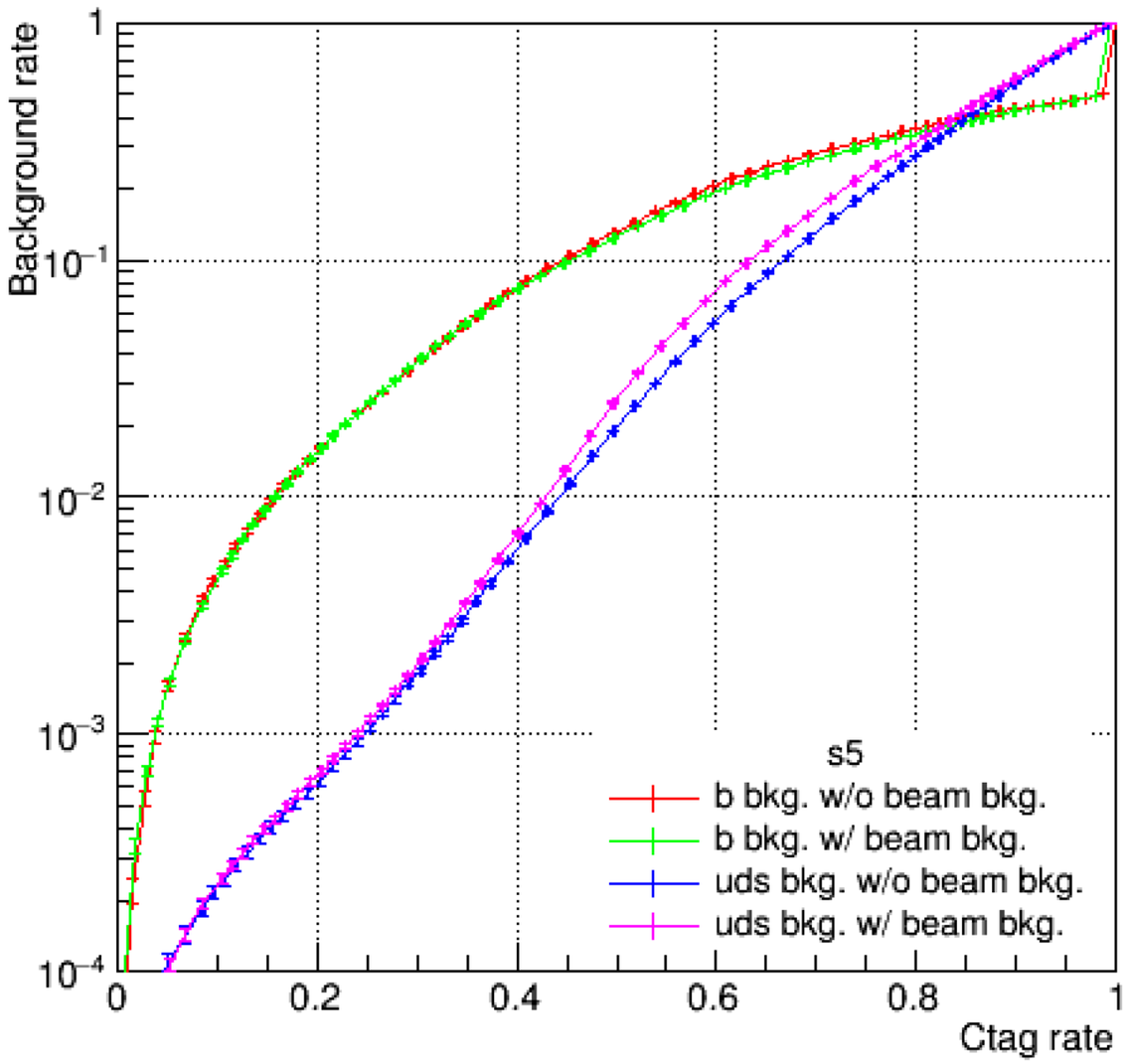


Flavor Tagging Performance (c tagging)

w/beam bkg sample as test sample
w/o beam bkg sample for training

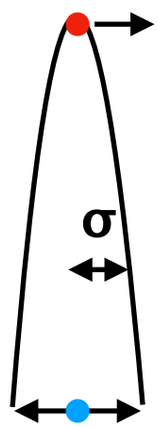
6b, 6c, 6q,√s=500GeV
ILD(l5) sample used





IP smearing

- **In primary vertex finding, beam constraint is useful to reject non-primary tracks. Use an additional point (=constraint point) in primary vertex fitting.**



**constraint point at (0,0,0)
with a certain error corresponding to
beam spot sizes (x,y,z).**

- **In real experiment, IP is distributed in luminous region.**
  - In DBD samples, IP is always fixed at (0,0,0)
  - Instead, we smear the centre point of the constraint with sigmas comparable to beam spot sizes in LCFIPlus.
  - In recent production, IP is smeared in simulation step. No need "constraint point" smearing in LCFIPlus.