Jet Flavour Tagging at e^+e^-

Franco Bedeschi, Loukas Gouskos, Michele Selvaggi

[full set of results: arXiv:2202.03285]

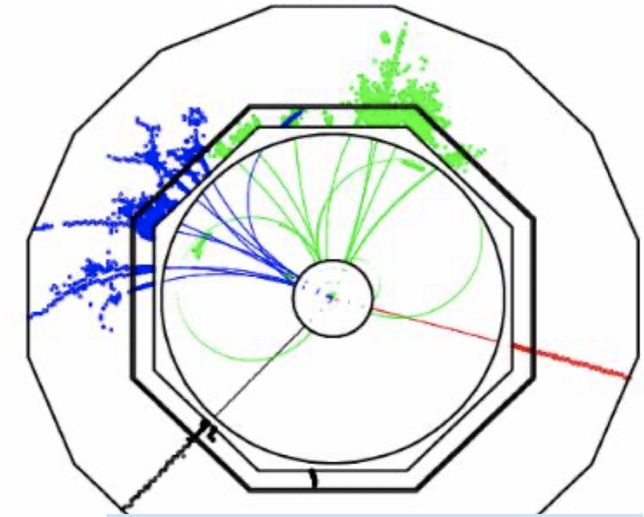
ECFA Higgs/Top/EW group Workshop
April 2022

Introduction

- Jet flavor identification ("tagging"): necessary tool to maximize physics outcome at colliders

- Today: Focus on e+e- colliders
  - provide a very clean environment
  - Much lower occupancy
  - no pileup compared to hadron colliders

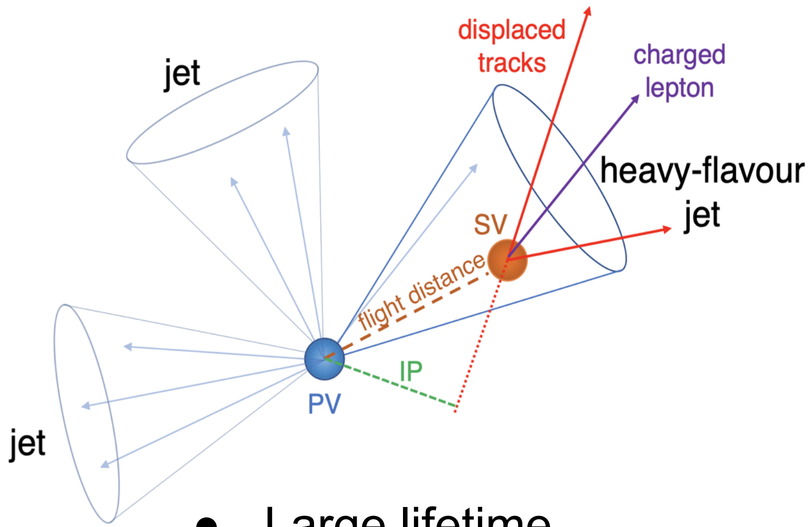
- Scope of this work:
  Build a general framework for developing flavor tagging algorithms for future colliders
  - Fast detector simulation
    - Understand detector requirements/optimize design
    - e.g., Vertexing and PID capabilities of the FCCee detectors
  - Develop a versatile jet flavor tagger for FCCee
    - Identify with high purity gluon / light / strange / charm / bottom quarks



e^+e^-: $Z(\rightarrow\mu\mu)H(\rightarrow bb)$

Basics of flavour tagging

bottom/charm-tagging

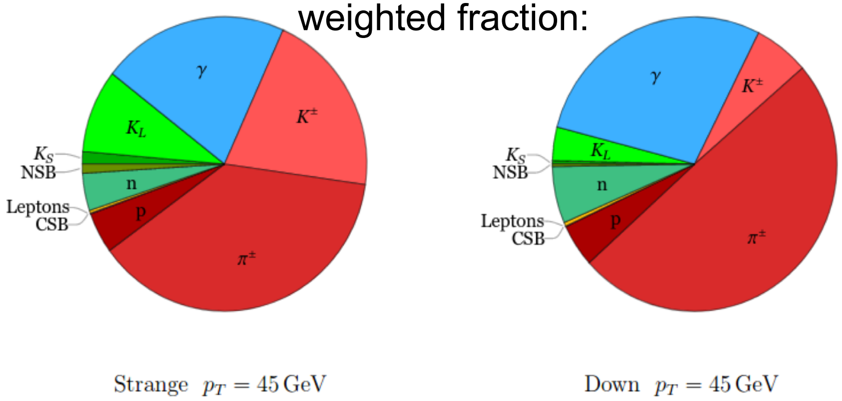


- Large lifetime
- Displaced vertices/tracks
- Large track multiplicity
- Non-isolated e/μ

strange-tagging

[2003.09517]

Momentum weighted fraction:



- Large Kaon content
  - Charged Kaon as track:
    - K/pi separation
  - Neutral Kaons:
    - $K_S \rightarrow \pi\pi$, K_L

Detector constraints:

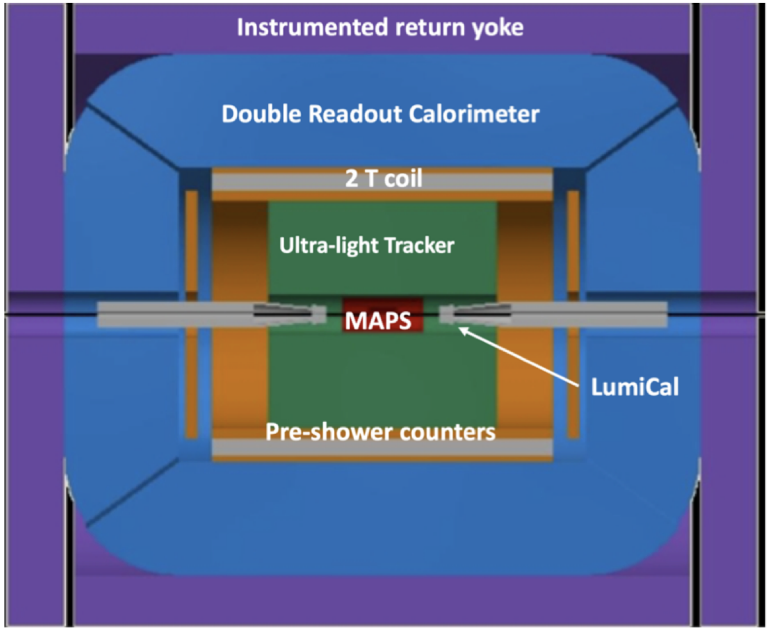
Pixel/tracking detectors
- Little material, spatial resolution, precise track alignment

PID detectors:
- timing capabilities, energy loss (gas/silicon)

FCCee detector

- Ideal for flavor 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
    - dE/dx (Si tracker) -- Cluster counting dN/dx (Drift)
    - Time of flight -- timing layer

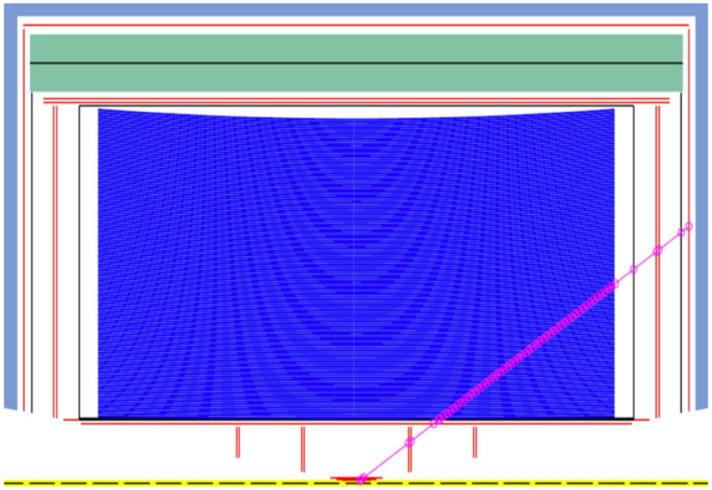
IDEA



Simulation

- Detector response based on Delphes:
  - Including FastTrackCovariance
  - Computes:
    - full track covariance matrix
      - Including multiple scattering
    - 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

IDEA



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

- Jets clusters with the generalized-kT algorithm using *p*=-1
  - Similar to the anti-k_T algorithm [IRC safe]

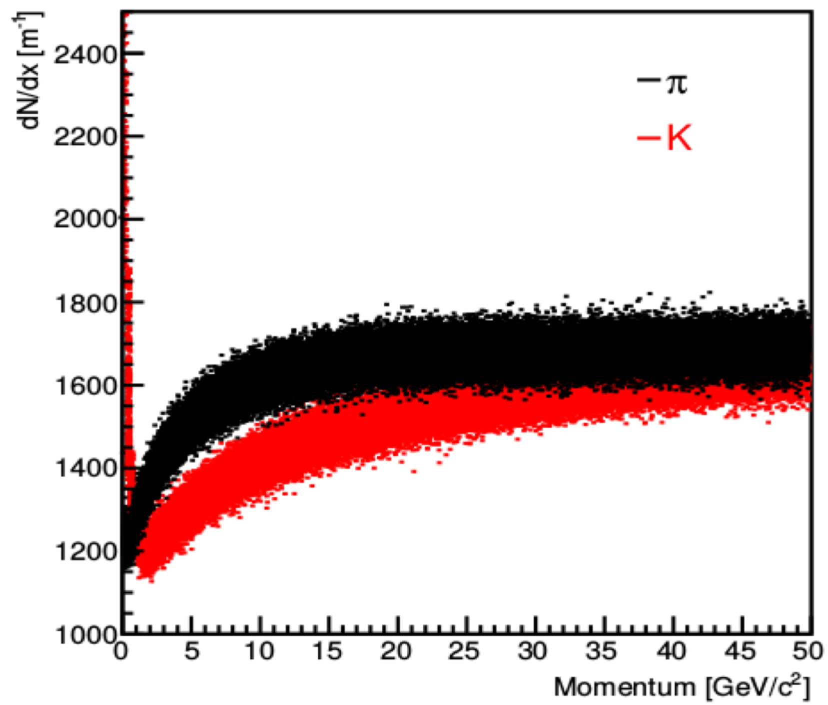
Cluster counting dN/dx

- 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:

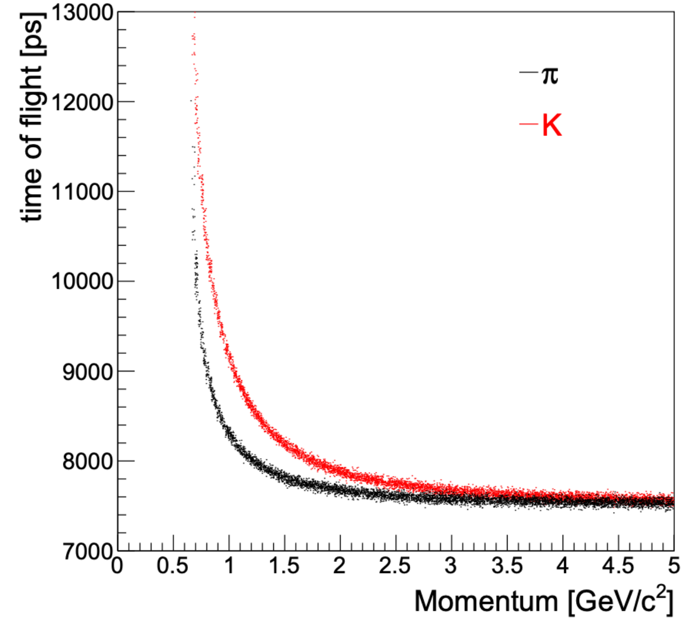
90% He / 10 % Isobutane



Time-of-flight

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

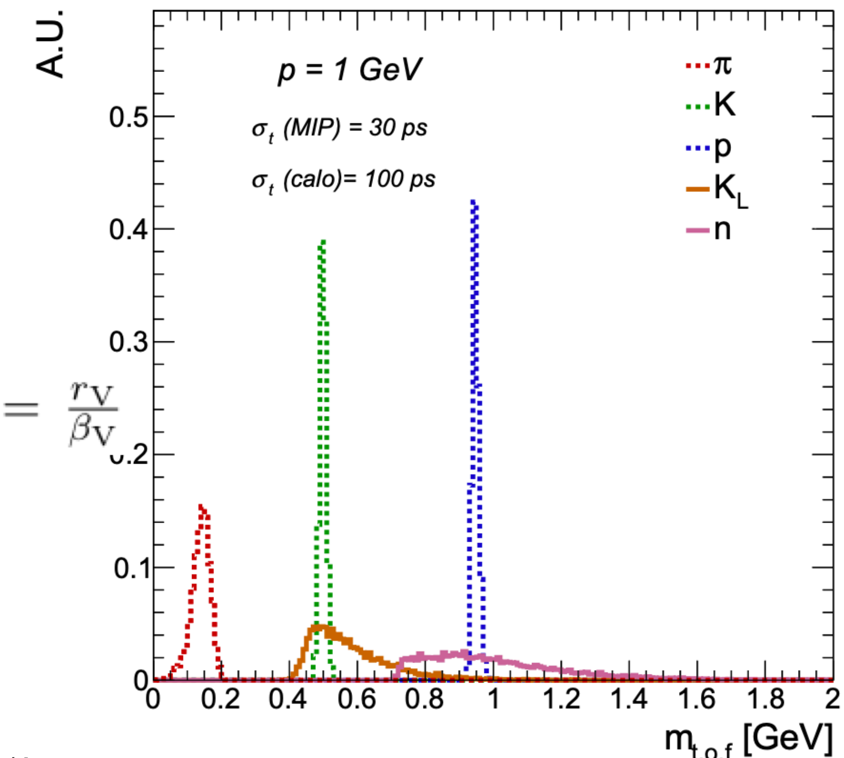
$$t_{\text{flight}} \equiv t_{\text{F}} - t_{\text{V}} = \frac{L}{\beta} = \frac{L\sqrt{p^2+m^2}}{p}$$



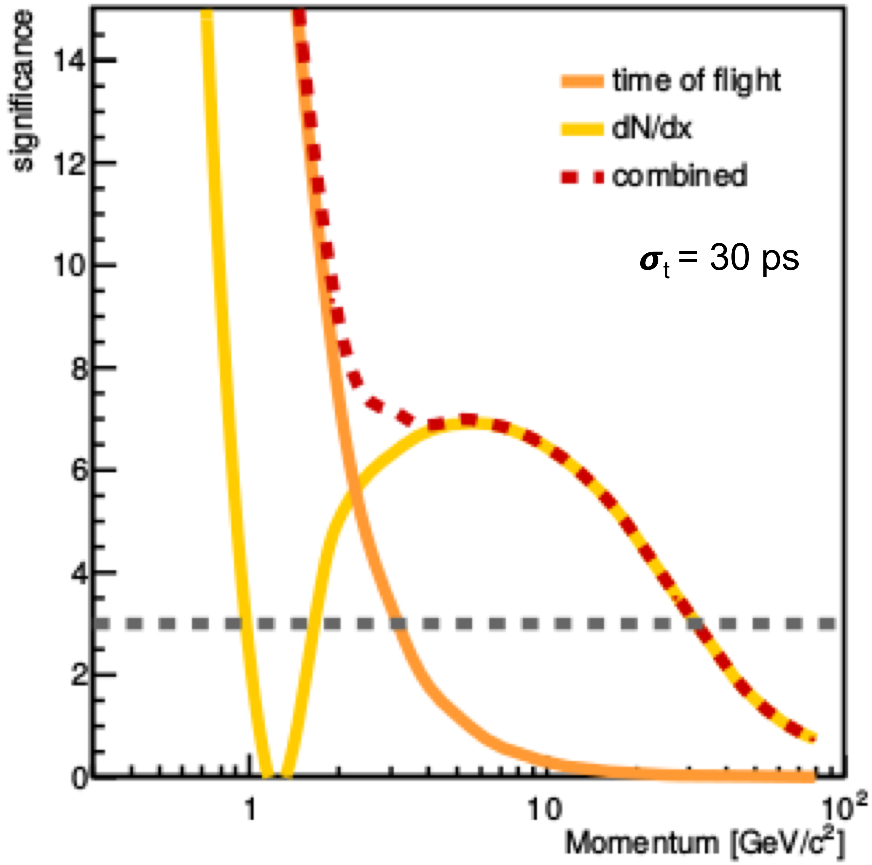
- Need to make assumption on vertex time (crucial for highly displaced K_S) :

```
########################################
#   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
}
```

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



Combined PID

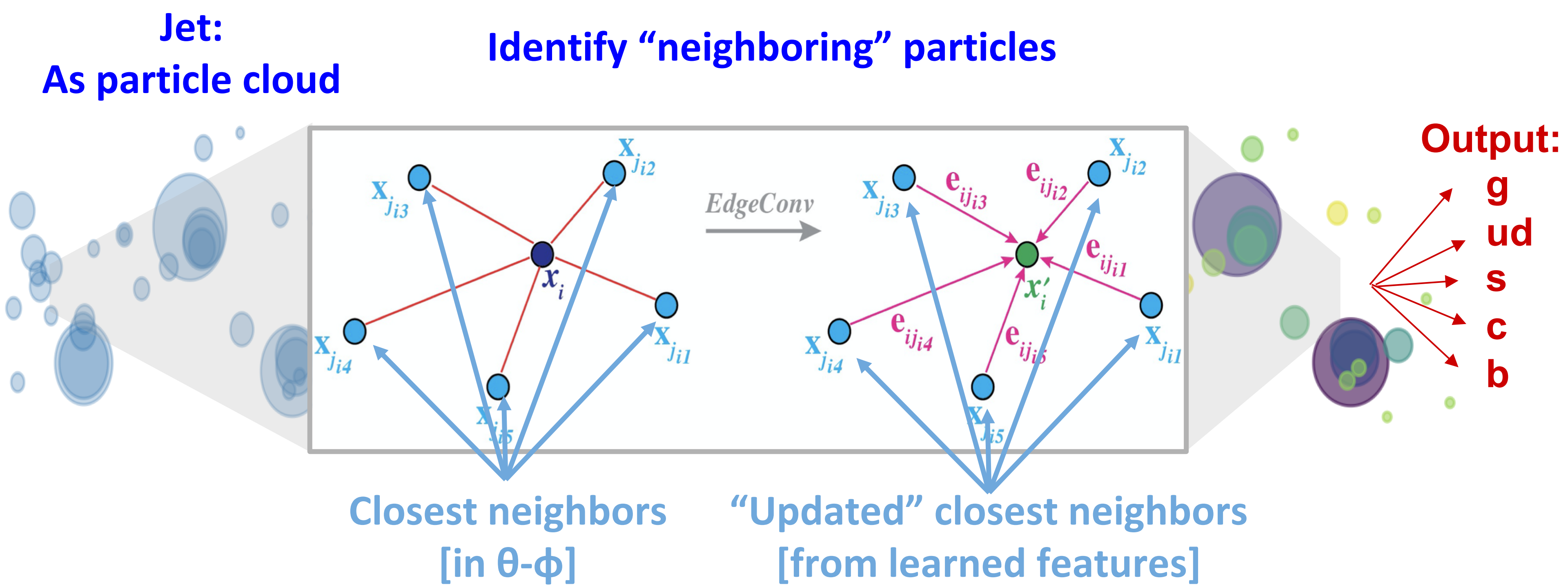


3 std deviation K/pi separation for tracks with p < 30 GeV

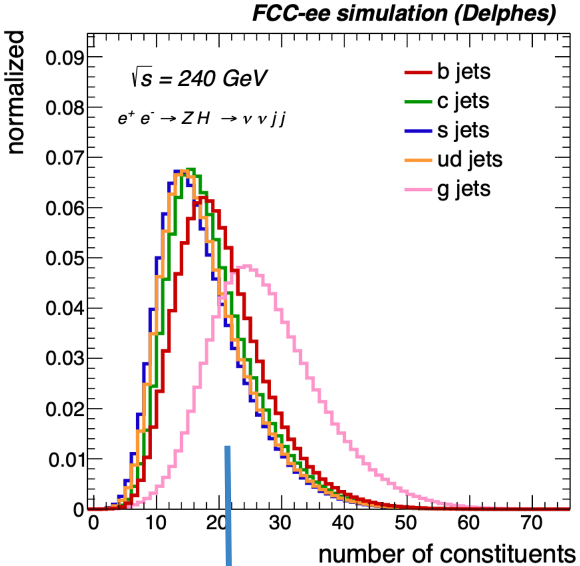
Flavour tagging using ParticleNet

PRD 101 (2020) 5, 056019
CMS-DP-2020-002

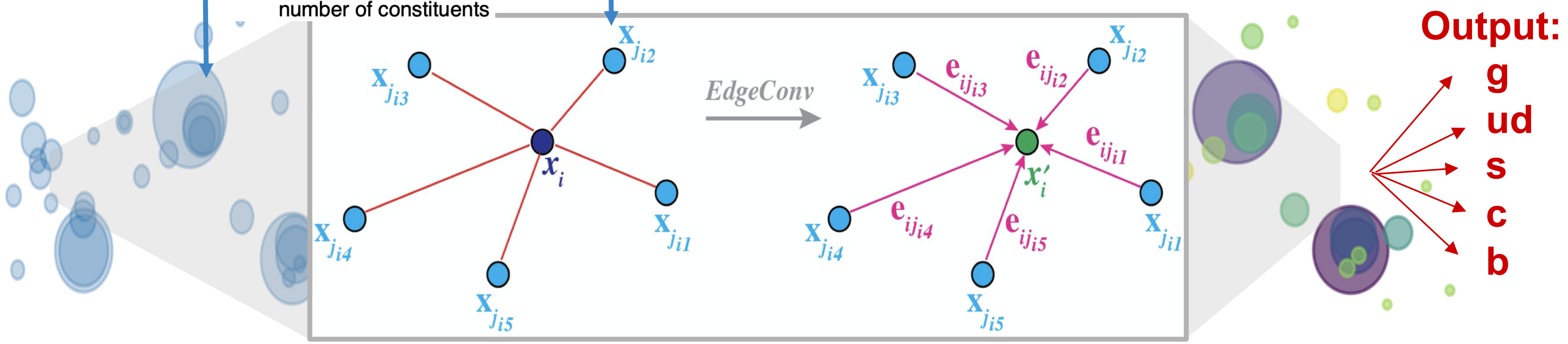
- Flavour tagging algorithm based on ParticleNet
  - Jet is represented as a "particle cloud"
- Follow a hierarchical learning approach:
  - **First:** Learn "local" structures; **Then:** move to more "global" features
  - Treat the particle cloud as a graph
    - **Particles** are the **vertices** of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



Flavour tagging using ParticleNet (II)



Particle features: O(20)/particle | **Particle kinematics, particle charge, Impact parameter (d0, dz) and significance, particle type (el, mu, γ,...)**

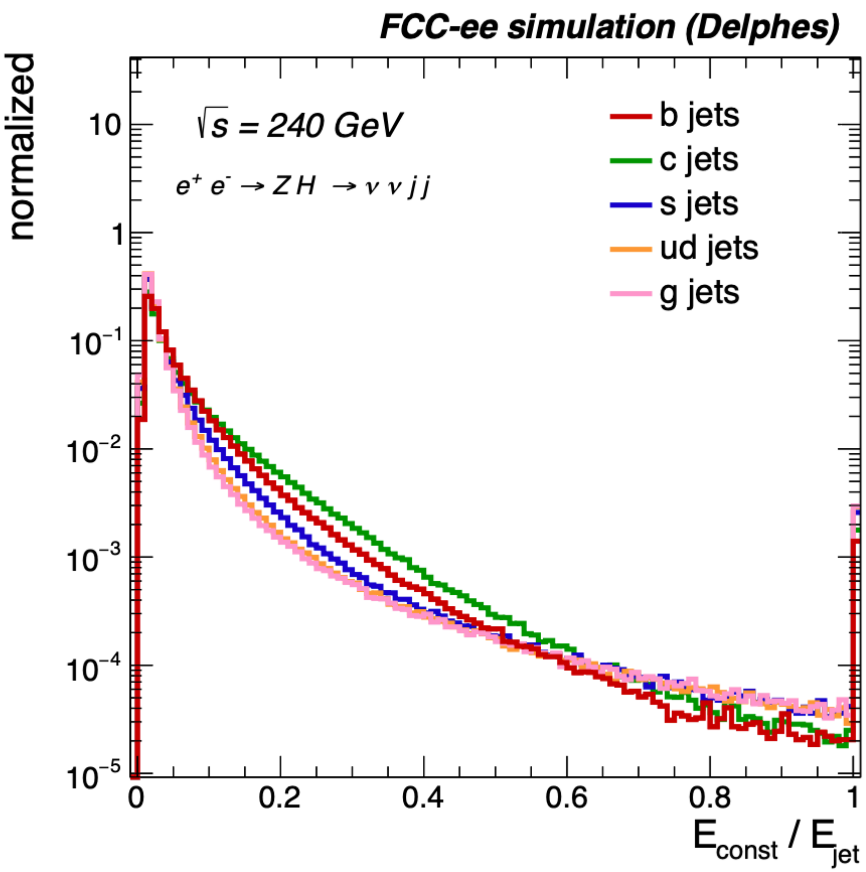


Inputs:
75 particles/jet

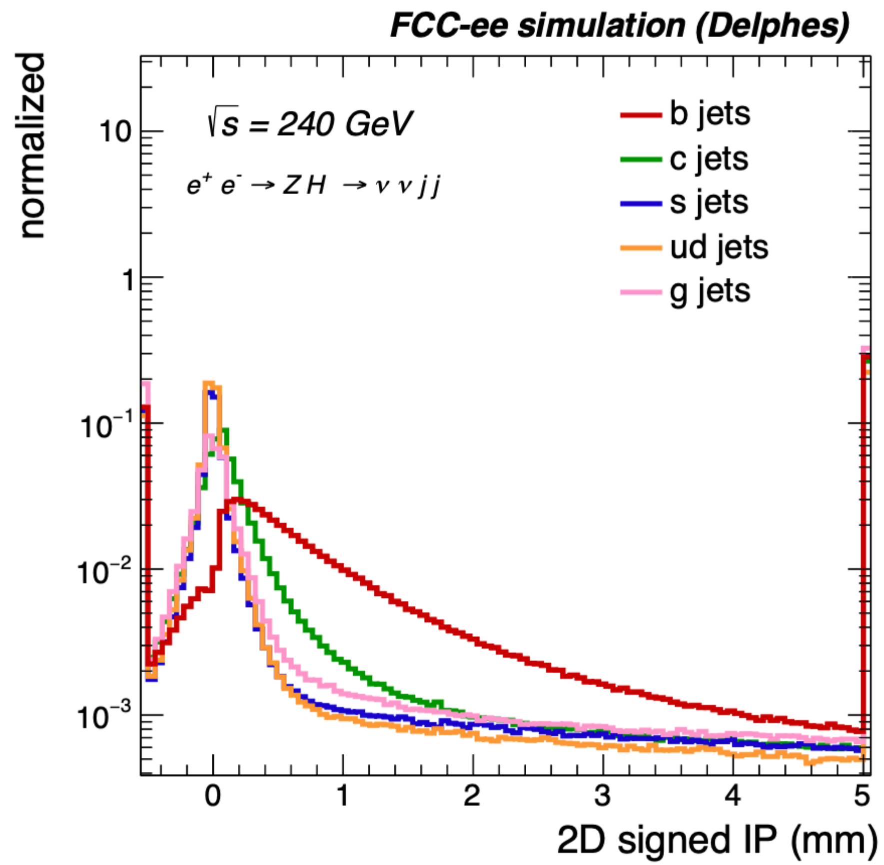
Input variables

- Comparison of input distributions for different jet flavors

Constituent relative energy



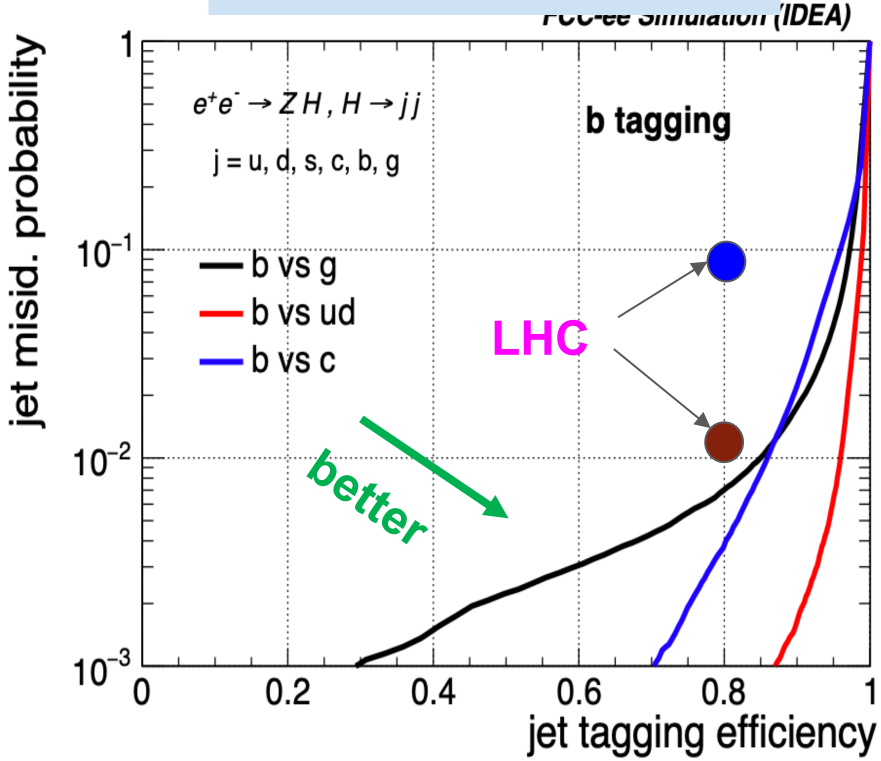
Impact parameter (d_0)



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

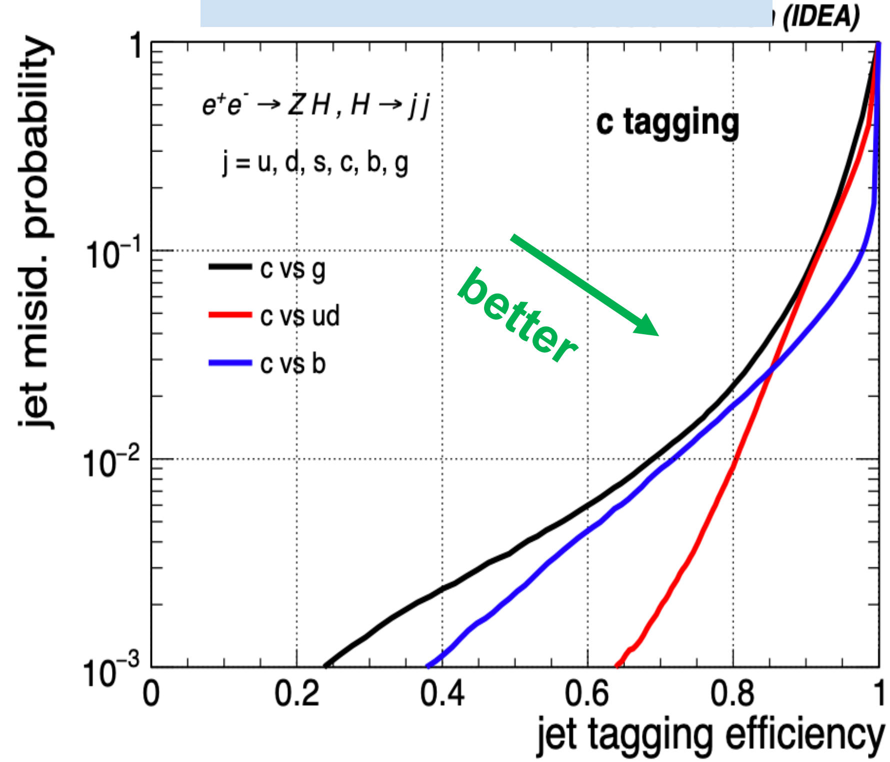
Performance (b/c)

b-tagging



WP	Eff (b)	Mistag (g)	Mistag (ud)	Mistag (c)
Loose	90%	2%	0.1%	2%
Medium	80%	0.7%	<0.1%	0.3%

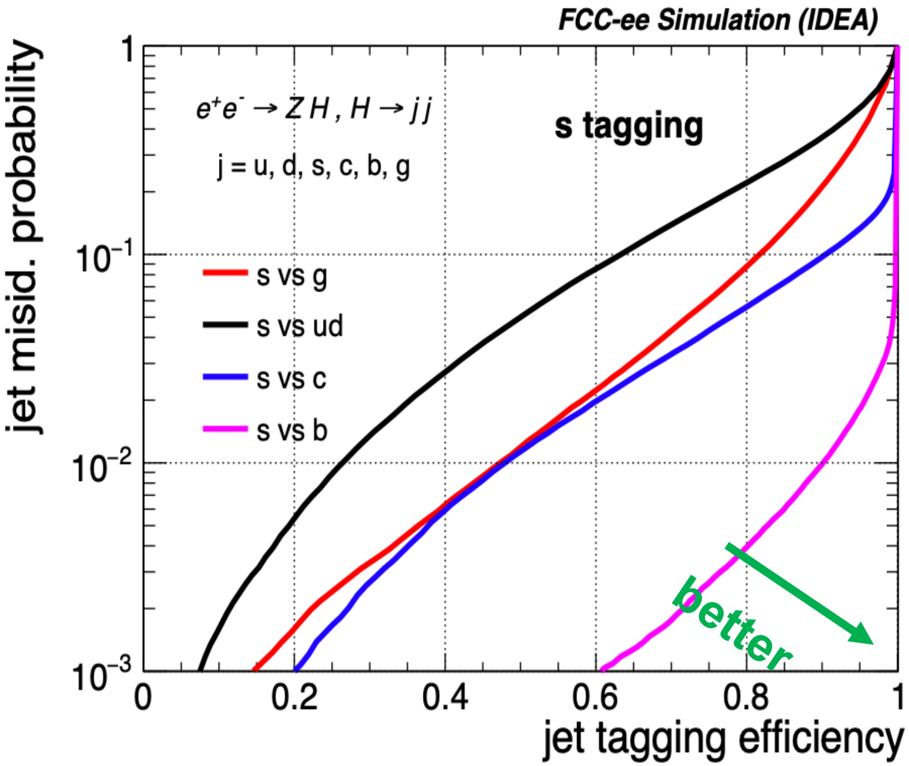
c-tagging



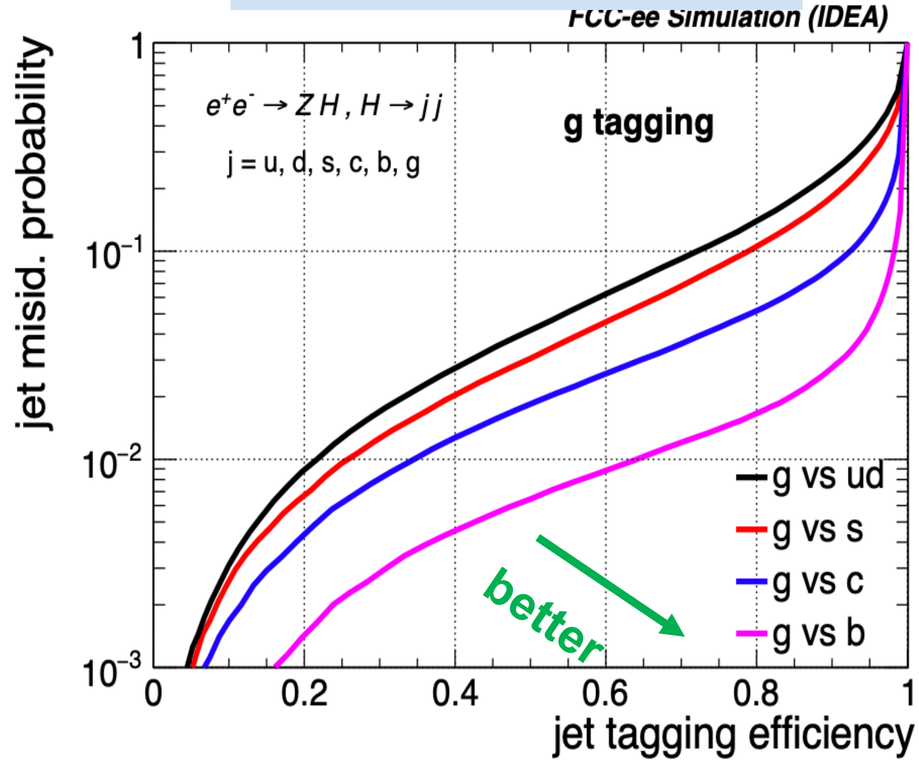
WP	Eff (c)	Mistag (g)	Mistag (ud)	Mistag (b)
Loose	90%	7%	7%	4%
Medium	80%	2%	0.8%	2%

Performance (strange/gluon)

strange-tagging



gluon -tagging

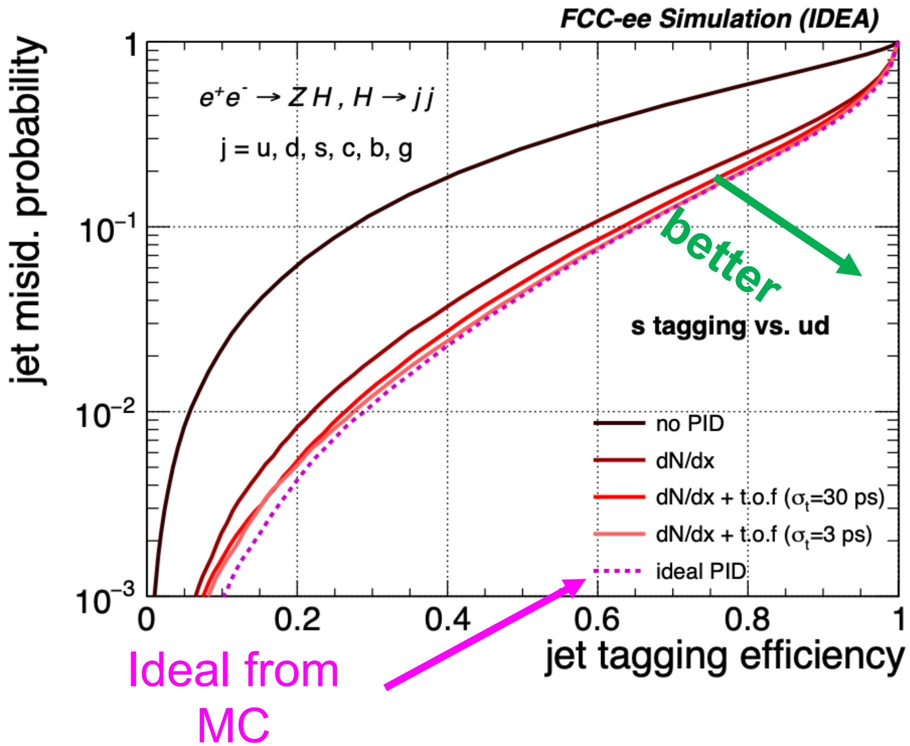


WP	Eff (s)	Mistag (g)	Mistag (ud)	Mistag (c)	Mistag (b)
Loose	90%	20%	40%	10%	1%
Medium	80%	9%	20%	6%	0.4%

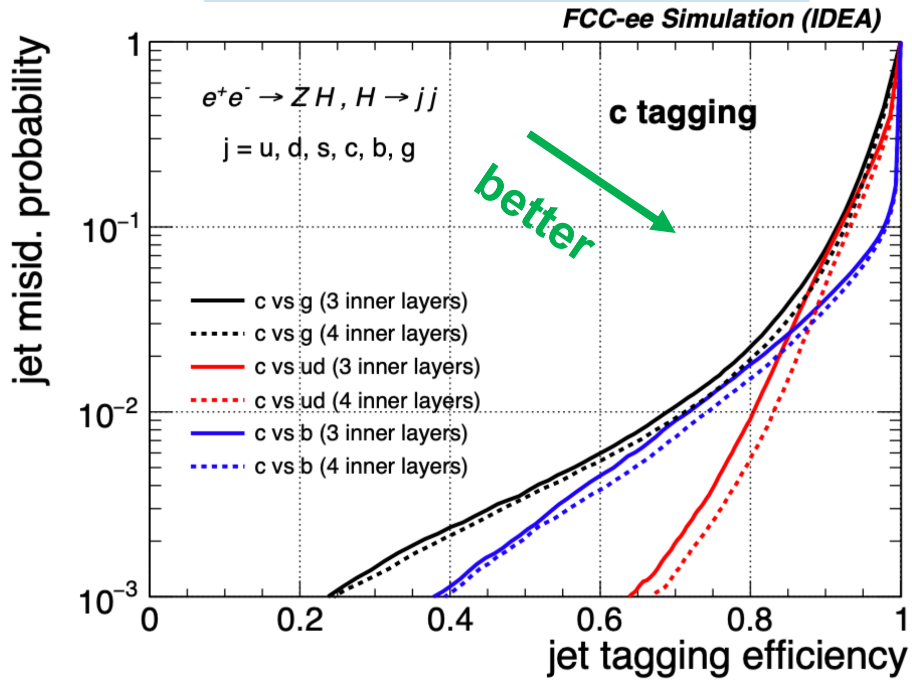
WP	Eff (g)	Mistag (ud)	Mistag (c)	Mistag (b)
Loose	90%	25%	7%	2.5%
Medium	80%	15%	5%	2%

Impact of detector configurations

Strange tagging [PID]

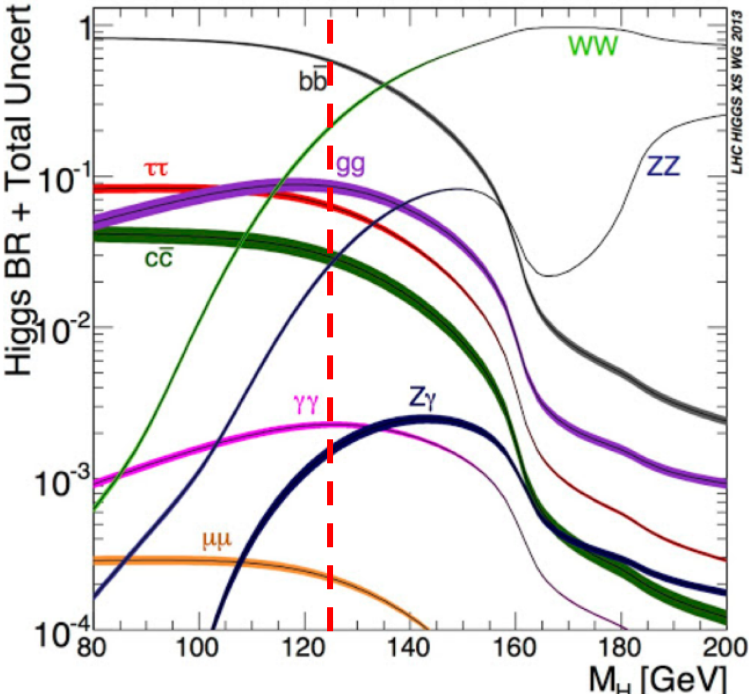


c-tagging [PIX layers]



- dN/dX brings most of the gain; additional gain w/ TOF (30ps)
  - TOF (3ps) brings marginal improvement
  - dN/dX+TOF(30ps): very close to perfect PID
- Improvement up to 2x for charm tagging
  - marginal/no improvement in b-tagging

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	
$H \to b\bar{b}$	±0.3	±3.1	±0.5	±0.9
$H \to c\bar{c}$	±2.2		±6.5	±10
H → gg	±1.9		±3.5	±4.5
$H \to W^+W^-$	±1.2		±2.6	±3.0
H → ZZ	±4.4		±12	±10
$H \to \tau\tau$	±0.9		±1.8	±8
$H \to \gamma\gamma$	±9.0		±18	±22
$H \to \mu^+\mu^-$	±19		±40	
H → invis.	< 0.3		< 0.6	

Ref: P. Janot 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: 4%, g-mistag: 7%

- **Scenario 1:** Z(→vv)H
  δ(σxBR)/σxBR (%) ~ 1.5 [no systematics]
- **Scenario 2:** 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: $H\rightarrow ss$

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

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

Use Tight WP:
[**s-tag: 60%**, **g-mistag, c-mistagm and b-mist**: negligible

- The most challenging BKG is ZZ
with one Z off-shell ~125 GeV [~10% of the Higgs signal]

- Optimistic assumption:
- 100% of the Higgs events (i.e. the 1M events above) are reconstructed
- 100k ZZ events; (BR for Z→ssbar) ~15%
- 15k ZZ events. After applying the Tight WP of the tagger:
- 5.4k events → 88/sqrt(5400) = 1.2σ

Rough numbers from FCC Workshop 2022 [slides]

Back-of-the envelope estimates

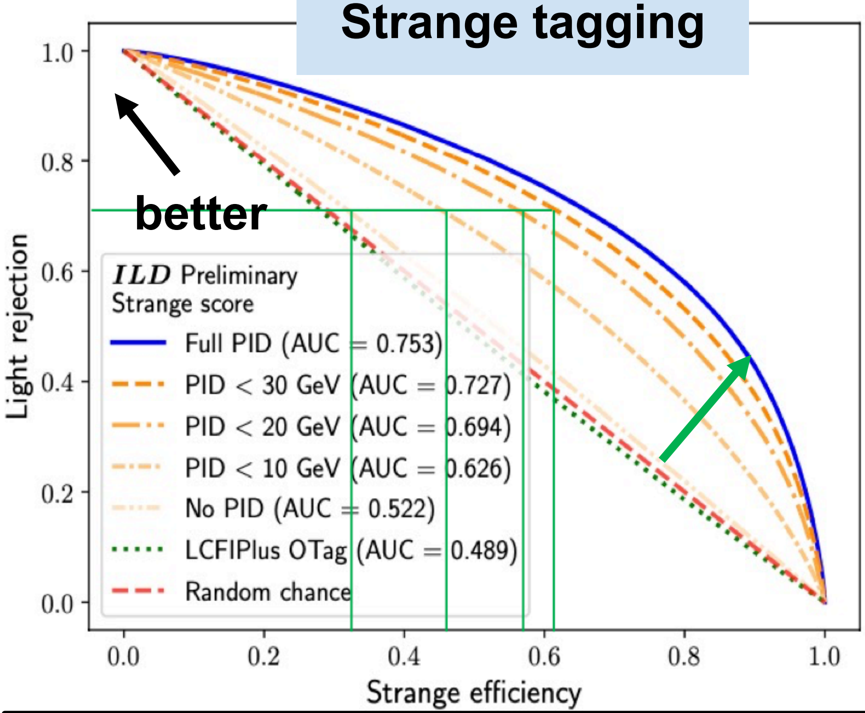
THOROUGH STUDIES NEEDED

Alternative approaches

2203.07535

- Jet tagging using Recurrent Neural Net (RNN)
  - **Inputs:** jet-level info + particle-level info [10 highest-p_T particles]
  - **Multiclass output:** b, c, s, ud, gluon
  - Designed for **ILD**; uses **FullSim**

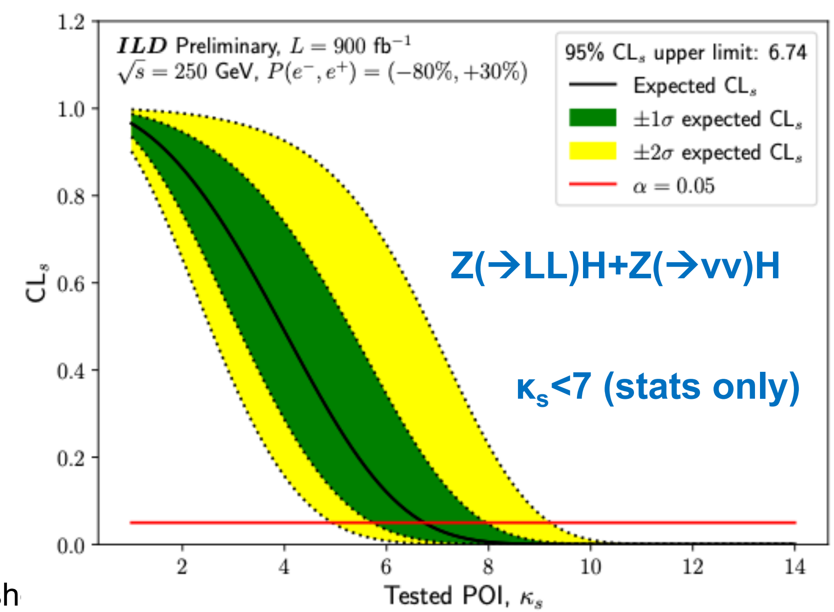
Strange tagging



- Significant improvement compared to previous ILD tagging algorithm (LCFIPlus)
- PID capabilities up to p_T~30 GeV important

Application on H→ss

- ILC @ 250 GeV; 900 fb^{-1}
- Signal: Z(→vv)H and Z(→LL)H
- Analysis design: selection on evt-level vars
- Signal extraction: fit strange-tagging discriminant

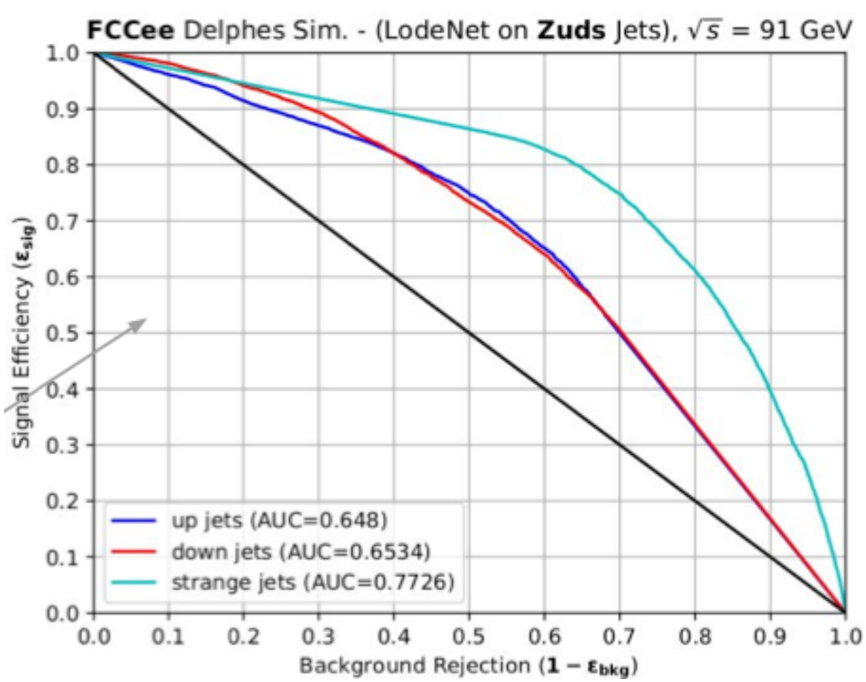


Alternative approaches (II)

[Talk @ FCC Physics Perf. meeting]

- Jet tagging using CNN-2D [focusing on strange-tagging]
  - **Inputs:** jet images; several channels: K, π, e, μ, γ …
  - **Multiclass output:** s, u, d
  - **IDEA** detector; use **FastSim**
  - **Bonus:** Improved Jet flavor assignment

Strange tagging



Impact of PID

Signal Efficiency	10% fake rate	5% fake rate	1% fake rate
Generator	47.2%	27.7%	7.5%
PF only	17.7%	9.7%	2.0%
PF + Ks	21.9%	12.9%	4.4%
PF + Ks + K+-	39.5%	24.8%	7.0%

- Up to 2x improved ε(SIG) with K_S / $K^{+/-}$ separation

Summary & outlook

- Powerful jet flavour identification essential for the success of the e^+e^- physics program

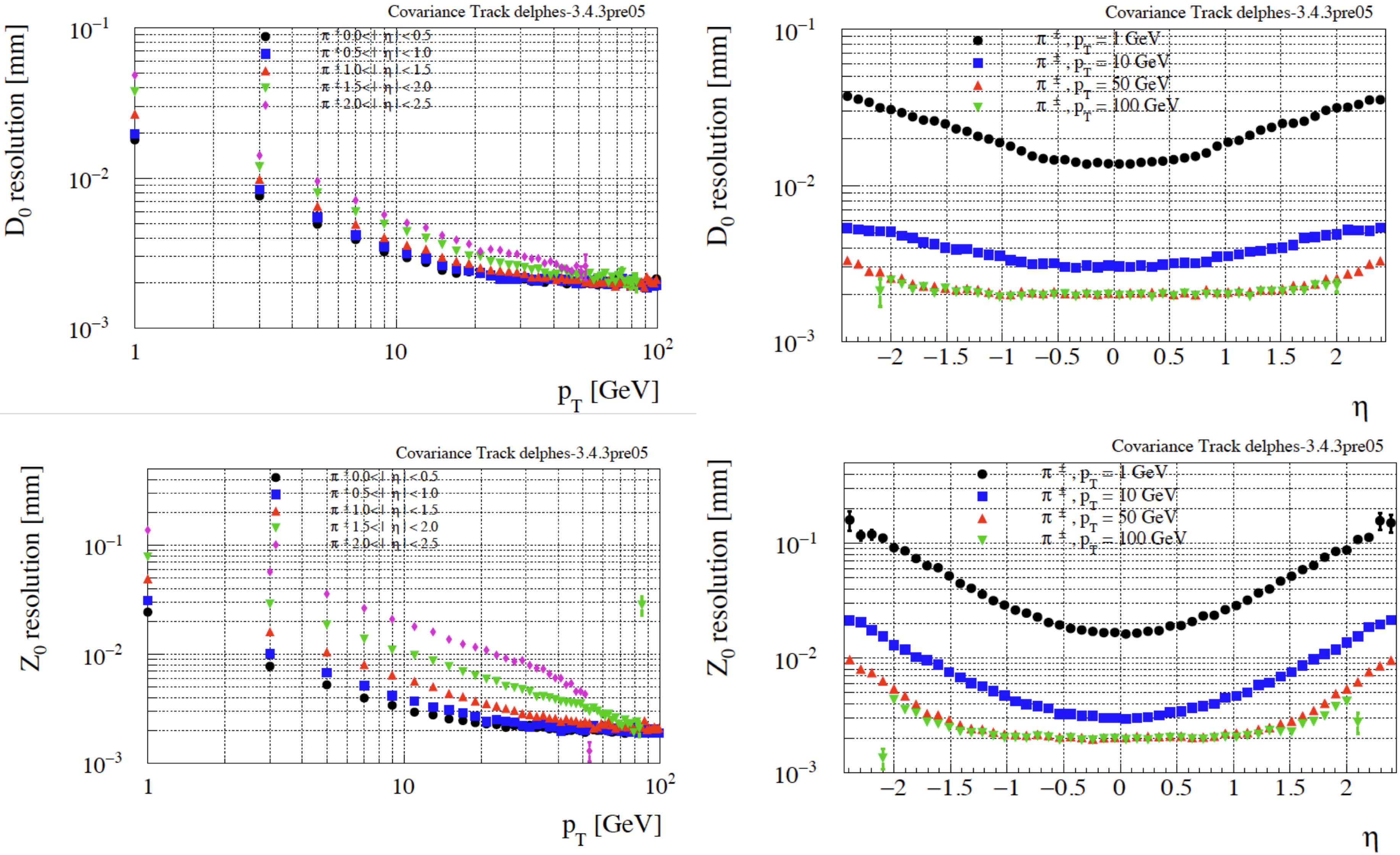
- A first version of a jet identification algorithm based on **PF candidates** and **PID** and **advanced ML** in place
  - Multi-class classifier b/c/s/ud/g; Results extremely promising
  - More details: arXiv:2202.03285
- Conclusions:
  - adding an additional vertex layer does not tremendously improve b-tagging performance (resolution of ~ 2um already outstanding)
    - but improves charm tagging
  - Some room for improved in strange tagging with more powerful PID
- Next steps/work in progress
  - Implementation in FCCSW
  - Test performance using FullSim
  - Application on physics analyses [e.g., H→cc, H→ss]

Backup

Impact parameter performance

Credits to Sylvie Braibant

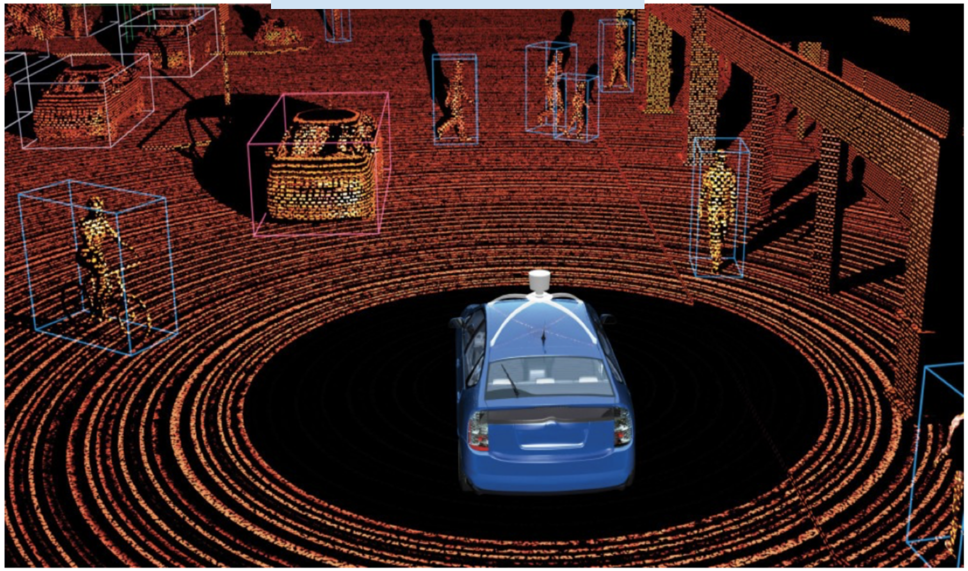
IDEA detector:



2μm IP resolution at high-p_T

Designing a jet flavour tagging algorithm

A point cloud

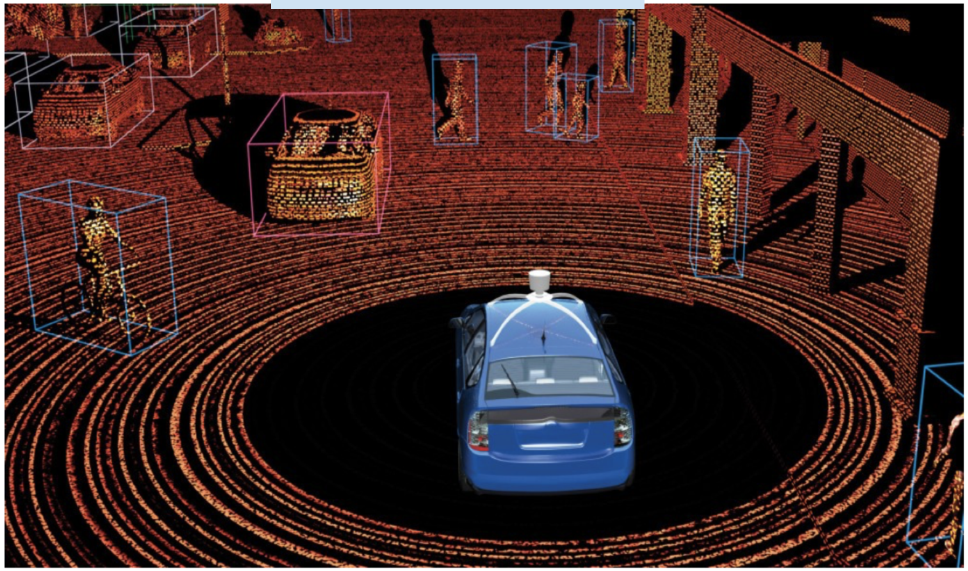


Source:https://news.voyage.auto/an-introduction-to-lidar-the-key-self-driving-car-sensor-a7e405590cff

- Point cloud (Wikipedia):
  - A set of data **points** in space
  - Produced by 3D scanners, which measure a large number of points on the external surfaces of objects around them

From point clouds to particle clouds

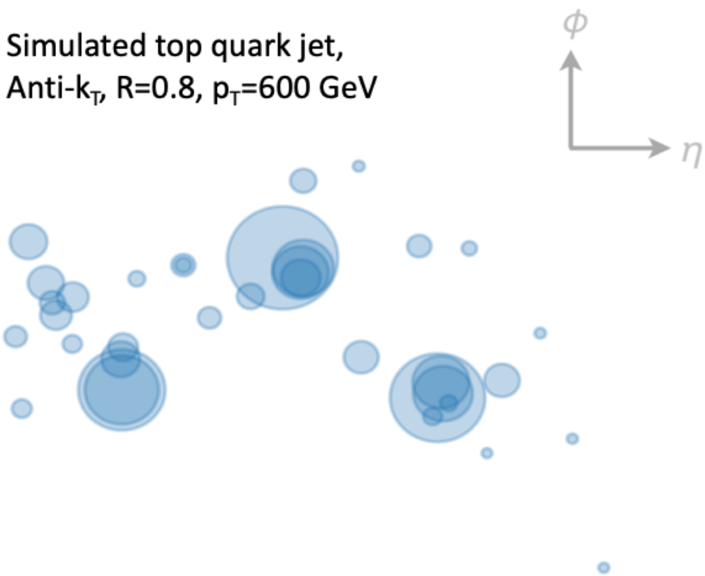
A point cloud



Source:https://news.voyage.auto/an-introduction-to-lidar-the-key-self-driving-car-sensor-a7e405590cff

A particle cloud

Simulated top quark jet,
Anti-k_T, R=0.8, p_T=600 GeV



- Point cloud (Wikipedia):
  - A set of data **points** in space
  - Produced by 3D scanners, which measure a large number of points on the external surfaces of objects around them
- Particle cloud :
  - A set of **particles** in space
  - Produced by clustering a large number of particles measured by the detectors

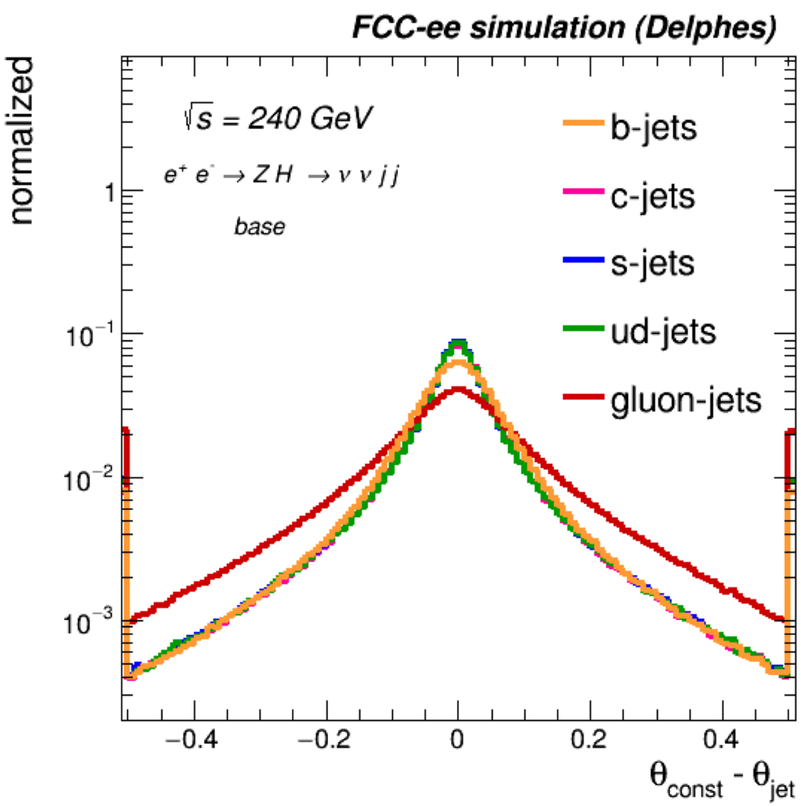
Full list of input variables

Variable	Description
Kinematics	
$E_{\mathrm{const}}/E_{\mathrm{jet}}$	energy of the jet constituent divided by the jet energy
θ_{rel}	polar angle of the constituent with respect to the jet momentum
ϕ_{rel}	azimuthal angle of the constituent with respect to the jet momentum
Displacement	
d_{xy}	transverse impact parameter of the track
d_z	longitudinal impact parameter of the track
$\mathrm{SIP}_{\mathrm{2D}}$	signed 2D impact parameter of the track
$\mathrm{SIP}_{\mathrm{2D}}/\sigma_{\mathrm{2D}}$	signed 2D impact parameter significance of the track
$\mathrm{SIP}_{\mathrm{3D}}$	signed 3D impact parameter of the track
$\mathrm{SIP}_{\mathrm{3D}}/\sigma_{\mathrm{3D}}$	signed 3D impact parameter significance of the track
d_{3D}	jet track distance at their point of closest approach
$d_{\mathrm{3D}}/\sigma_{d_{\mathrm{3D}}}$	jet track distance significance at their point of closest approach
C_{ij}	covariance matrix of the track parameters
Identification	
q	electric charge of the particle
$m_{\mathrm{t.o.f.}}$	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

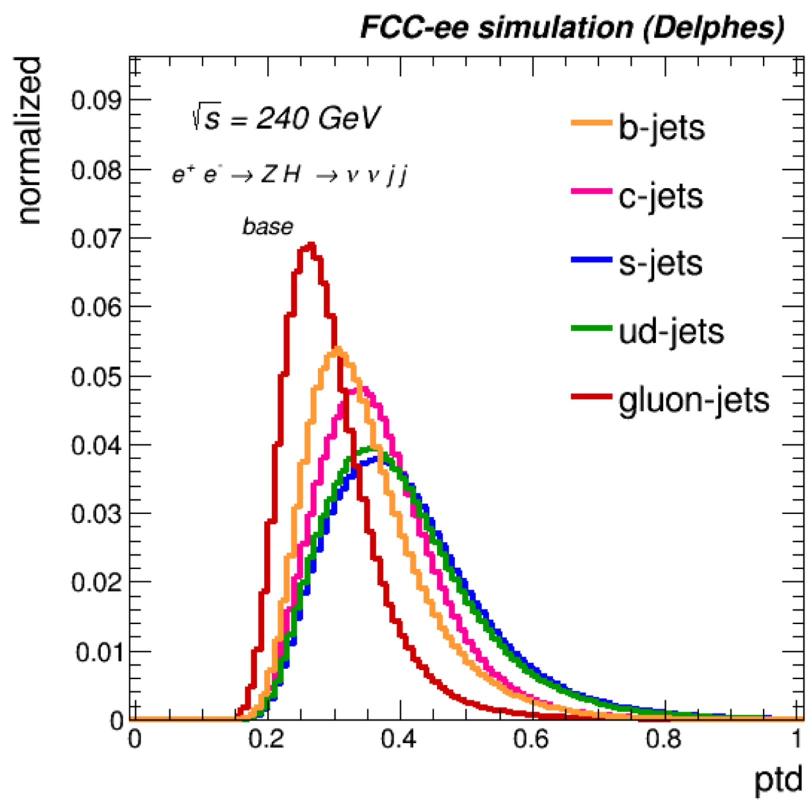
Input variables

- Comparison of input distributions for different jet flavors

Projection || to jet axis

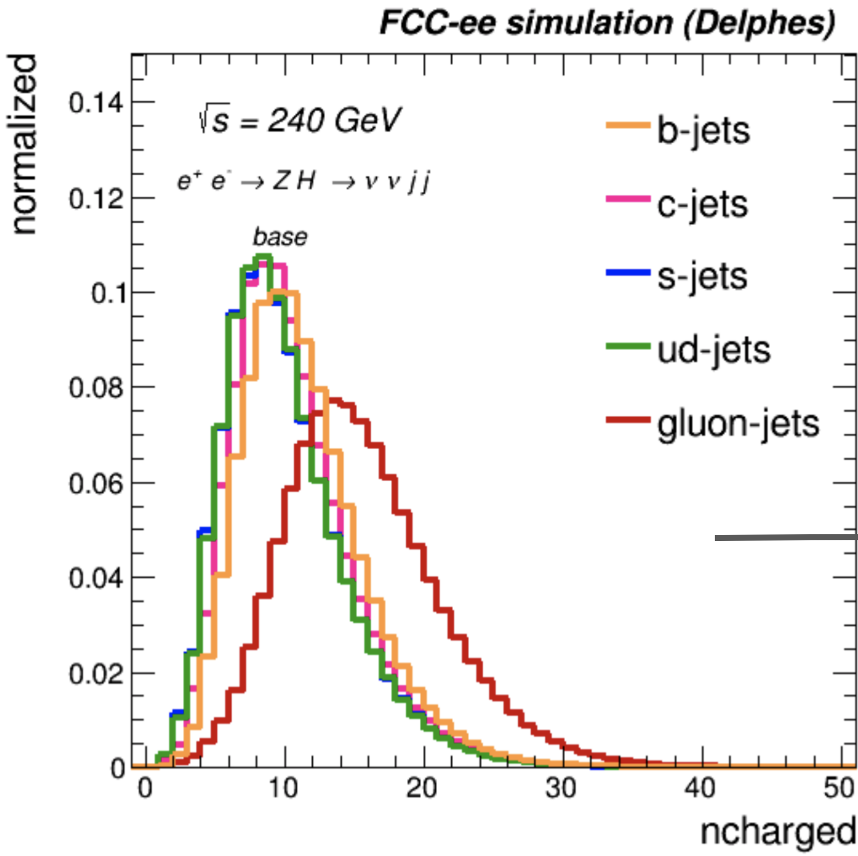


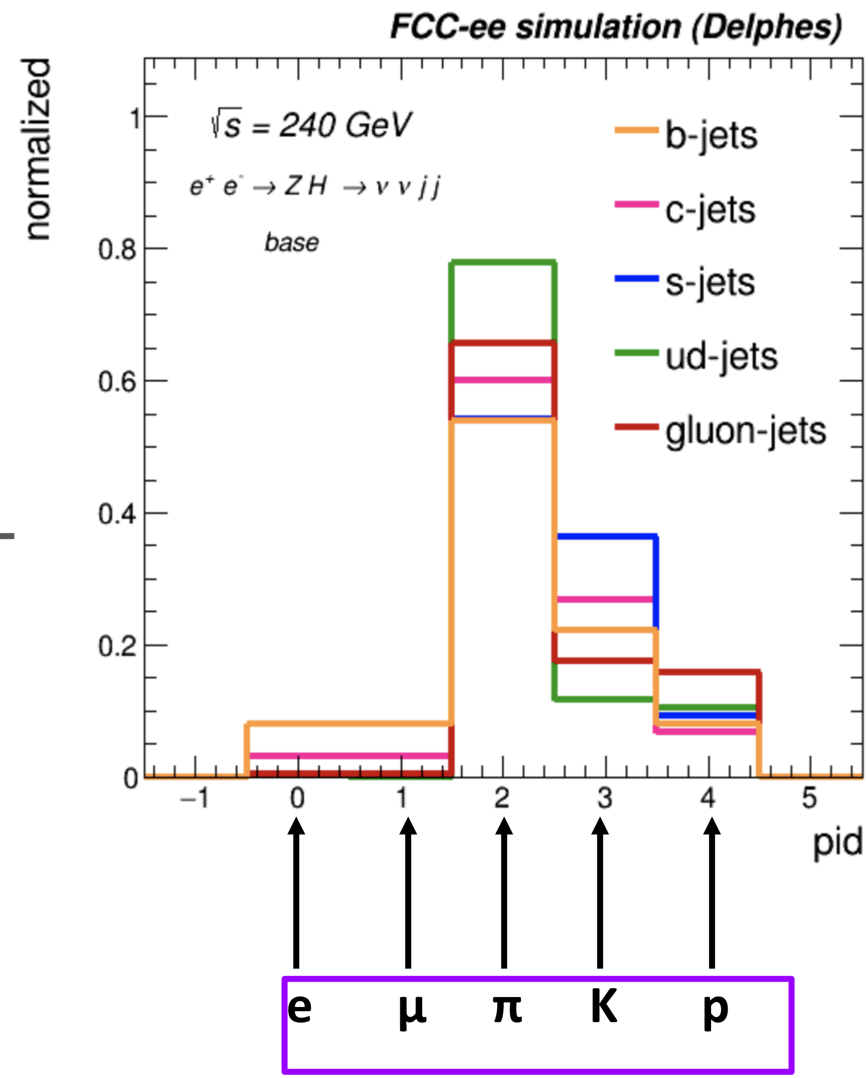
$p_T D$



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

Performance w/ PID





no PID: only charge

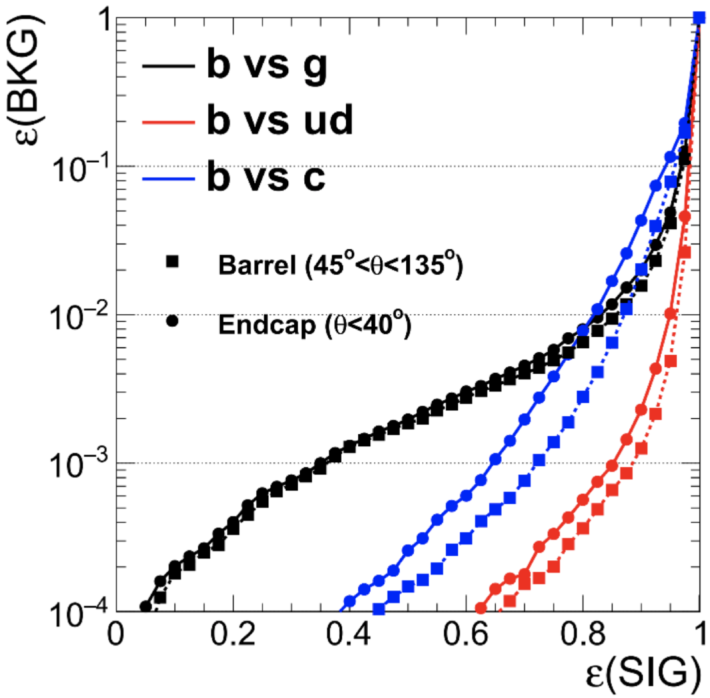
realistic: e, μ, mtof, dNdx

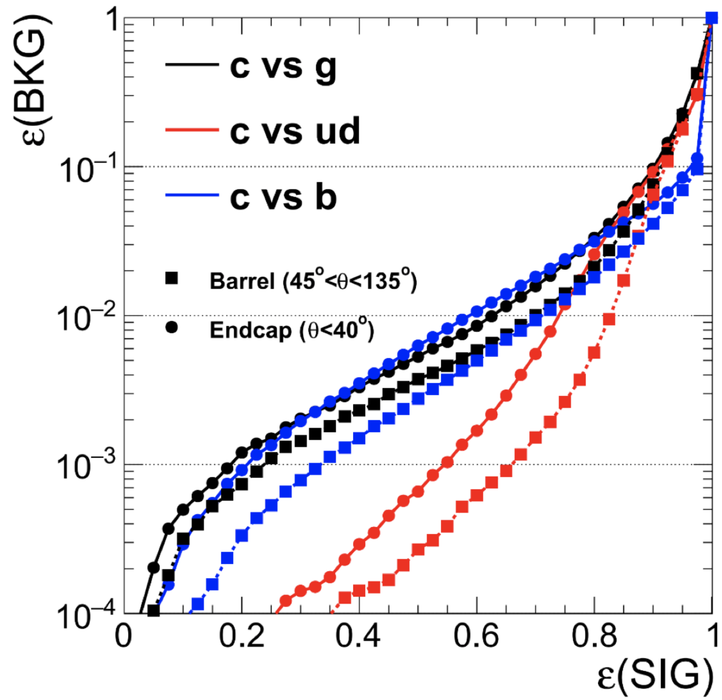
perfect PID: e, μ, + π, K, p from MC truth

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