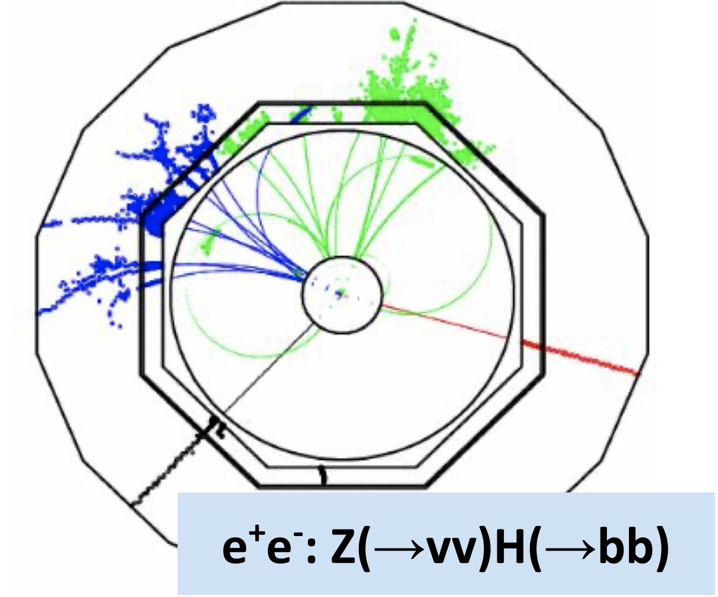
Jet Flavour Tagging at the FCC-ee

Franco Bedeschi, Loukas Gouskos,
Michele Selvaggi

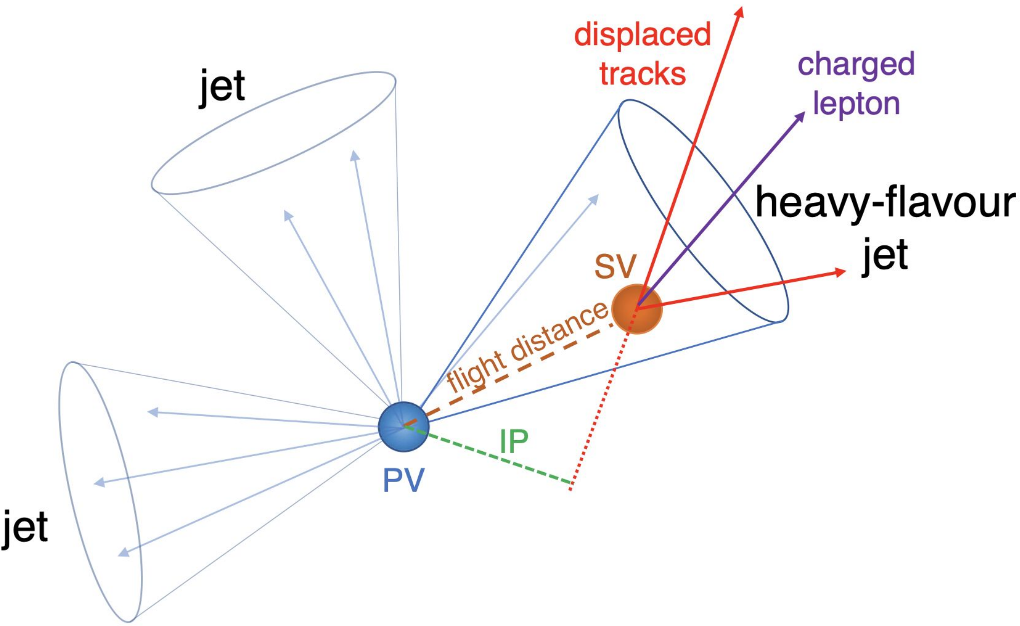
June 29, 2021

Introduction



- **Goals:**
  - Develop a versatile jet flavour tagger for the FCC-ee:
    - Identify with high purity light / strange / charm / beauty jets
      - multi-class classifier
  - Understand the detector requirements/optimize design
    - vertexing and PID capabilities of the FCCee detector

Basics of flavour tagging (b/c)



Detector constraints:

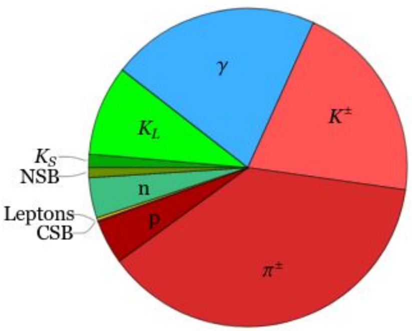
Need power pixel/tracking detectors

- Good spatial resolution
- As little material as possible
- Precise track alignment

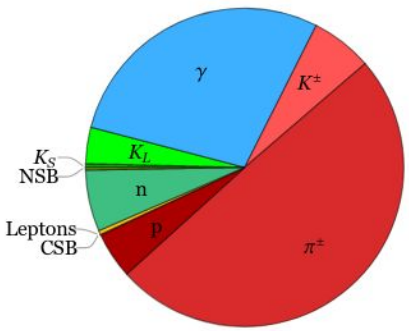
- Large lifetime
  - b (c) lifetime ~ps (~0.1ps)
  - b (c) decay length: ~5 (2-3) mm for ~50 GeV boost
- Displaced vertices/tracks
  - Large impact parameters
  - Tertiary vertices when B hadron decays to C hadron
- Large track multiplicity
  - ~5 (~2) charged tracks/decay
- Non-isolated e/μ
  - ~20 (10)% in B (C) decays

Basics of flavour tagging (strange)

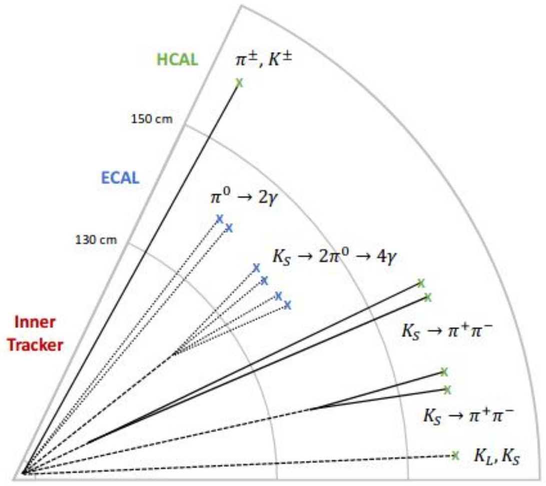
[2003.09517] Momentum weighted fraction:



Strange $p_T = 45\,\mathrm{GeV}$



Down $p_T = 45\,\mathrm{GeV}$



- Large Kaon content
  - Charged Kaon as track:
    - K/pi separation
      - TOF
      - dEdx/dNdx
  - Neutral Kaons:
    - $K_S \to \pi\pi$
      - Displaced 2 track vertex
      - 4 photons
    - K_L
      - TOF vs n ?

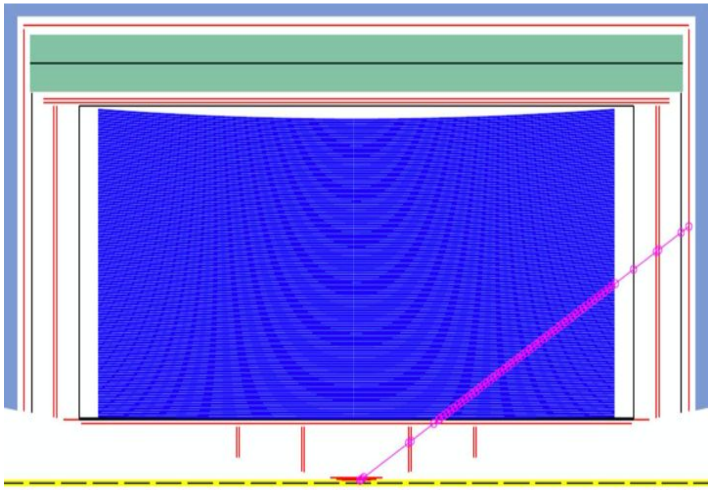
Detector constraints:

Need power pixel/tracking detectors
- good spatial resolution
- timing detectors
- charged energy loss (gas/silicon)

Simulation

- MC Samples:
  - MG5+Pythia8 used to generate:
    - ee→ ZH→vvXX events (X: g, ud, s, c, b)

IDEA

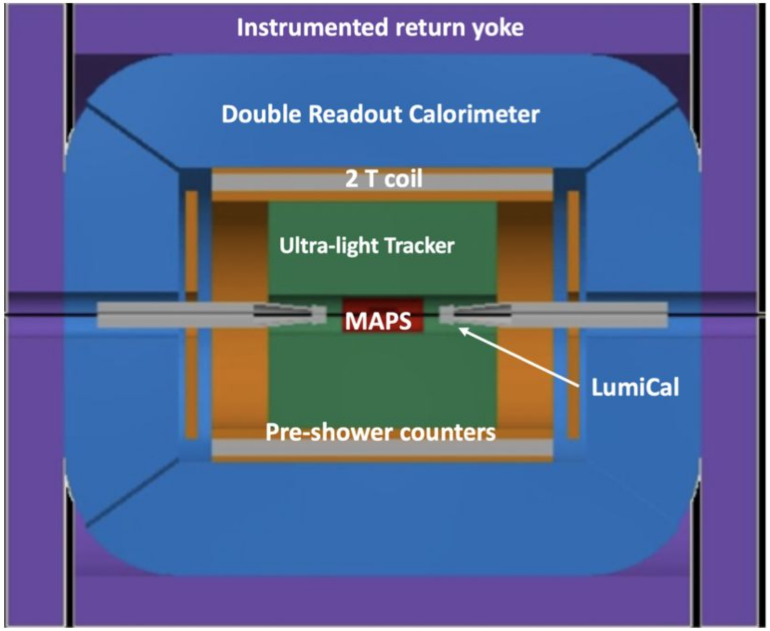


- Detector response based on Delphes:
  - Including FastTrackCovariance
  - Computes:
    - full track covariance matrix (5x5)
      - Including MS
    - smeared track using the off-diagonal terms
    - path length and dN/dx for various gas mixes
    - Allows fast turn-around when trying different detector options
- Jets clusters with the generalized-kT algorithm using R=1.5
  - Similar to the anti-kT algorithm [IRC safe]

FCCee detector

- Ideal for flavour identification [hence: measure Higgs couplings]
  - Impact parameter resolution
    - Low material budget tracker (minimise multiple scattering)
    - Small beam-pipe 1.5 cm -- investigating 1 cm
  - PID capabilities
    - dEdx (Si tracker) -- Cluster counting (Drift)
    - Time of flight -- timing layer

IDEA



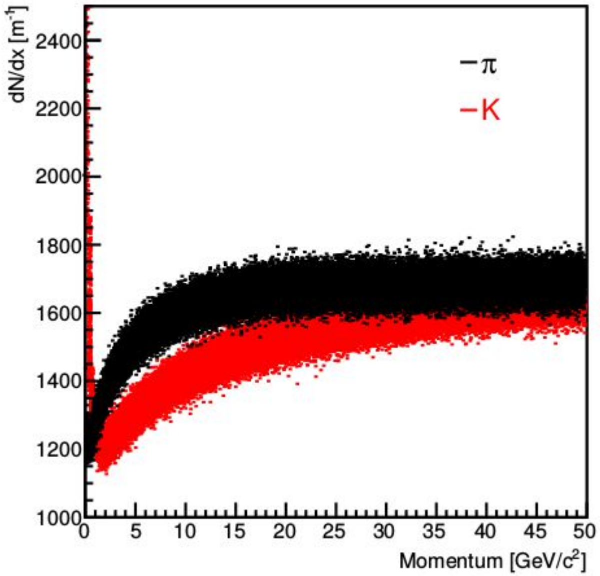
Cluster counting dN/dx

- Count number of **primary ionisation** clusters along track path
- Avoids large landau flukes (**poisson distributed**)
- Requires high granularity
- Module added in Delphes

```
###################
# Cluster Counting
###################

module ClusterCounting ClusterCounting {

  add InputArray TrackSmearing/tracks
  set OutputArray tracks

  set Bz $B

  ## check that these are consistent with DCHCANI/DCHNANO parameters in TrackCovariance module
  set Rmin $DCHRMIN
  set Rmax $DCHRMAX
  set Zmin $DCHZMIN
  set Zmax $DCHZMAX

  # gas mix option:
  # 0:  Helium 90% - Isobutane 10%
  # 1:  Helium 100%
  # 2:  Argon 50% - Ethane 50%
  # 3:  Argon 100%

  set GasOption 0

}
```

IDEA detector:

90% He / 10 % Isobutane



Time-of-flight

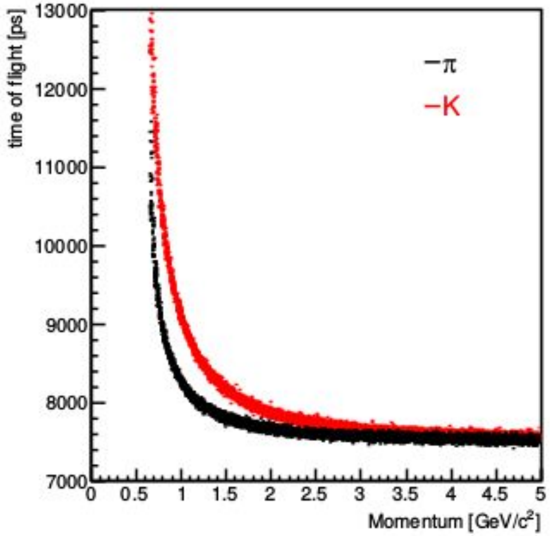
- Allows for good K/pi separation at low momenta:

$$t_{\text{flight}} \equiv t_{\text{F}} - t_{\text{V}} = \frac{L}{\beta} = \frac{L\sqrt{p^2 + m^2}}{p}$$

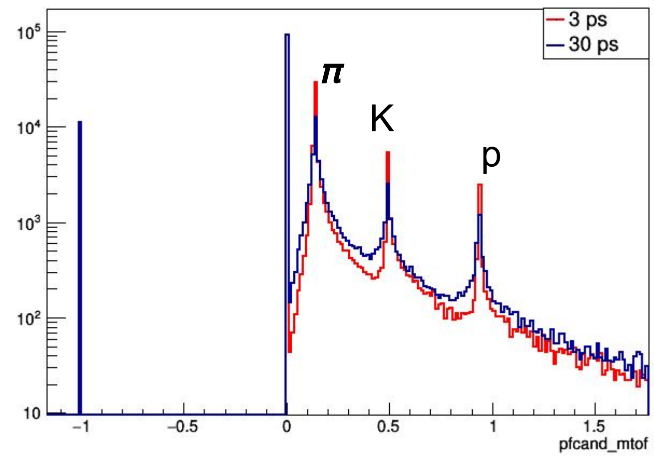
- Need to make assumption on vertex time (crucial for highly displaced K_S) :

```
########################################
#   Time Of Flight Measurement
########################################

module TimeOfFlight TimeOfFlight {
  set TrackInputArray TimeSmearing/tracks
  set VertexInputArray TruthVertexFinder/vertices

  set OutputArray tracks

  # 0: assume vertex time tV from MC Truth (ideal case)
  # 1: assume vertex time tV = 0
  # 2: calculate vertex time as vertex TOF, assuming tPV=0

  set VertexTimeMode 2
}
```

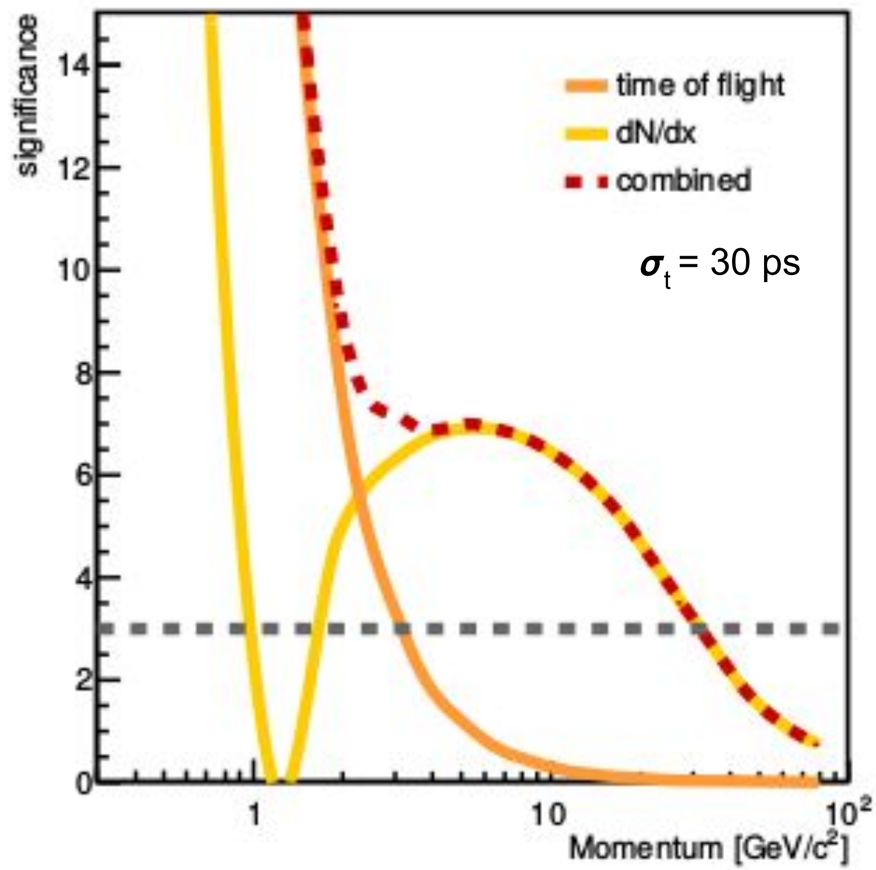
$$t_{\text{V}} = \frac{r_{\text{V}}}{\beta_{\text{V}}}$$



σ_t = 30 ps



Combined PID

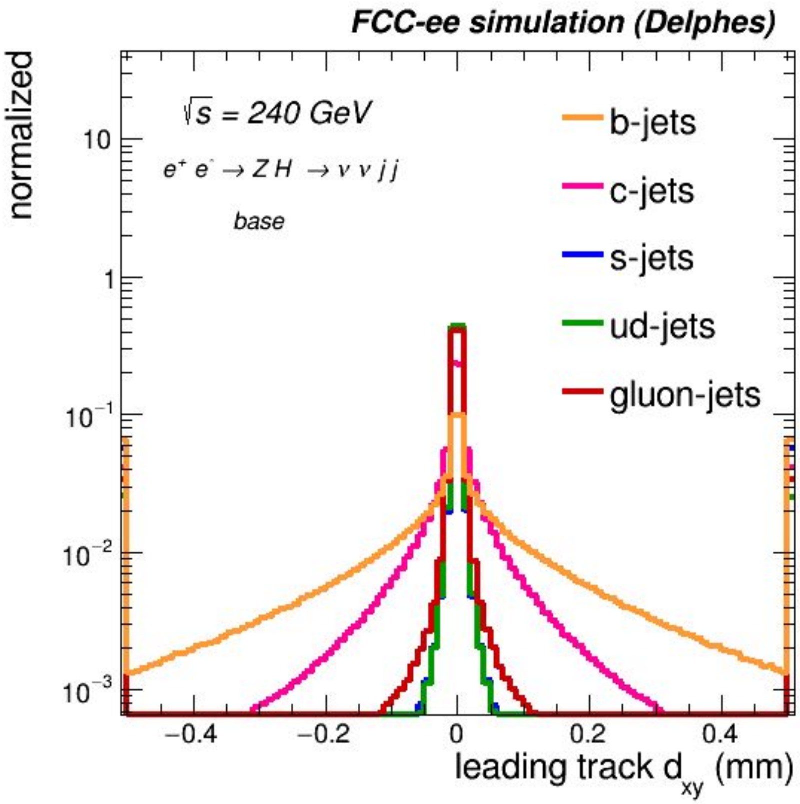


3 std deviation K/pi separation for tracks with p < 30 GeV

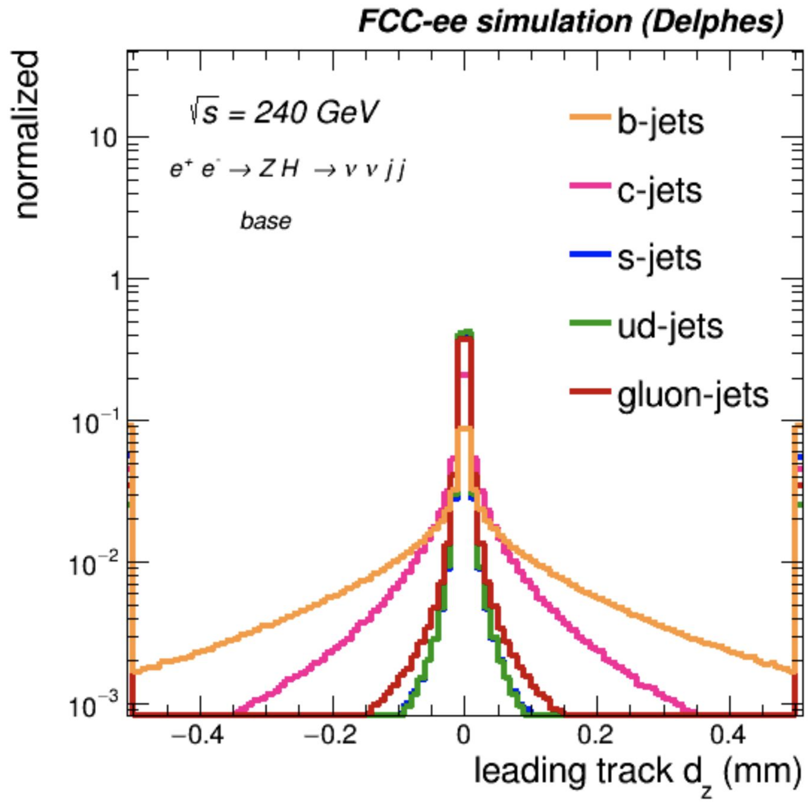
Input variables

- Comparison of input distributions for different jet flavors

Impact parameter (d_0)



Impact parameter (d_z)



- More comparisons:
  https://selvaggi.web.cern.ch/selvaggi/FCC/FCCee/FlavourTagging/

Input Features

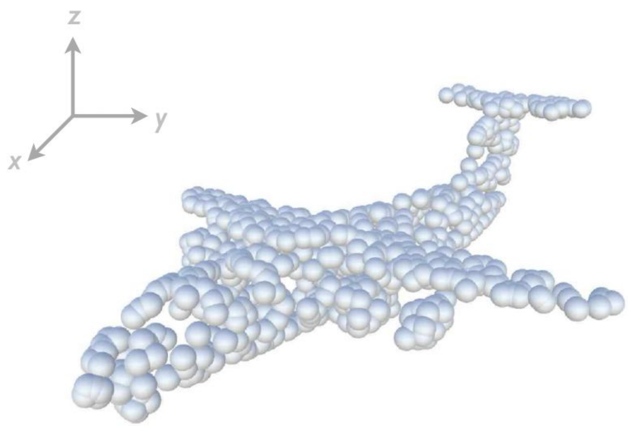
Variable	Description
	Kinematics
$\log E$	logarithm of the particle's energy
θ	absolute value of particle's theta
ϕ	absolute value of particle's azimuhtal angle
$\Delta\theta(\text{jet})$	difference in θ between the particle and the jet axis
$\Delta\phi(\text{jet})$	difference in azimuthal angle between the particle and the jet axis
	Displacement
d_z	longitudinal impact parameter of the track
d_z/σ_{d_z}	significance of the longitudinal impact parameter
d_{xy}	transverse impact parameter of the track
$d_{xy}/\sigma_{d_{xy}}$	significance of the transverse impact parameter
η_{rel}	pseudorapidity of the track relative to the jet axis
$p_{\text{T,rel}}$ ratio	ratio of track momentum perpendicular to the jet axis and track momentum
$p_{\text{par,rel}}$ ratio	ratio of track momentum parallel to the jet axis and track momentum
d_{3D}	signed 3D impact parameter of the track
$d_{\text{3D}}/\sigma_{\text{3D}}$	signed 3D impact parameter significance of the track
`trackDistance`	distance between the track and the jet axis at their point of closest approach
	PID
q	electric charge of the particle
$m_{\text{t.o.f.}}$	mass calculated from time-of-flight
dN/dx	number of primary ionisation clusters along track
`isMuon`	if the particle is identified as a muon
`isElectron`	if the particle is identified as an electron
`isPhoton`	if the particle is identified as a photon
`isChargedHadron`	if the particle is identified as a charged hadron
`isNeutralHadron`	if the particle is identified as a neutral hadron

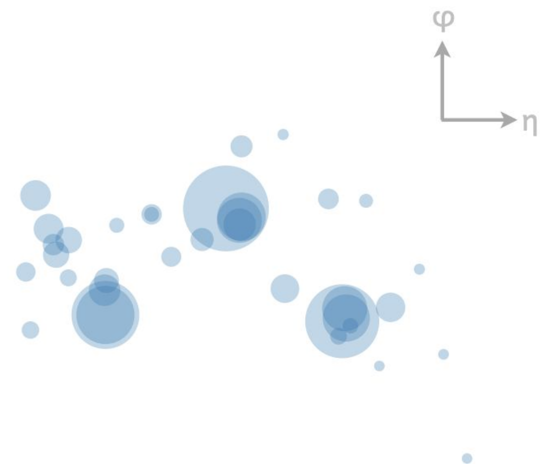
Inputs:
50 particles/jet

Training details:

- 1M jets split equally between classes
- 30 epochs
- Still room for improving the training details

Jet as “particle” cloud





Point cloud:

- points (un-ordered)
- input features: (x,y,z) 3D coordinates

Learn “local” structure, move to more “global” features

Particle cloud:

- particles (un-ordered)
- input features:
  - 2D coordinates (eta x phi)
  - momentum
  - charge/ID
  - displacement …

Graph Neural networks:

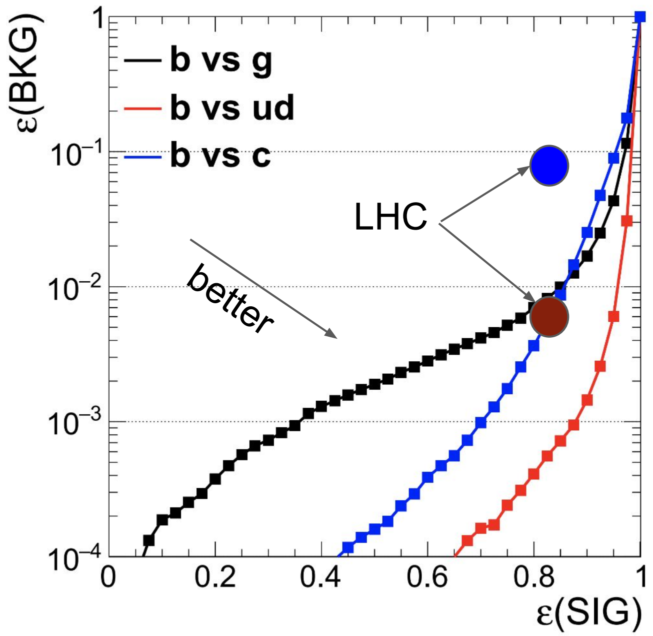
Generalizing Convolutional neural network for un-ordered/sparse images

DGCNN: [arXiv:1801.07829]

ParticleNet: [arXiv:1902.08570]

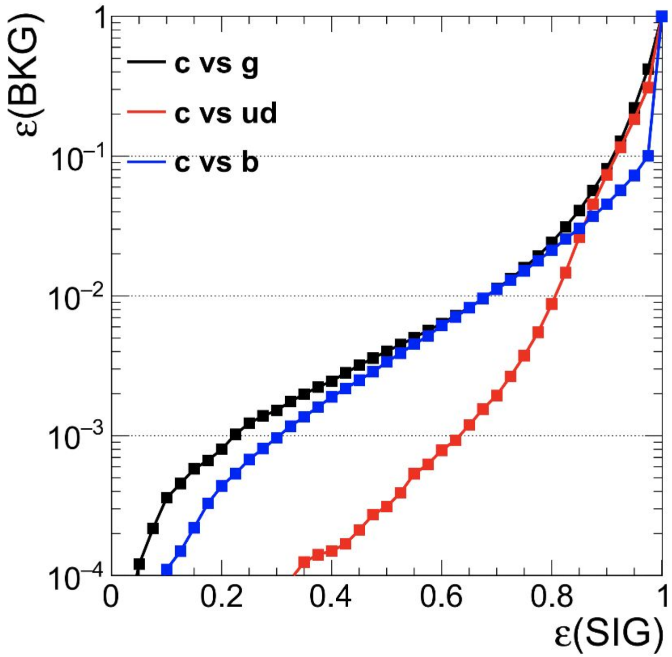
Performance (b/c)

b-tagging



WP	Eff (b)	Mistag (g)	Mistag (ud)	Mistag (c)
Loose	90%	2%	0.2%	3%
Medium	80%	0.7%	<0.1%	0.4%

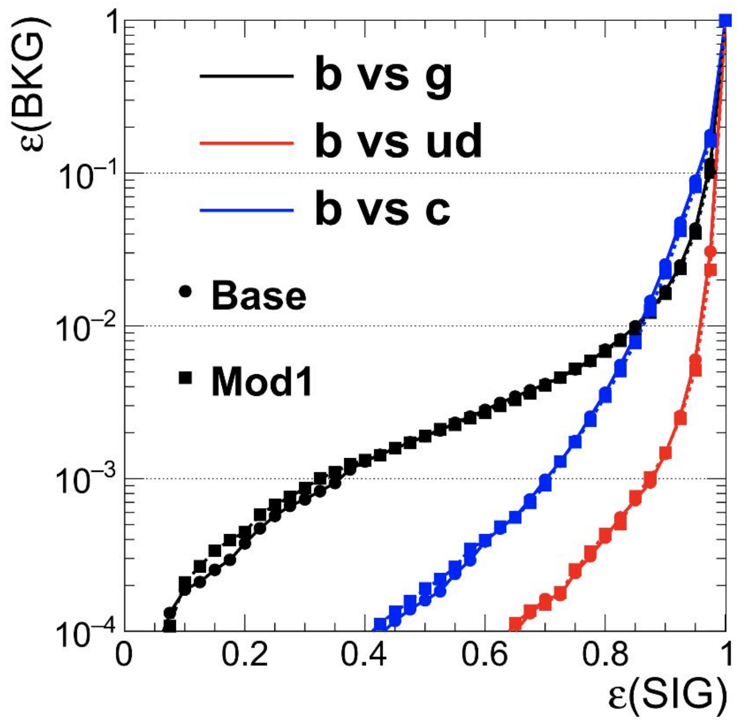
c-tagging



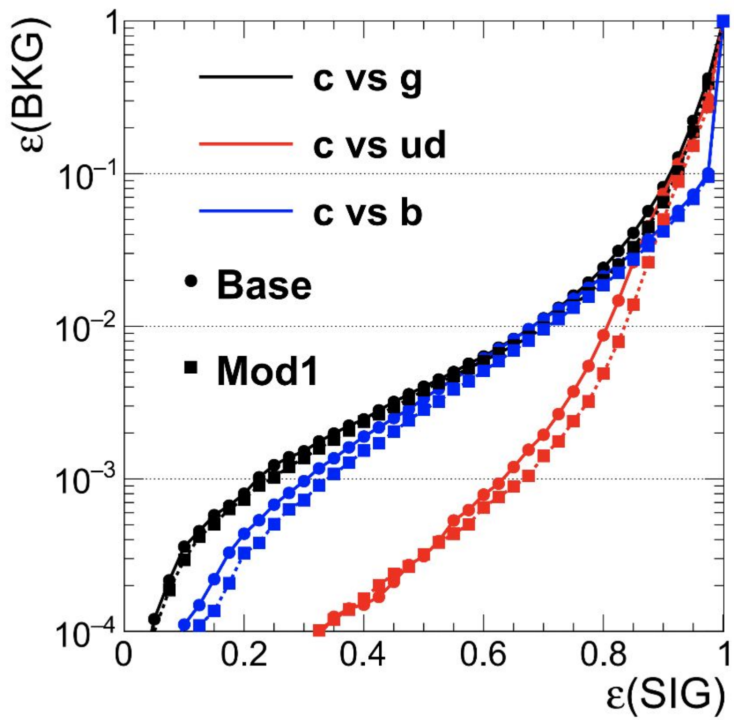
WP	Eff (c)	Mistag (g)	Mistag (ud)	Mistag (b)
Loose	90%	8%	7.5%	5%
Medium	80%	3%	0.9%	2.5%

Performance (smaller beampipe / closer vtx)

b-tagging



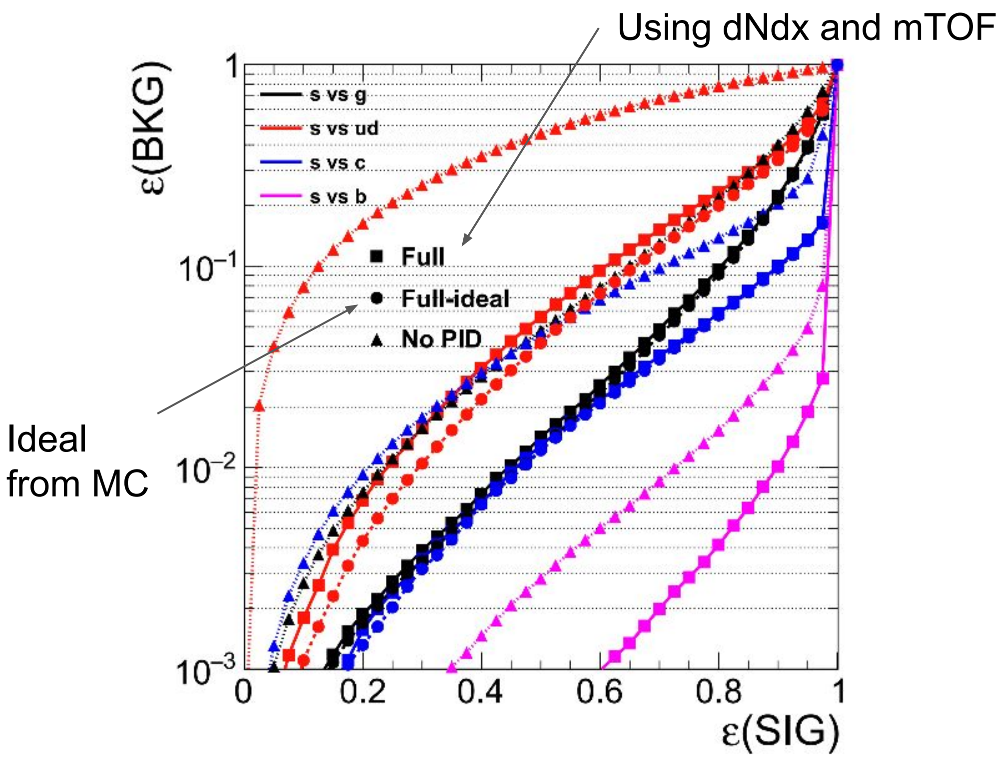
c-tagging

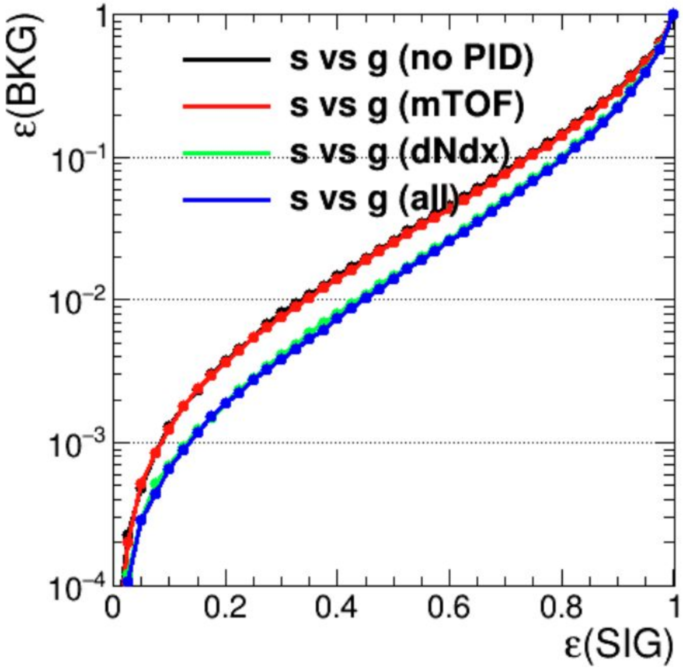


Mod1: additional vertex layer @1 cm

- Effect is small in b-tagging:
  - Effect is significant in c-tagging, as expected
  - Gain ~ 10-30% background rejection (vs. gluon and light)
    - possibly explore thinner beam-pipe (to limit MS)

Strange tagging



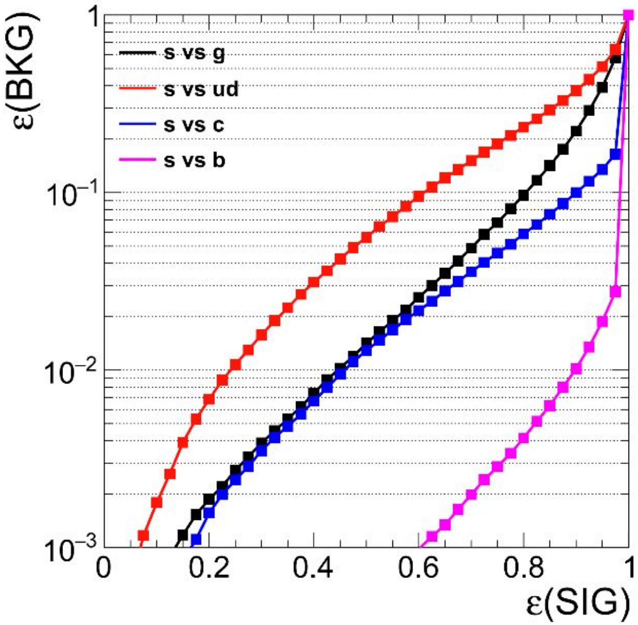


- Small room for improvement on the PID, in particular for strange tagging
  - TOF does not contribute as much as dNdx (30 ps resolution enough?)
    - low pT tracks are not discriminating ?
    - Can be further improved using timing resolution for neutral K_L vs n ?

REQUIRES FURTHER INVESTIGATION

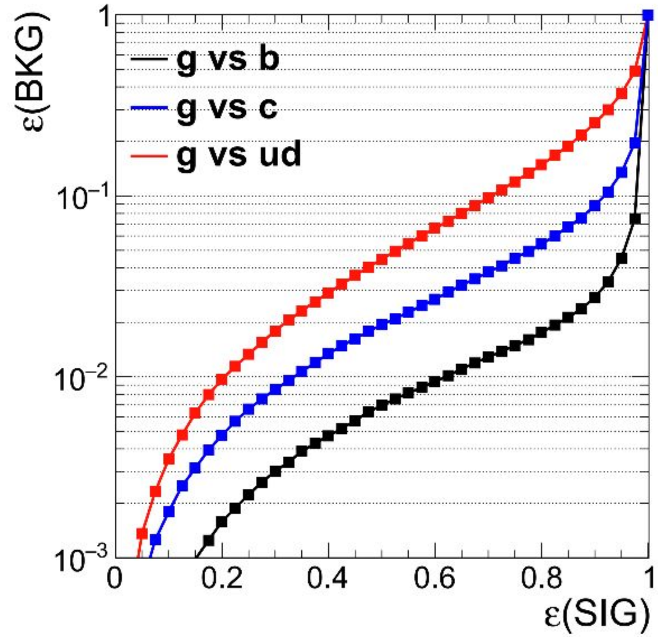
Performance (strange/gluon)

strange-tagging



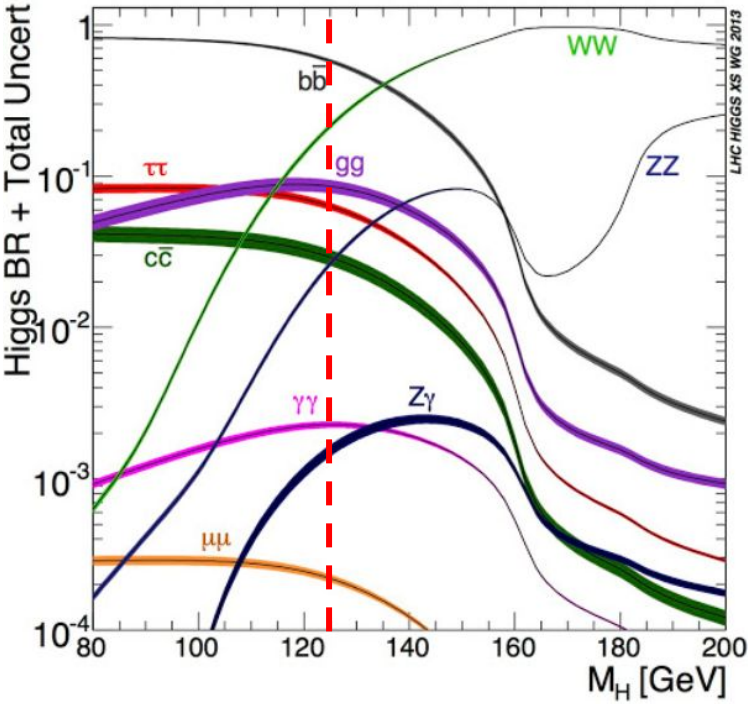
WP	Eff (s)	Mistag (g)	Mistag (ud)	Mistag (c)	Mistag (b)
Loose	90%	20%	40%	10%	1%
Medium	80%	10%	20%	6%	0.4%

gluon -tagging



WP	Eff (g)	Mistag (ud)	Mistag (c)	Mistag (b)
Loose	90%	25%	9%	3%
Medium	80%	15%	5%	2%

Higgs couplings: H→ cc



$\sqrt{s}$ (GeV)	240		365	
Luminosity (ab^{-1})	5		1.5	
$\delta(\sigma BR)/\sigma BR$ (%)	HZ	$\nu\bar{\nu}$ H	HZ	$\nu\bar{\nu}$ H
H → any	±0.5		±0.9	
H → $b\bar{b}$	±0.3	±3.1	±0.5	±0.9
H → $c\bar{c}$	±2.2		±6.5	±10
H → gg	±1.9		±3.5	±4.5
H → W^+W^-	±1.2		±2.6	±3.0
H → ZZ	±4.4		±12	±10
H → $\tau\tau$	±0.9		±1.8	±8
H → $\gamma\gamma$	±9.0		±18	±22
H → $\mu^+\mu^-$	±19		±40	
H → invis.	< 0.3		< 0.6	

Ref: Patrick's talk at the CDR Symposium; March 2019

FCCee: σ_{ZH}~200fb, L~5 ab^{-1} (2 IP): **~1M ZH**
[600k H→bb, 100k H→gg, **30k H→cc**]

Use Loose WP:
[c-tag: 90%, b-mistag: 5%, g-mistag: 10%

- **Scenario 2:** Z(→νν)H
  δ(σxBR)/σxBR (%) ~ 1.5 [no systematics]
- **Scenario 1:** Z(→all)H
  δ(σxBR)/σxBR (%) ~ 0.7 [no systematics]

- **Stat limit [i.e. no BKG]:**
  δ(σxBR)/σxBR (%) ~0.6%
- **No BKG rejection:**
  δ(σxBR)/σxBR (%) ~2.9%

Results look promising

Higgs couplings: H$\rightarrow$ ss

BR(H$\rightarrow$ ss) = BR (H$\rightarrow$ cc) $(m_s/m_c)^2$ ~ 2.3 10^{-4}

Back-of-the envelope estimates

THOROUGH STUDIES NEEDED

FCCee: σ_{ZH}~200fb, L ~ 5 ab^{-1} (2 IP): **~1M ZH**
[600k H$\rightarrow$bb, 100k H$\rightarrow$gg, 30k H$\rightarrow$cc, **200 H$\rightarrow$ ss**]

Use Loose WP:
[**s-tag: 90%**, **g-mist: 10%**, c-mist: 1%, b-mist: 0.4%

- **Scenario 1:** Z($\rightarrow$all)H:

N_{ss} = 150, N_b = 1000
(neglecting ee$\rightarrow$ VV backgrounds)

δ(σxBR)/σxBR (%) ~ 21 % (~ 5σ) [no systematics, only higgs backgrounds, no combinatorics]

- **Scenario 2:** Z($\rightarrow$vv)H:

N_{ss} = 30, N_b = 200
(neglecting ee $\rightarrow$ vvqq and ee $\rightarrow$ qq, can be important given large q $\rightarrow$ s fake prob.)

δ(σxBR)/σxBR (%) ~ 49% (~ 2σ) [no systematics]

Summary & outlook

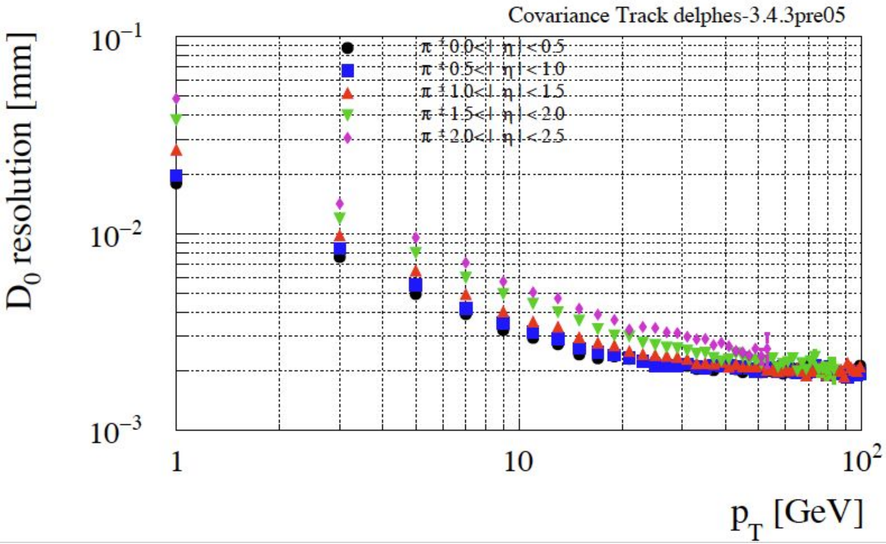
- A first version of a jet identification algorithm based on **PF candidates** and **PID** and **advanced ML** in place
  - Multi-class classifier b/c/s/ud/g
    - Results promising, in particular for charm and strange tagging
- PRELIMINARY conclusions:
  - adding an additional vertex layer does not tremendously improve b-tagging performance (resolution of ~ 2um already outstanding)
    - but improves charm tagging
  - There seems to be room for improving strange tagging with more powerful PID
- Next [short-term] steps:
  - propagate detector design choice to final sensitivity
  - address tagger calibration (at the Z pole)
  - Provide framework for training/testing with the FCCSW

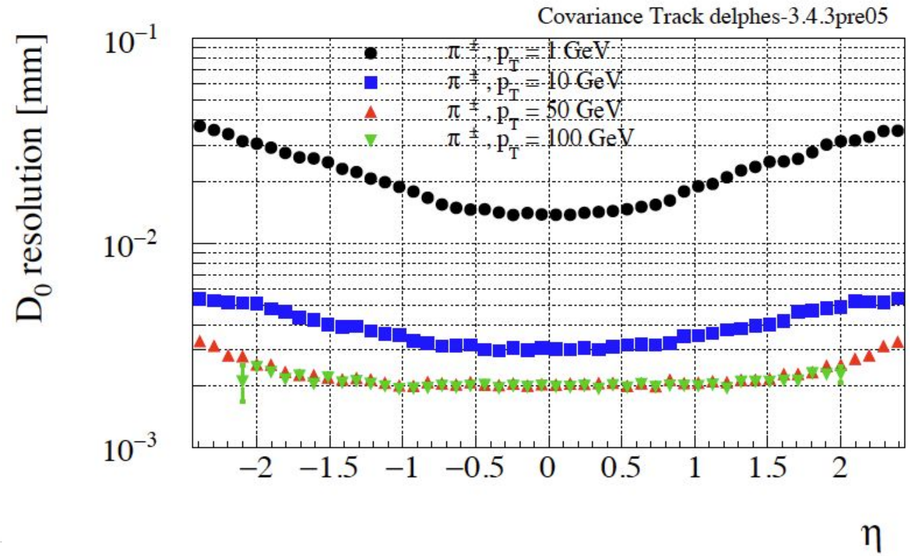
Backup

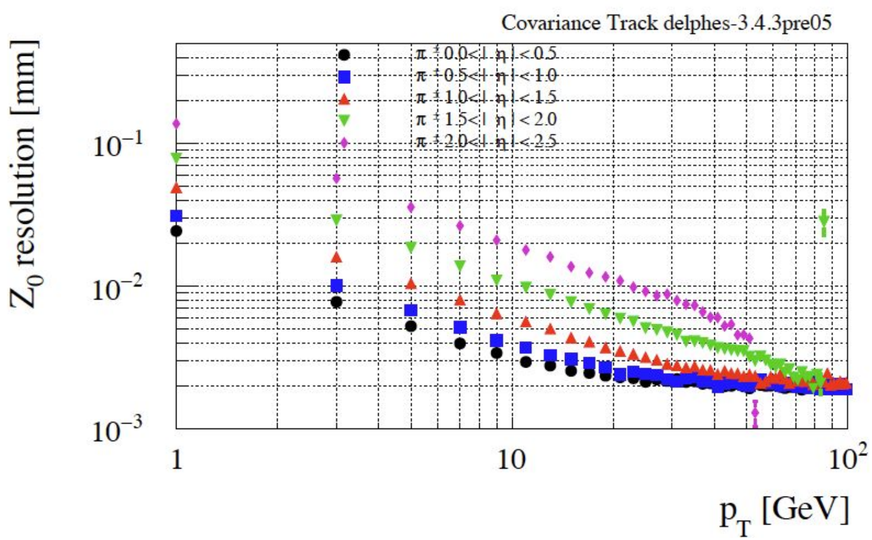
Impact parameter performance

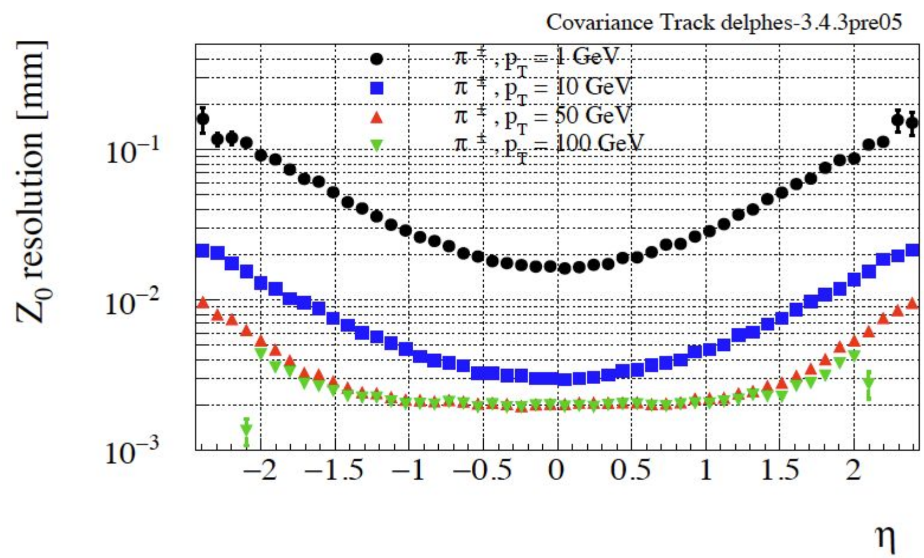
Credits to Sylvie Braibant

IDEA detector:







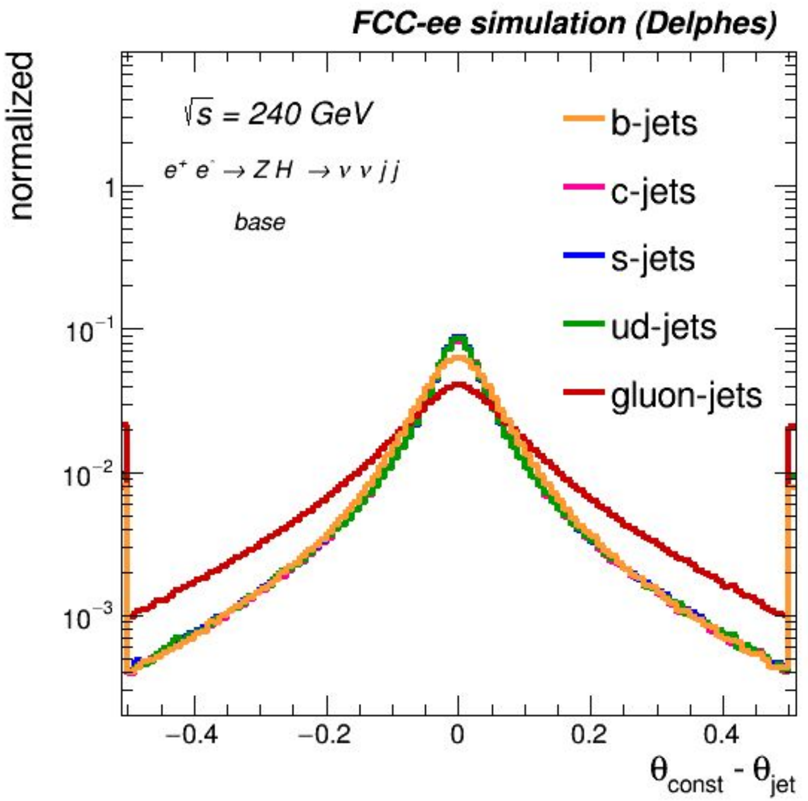


2µm IP resolution at high-p_T

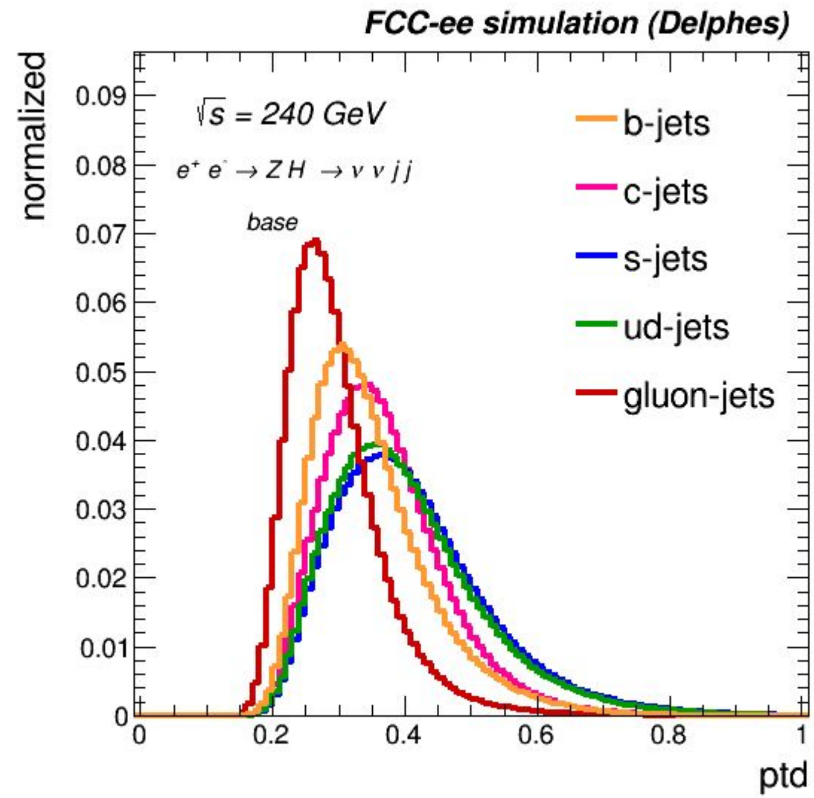
Input variables

- Comparison of input distributions for different jet flavors

Projection || to jet axis

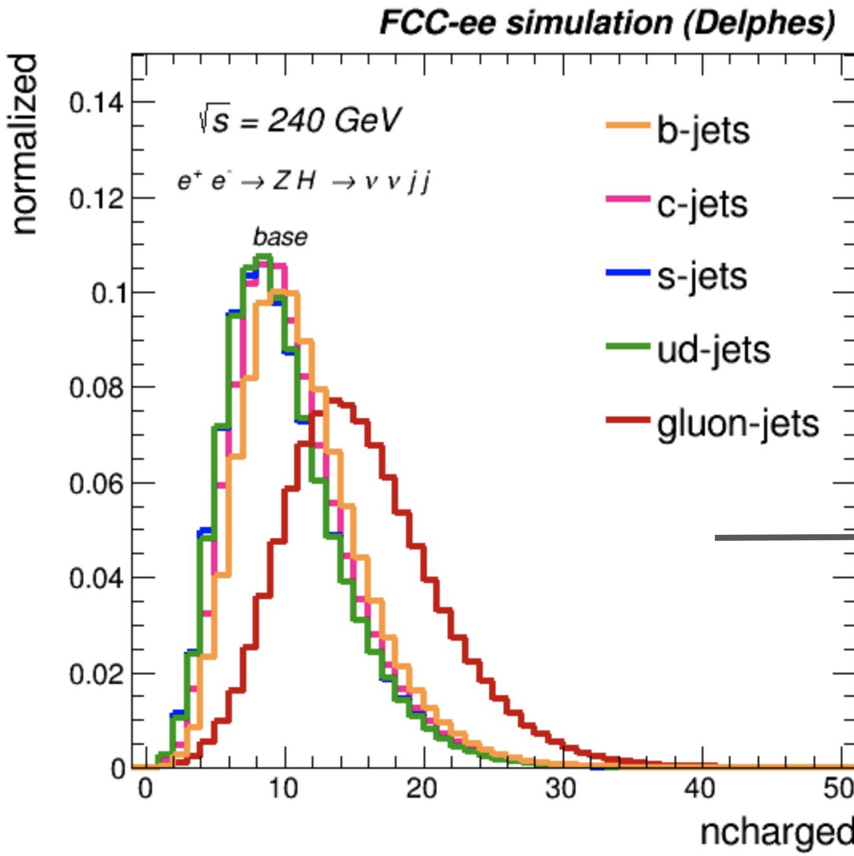


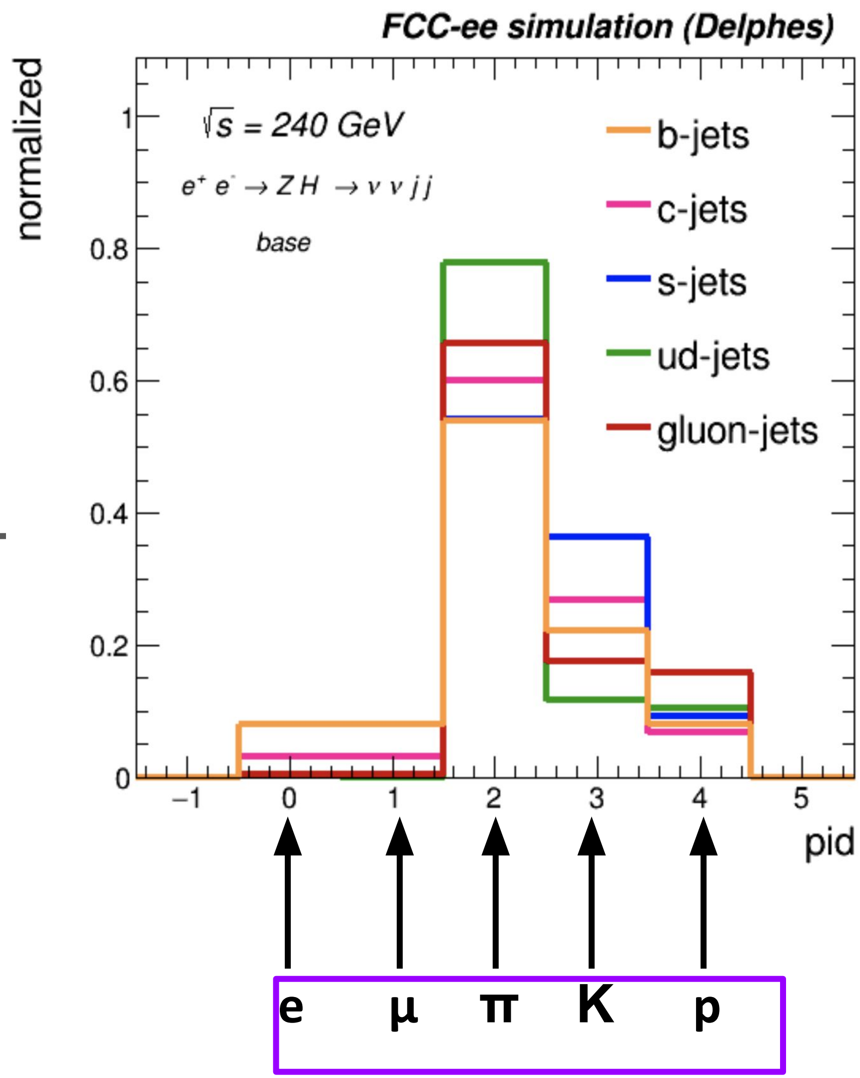
p_TD



- More comparisons:
  https://selvaggi.web.cern.ch/selvaggi/FCC/FCCee/FlavourTagging/

Performance w/ PID





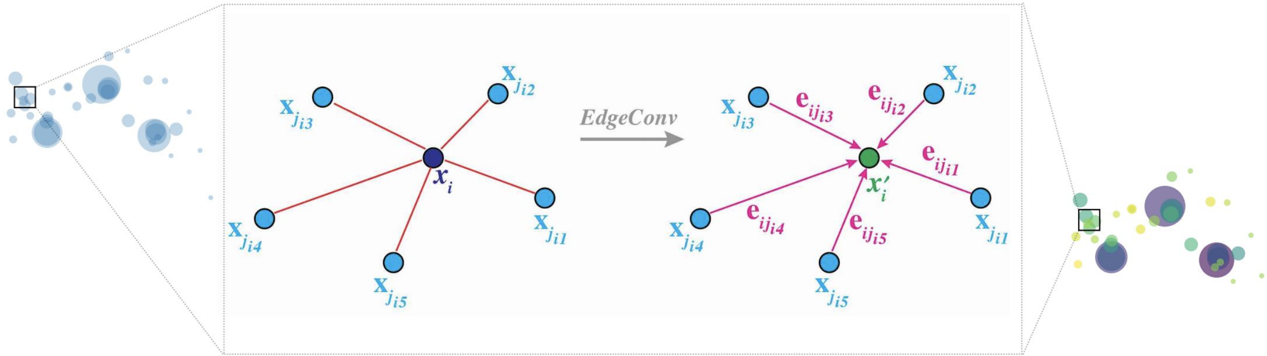
no PID: only charge

realistic: e, μ, mtof, dNdx

perfect PID: e, μ, + π, K, p from MC truth

Convolution on point cloud: EdgeConv

DGCNN: [arXiv:1801.07829]



EdgeConv: convolution on a graph

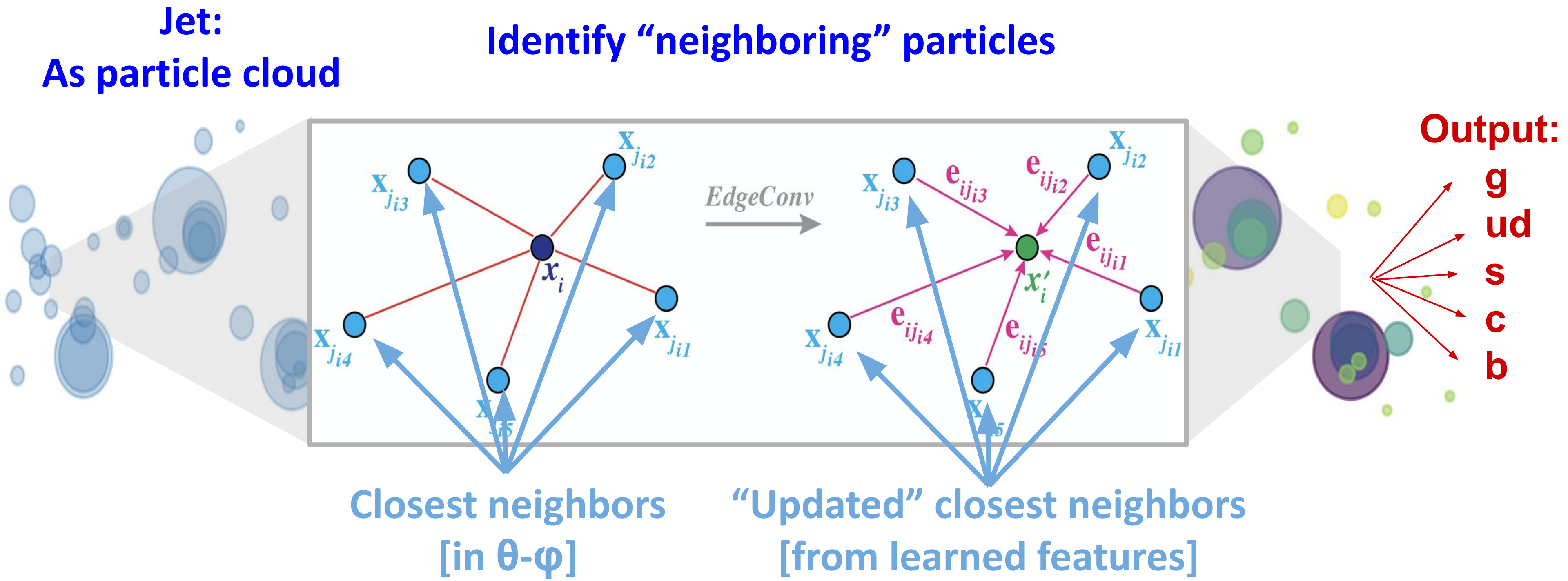
- **point cloud** is treated as **graph**, where each point is a **vertex**
- **local patch** defined by finding k-nearest neighbours
- **convolution** function:
    - define "edge feature" for each center-neighbour pair
        - $e_{ij} = h(x_i, x_j)$
    - aggregate all the features **symmetrically**:
        - $x'_i = \text{mean}_j\, e_{ij}$

Key point:

Generalizing CNN for
un-ordered/sparse images

Flavour tagging using ParticleNet

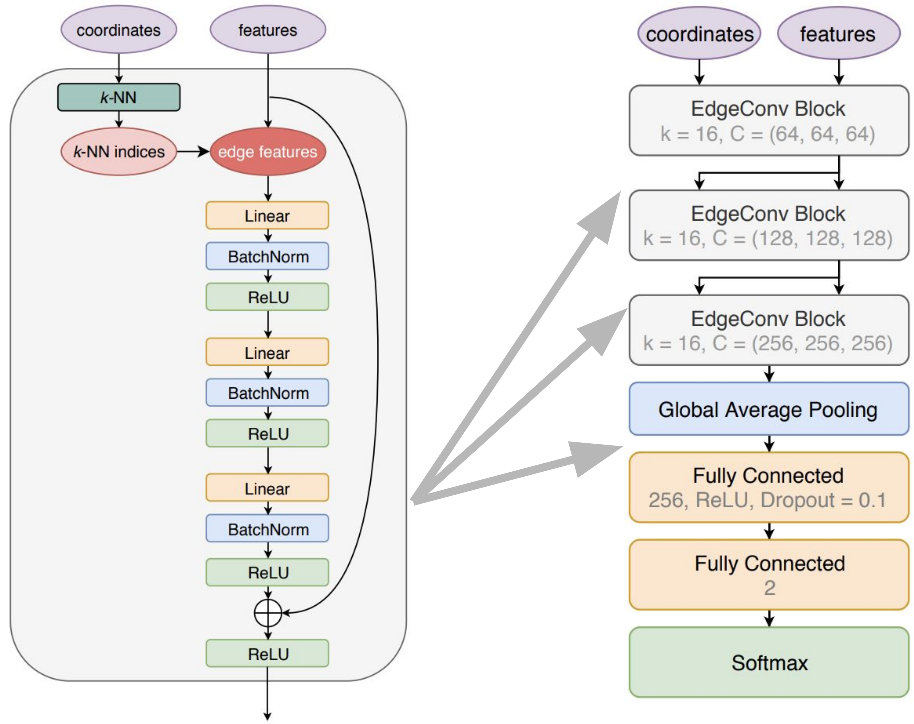
- Developing a flavour tagging algorithm based on ParticleNet
  - Jet is represented as a "particle cloud"
- Follow a hierarchical learning approach:
  - **First:** Learn "local" structures; **Then:** move to more "global" features
  - Treat the particle cloud as a graph
    - **Particles** are the **vertices** of the graph
      Relationships between the particles are the **edges** of the graph



ParticleNet

ParticleNet: [arXiv:1902.08570]

- **local neighborhood** information automatically incorporated
- **EdgeConv** layers can be **stacked** (as CNNs), and learn **local** (shallow layers) and **global** features (deep layers)
- **new features** provide new coordinates (in some abstract latent space) to compute “local patch” in new iteration



EdgeConv block *ParticleNet architecture*

Designing a jet flavour tagging algorithm

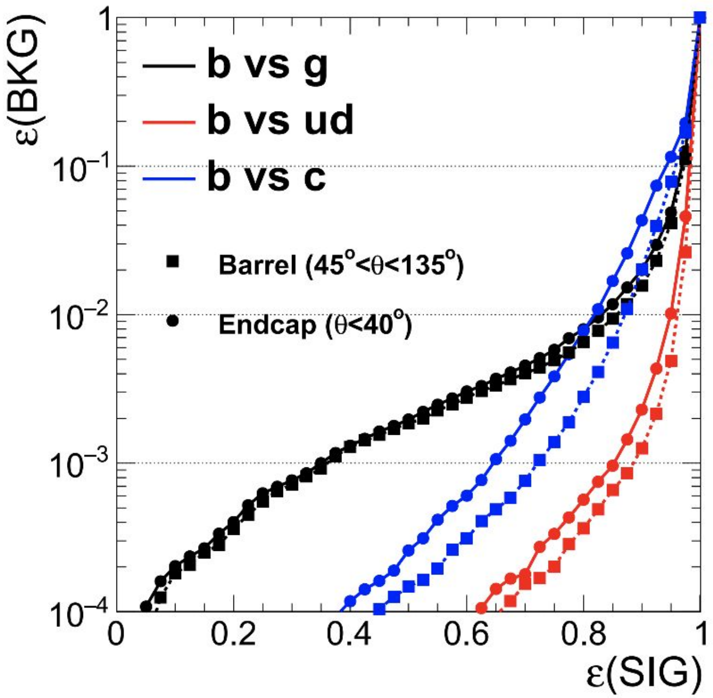
- How to represent a jet is one of the key aspects of algorithms for jet tagging
  - Improve performance → extend physics reach
  - Lead to fresh insight into jets → deepen our understanding of jet physics

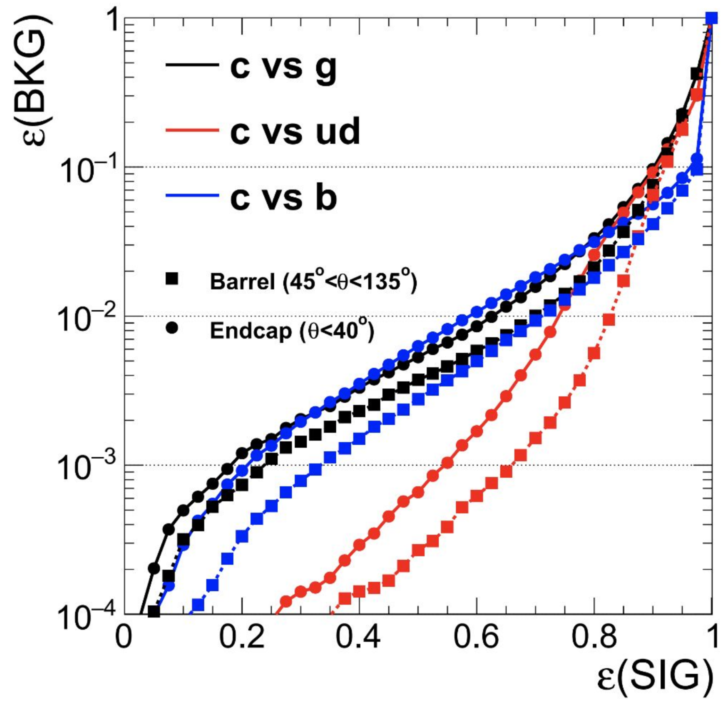
- Particles [associated to each jet] are intrinsically unordered
  - i.e., ordering by p_T(particle) or displacement from PV: suboptimal
  - Primary information: 2D coordinates in theta-phi space
  - Include additional features / particle: energy, displacement, charge, track quality, PID ...

Performance vs theta (b/c)

b-tagging | **c-tagging**

PRELIMINARY !! (LOW STATS TRAINING)





Tracking in Delphes

pattern recognition
track fitting

Full Simulation

-most accurate
-least flexible
-does not allow for change of geometry

parameterisation
$(d_0, d_z, p, \mathrm{ctg}\,\theta, \varphi)$

Delphes

TkLayout

-standalone tool
-predicts tracking performance for a given geometry
-allows for quick turnaroud
- no further implementation needed in Delphes
-requires intermediate step

FastTrackCovariance

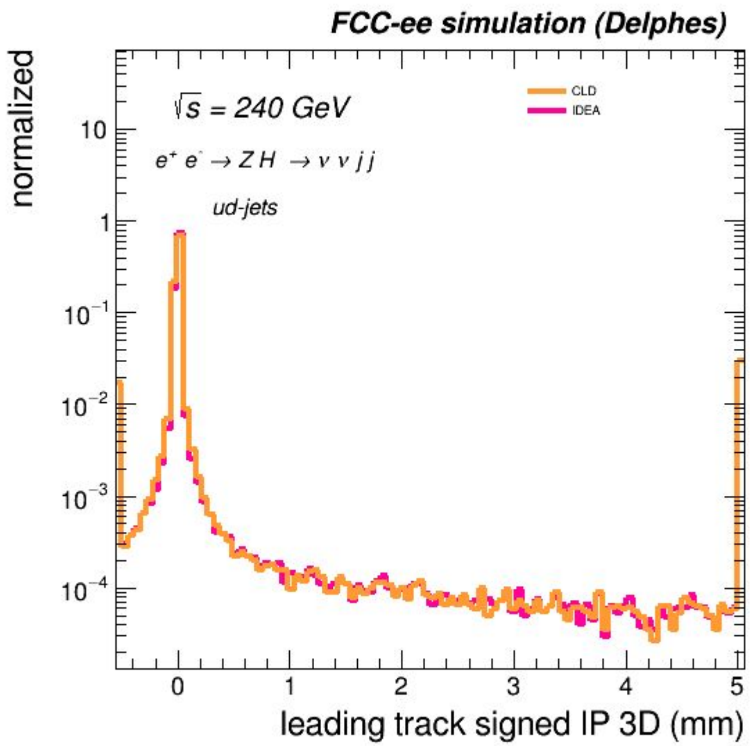
-(for now) standalone tool
-can be plugged into Delphes
-predicts performance given geometry
-allows for quick turnaroud
-not implemented in Delphes yet

23

Comparison: IDEA vs. CLD

- No big differences between in input variables between IDEA & CLD
  - small difference in material budget observed on light jets since dxy ~ 0
    - expect slightly better performance for IDEA detector for discrimination vs light

ud-jets



c-jets