QCD & Lund Jet Plane studies at FCC-ee

L. Panwar (Postdoc), L. Delagrange (PhD),

R.C. Camacho Toro, B. Malaescu, L. Poggioli

contact email: lata.panwar@cern.ch

Jamboree FCC, 2nd July 2024

Introduction and motivation

- **Analysis:** Prospects of **QCD studies related to jet formation** and **jet tagging** methods at FCC-ee using **Lund Jet Plane (LJP) representation**
  - Aim to study the **sensitivity to α_s at FCC-ee**, probing of α_s for different energies and tests of the renormalization group equation (RGE) in QCD; complementary to the 3/2 Jet cross-section ratio study for α_s
  - Aim to study the potential **use of LJP for improving jet tagging**

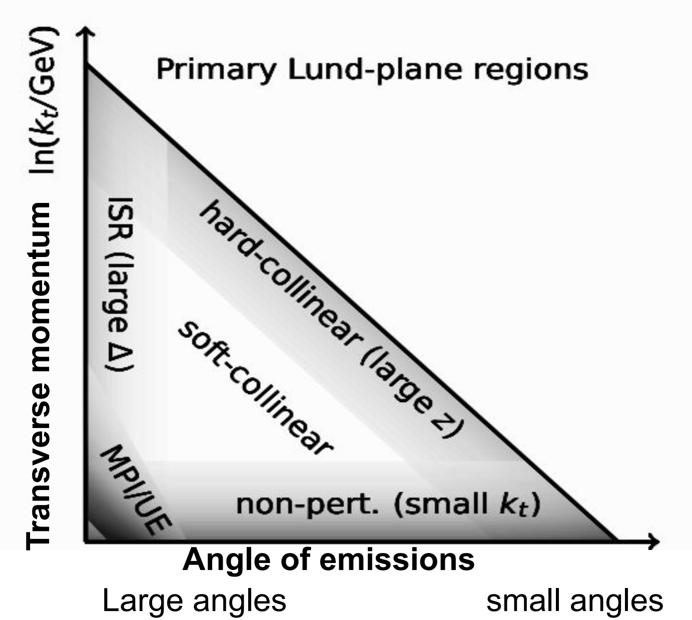
- **Why at FCC-ee?**
  - provides a clean collision environment with high statistics for precise measurements
  - potential impact of this study for the optimization of detector parameters
- Samples: Centrally produced Winter2023 Delphes samples for IDEA detector

Introduction and motivation

Benefits of Lund Jet Plane method (LJP):

- QCD jet formation involves perturbative and non-perturbative effects; presence of these effects impact the precision of any measurement based on jets
- LJP works as a handle to separate these effects in a 2D representation using angle (ΔR) and transverse momentum (k_t) of emissions within the jets and further opens a possibility to understand QCD behaviour separately for these perturbative and non-perturbative effects

(arXiv:1807.04758)



How to extract α_s ?

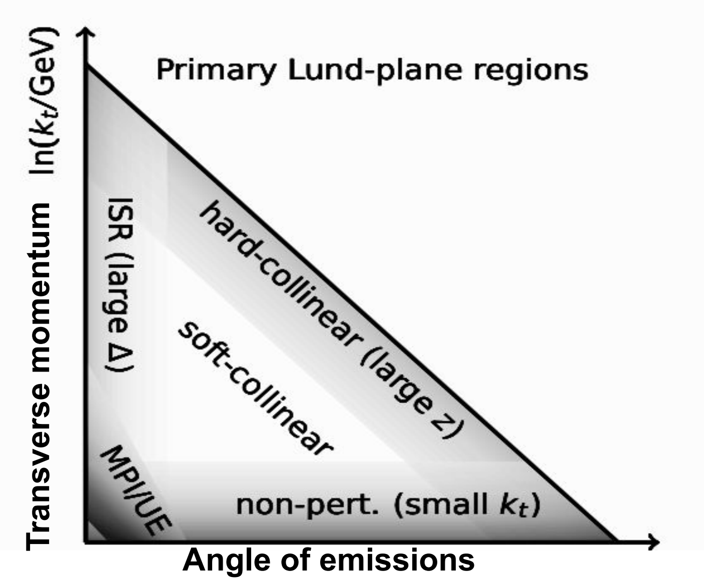
- QCD process behind jet formation is related to strong coupling constant $\boldsymbol{\alpha}_s$
  - Running constant which varies with different energies
  - Impacts both jet shape (distribution of emissions inside jet) and normalization

Average density of emissions in LJP can be given as:

$$\rho(k_T, \Delta R) \equiv \frac{1}{N_{\mathrm{jets}}} \frac{\mathrm{d}^2 N_{\mathrm{emissions}}}{\mathrm{d}\ln(k_T/\,\mathrm{GeV})\,\mathrm{d}\ln(R/\Delta R)} \approx \frac{2}{\pi} C_R \alpha_S(k_T)$$

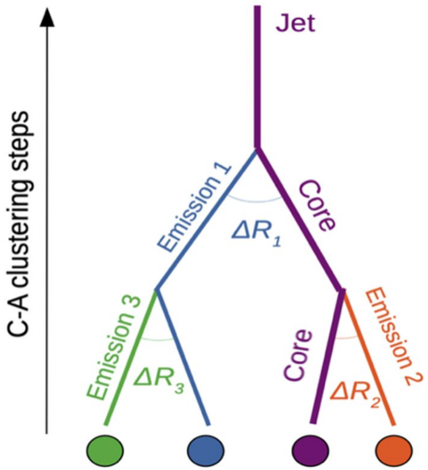
Where C_R = color factor

(arXiv:1807.04758)



Large angles small angles

How to build Lund Jet Plane?



- Start with a jet and cluster it again to have angular order information of emissions (*JHEP 12 (2018) 064*)
- Decluster them in reverse (start with wide angle emission first)
- Within the iterative declustering, harder branch is always taken as core branch
- Fill a triangle plane of two Lund variables (k_t and ΔR) from core and emission

NOTE:

- Angular ordered Cambridge/Aachen (C/A) declustering (following the theoretical proposal) depends on ΔR in (**y**, ϕ) plane used for LHC studies (given in backup)
- It is more accurate to perform ΔR-based declustering in the (**θ**, ϕ) plane for FCC-ee.

For “a” core and “b” emission branch

$$k_t \equiv p_{tb} \Delta R_{ab}$$

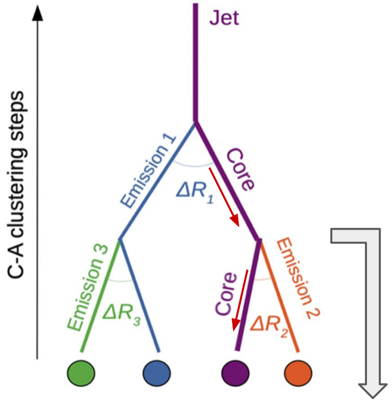
$$z \equiv p_{tb} / (p_{ta} + p_{tb})$$

ΔR_{ab} = angle of emission **b** wrt to core **a**

k_t = transverse momentum of **b** wrt **a**

z = momentum fraction taken by **b**

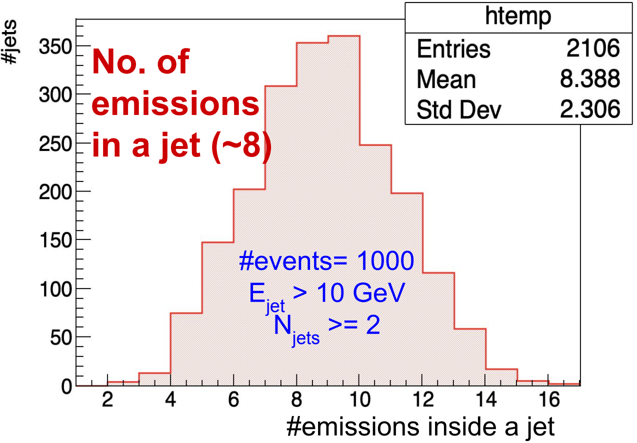
How to build Lund Jet Plane?



For R=1.5 jets clustered with k_t algorithm (Kt15)

ee → Z→ uu/dd @91 GeV

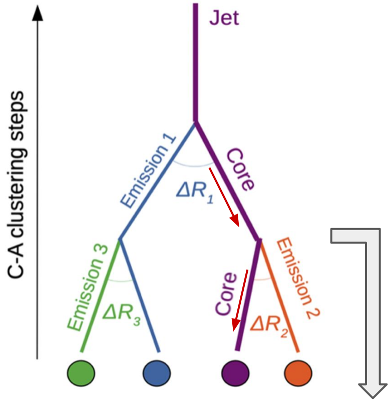
Emissions from the core branches



No. of emissions in a jet (~8)

#events= 1000
E_{jet} > 10 GeV
N_{jets} >= 2

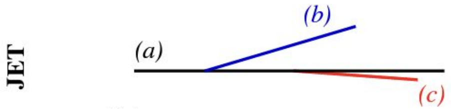
How to build Lund Jet Plane?

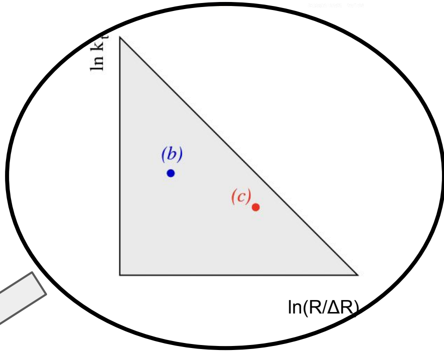


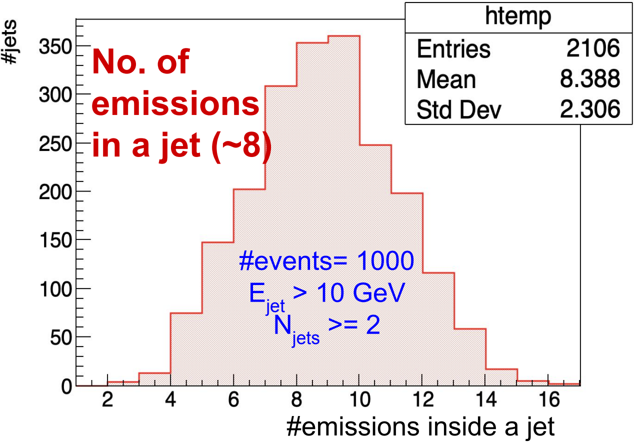
Emissions from the core branches

For R=1.5 jets clustered with k_t algorithm (Kt15)

ee → Z→ uu/dd @91 GeV

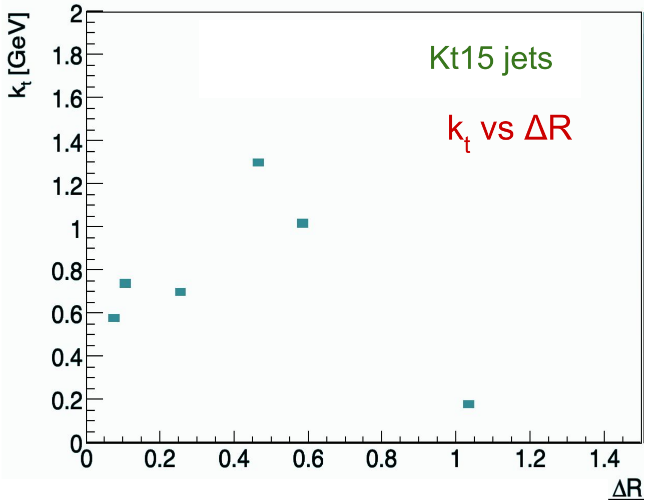




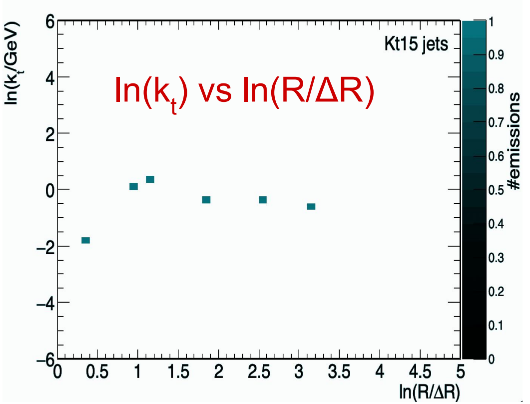


No. of emissions in a jet (~8)

#events= 1000
E_{jet} > 10 GeV
N_{jets} >= 2



k_t vs ΔR



$\ln(k_t)$ vs ln(R/ΔR)

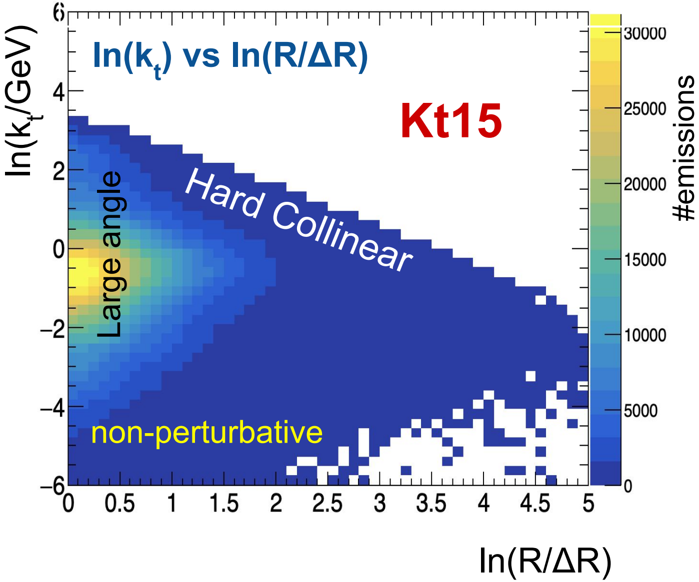
LJP representation for 1 jet of E_{jet} ~ 40 GeV

(both plots represent the same jet w/ and w/o log scale)

Recap from Annecy workshop

- Presented preliminary LJP representation for jets clustered with different jet algorithms
- First look at the LJP for possibility of jet tagging for b-jets and light quark jets (and gluon jets) at FCCee

ee→Z→ uu/dd @ √s = 91 GeV



- E_{jet} > 10 GeV and N_{jets} >= 2; selection efficiency > 99%
- LJP representations for two leading p_T jets

slides

Follow up from workshop

From the workshop we had some new ideas for analysis

1. For α_s scan, use **Madgraph** (**MG@NLO)** to generate samples

   (Need to move from Pythia (LO generator) to NLO generator)

2. Possibility of using **ee dedicated jet clustering/declustering algorithm**

Follow up from workshop

From the workshop we had some new ideas for analysis

1. For α_s scan, use **Madgraph (MG@NLO)** to generate samples

   (Need to move from Pythia (LO generator) to NLO generator)

   a. How to vary α_s ?

   b. Validation of α_s effect at generator level and after Delphes simulation

2. Possibility of using **ee dedicated jet clustering/declustering algorithm**

Sample generation (Madgraph+Pythia+Delphes)

- Presentation from G. Stagnitto for usage of MG@NLO for FCCee-dedicated studies
- Appropriate for α_s studies since generates events at NLO level
- **Technical challenge:** how to generate samples with different α_s values
  - Got help from experts (thanks to M. Selvaggi & D. Enterria) to understand the α_s propagation in event generation
  - **Generate LHE events at Z-pole $\sqrt{s}$ = 91.118 GeV** and validate them
  - Generated level LHE events are further simulated using Pythia+Delphes
    - Study ongoing
  - More details are in backup

Validation studies-LHE level

define q = u d u~ d~

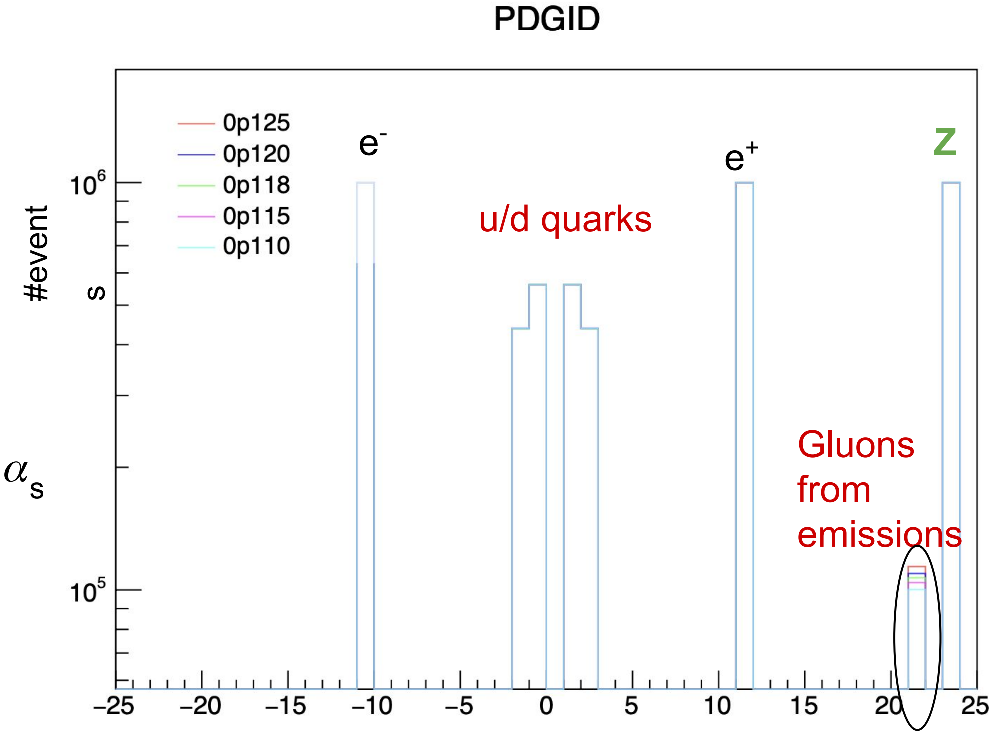
generate e+ e- > **Z** > **q q** [QCD]

α_s= [0.110, 0.115, **0.118**, 0.120, 0.125]

world average for α_s

#events = 1 M/sample

emitted gluons multiplicity increases with α_s



Validation studies-LHE level

define q = u d u~ d~

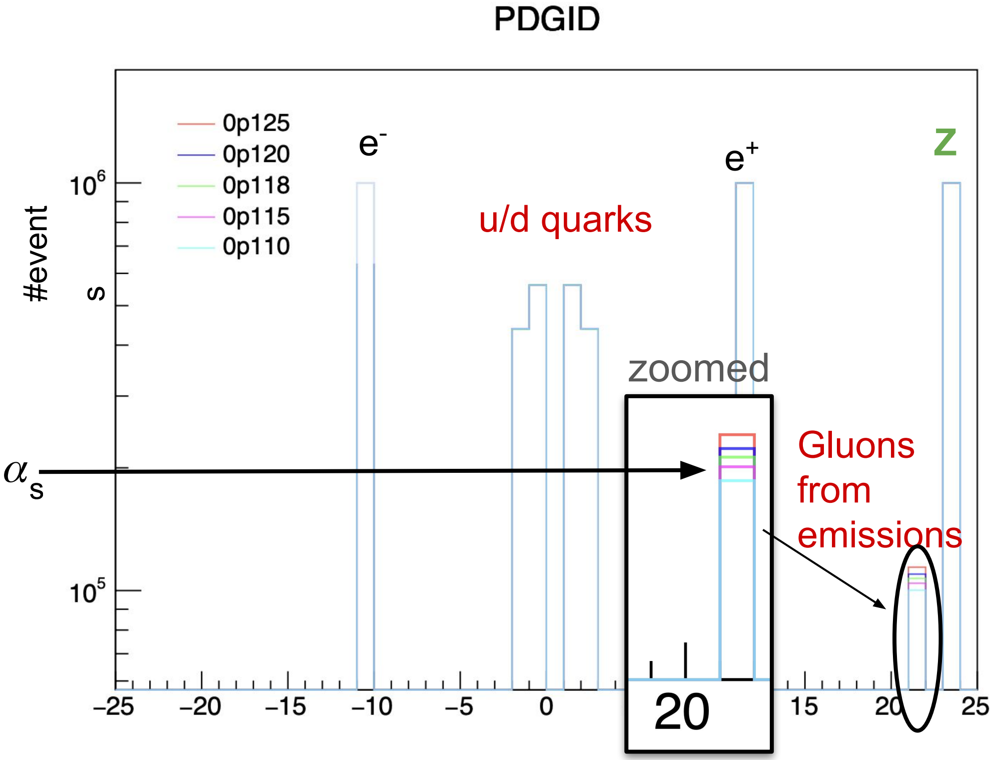
generate e+ e- > **Z** > **q q** [QCD]

α_s= [0.110, 0.115, **0.118**, 0.120, 0.125]

world average for α_s

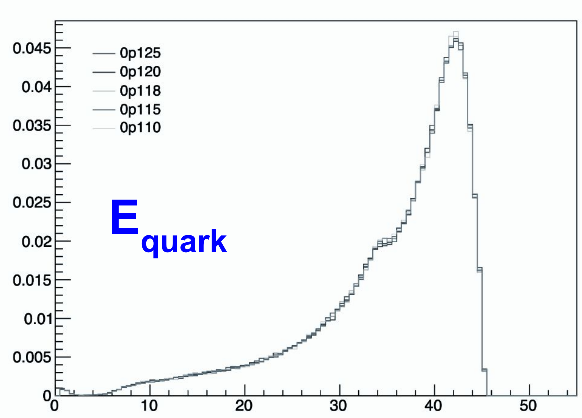
#events = 1 M/sample

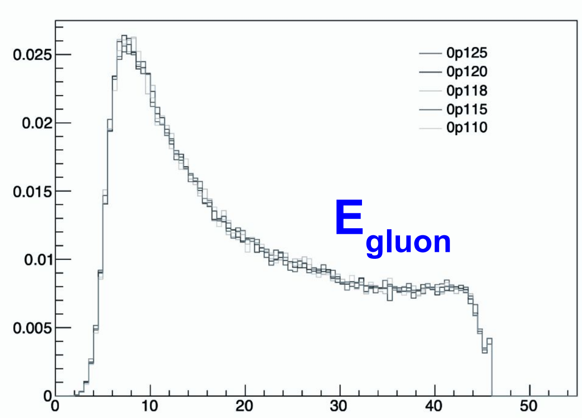
emitted gluons multiplicity increases with α_s

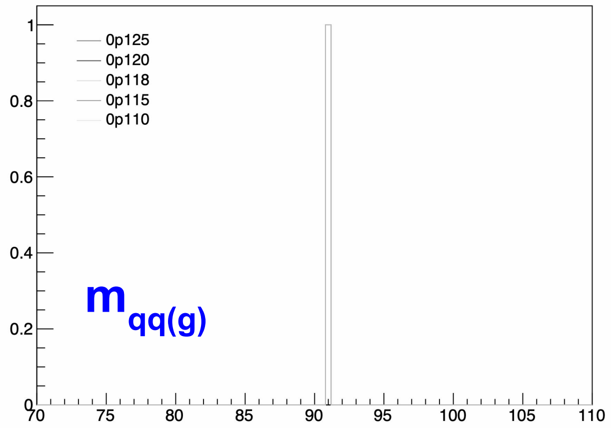


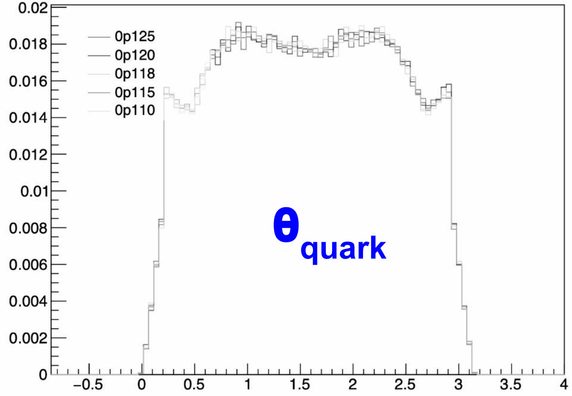
Validation studies:LHE level

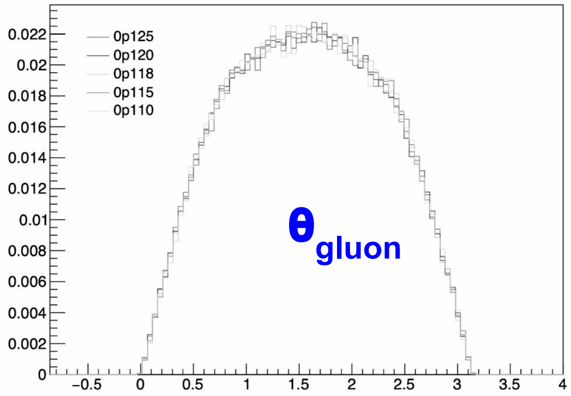
- Distributions are shown for different α_s values and are shape normalized
- No selection at generator level
- Other distributions are in backup











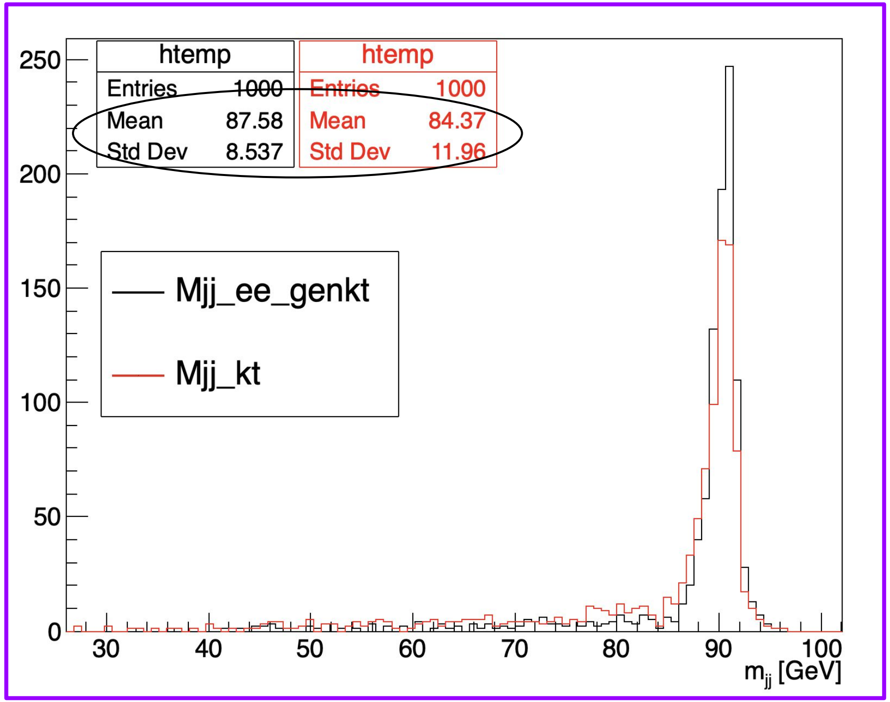
Follow up from workshop

From the workshop we had some new ideas for analysis

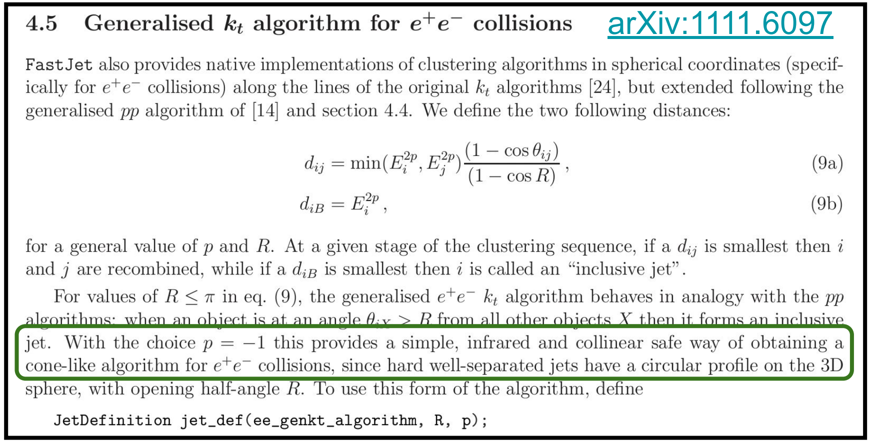
1. For α_s scan, use **MG@NLO** to generate samples

2. Possibility of using **ee dedicated jet clustering/declustering algorithm**
   a. Include ee generalised-kt (ee_gen_kt) algorithm for jet clustering
   b. Following suggestion from G. Salam and A. Karlberg, work on jet declustering using EECambridgePlugin
   (Similar to Cambridge/Aachen (C/A) but does ΔR(**θ**,ϕ)-based declustering)

ee generalised k_t-based jet clustering



Better m_{jj} resolution with θ-based k_t algorithms wrt ΔR(y,ϕ)-based k_t algorithms



4.5 Generalised k_t algorithm for e^+e^- collisions

arXiv:1111.6097

FastJet also provides native implementations of clustering algorithms in spherical coordinates (specifically for e^+e^- collisions) along the lines of the original k_t algorithms [24], but extended following the generalised pp algorithm of [14] and section 4.4. We define the two following distances:

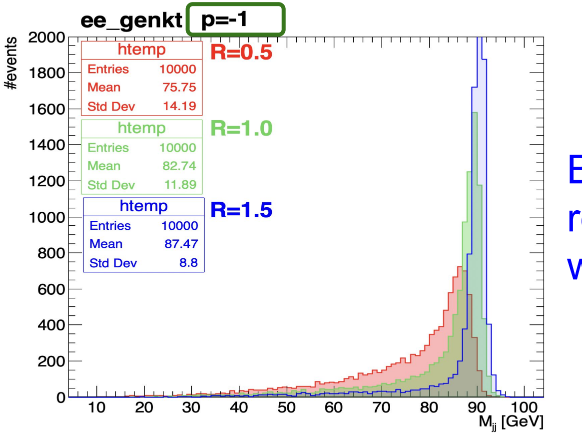
$$d_{ij} = \min(E_i^{2p}, E_j^{2p}) \frac{(1-\cos\theta_{ij})}{(1-\cos R)}, \tag{9a}$$

$$d_{iB} = E_i^{2p}, \tag{9b}$$

for a general value of p and R. At a given stage of the clustering sequence, if a d_{ij} is smallest then i and j are recombined, while if a d_{iB} is smallest then i is called an "inclusive jet".

For values of $R \le \pi$ in eq. (9), the generalised e^+e^- k_t algorithm behaves in analogy with the pp algorithms: when an object is at an angle $\theta_{iX} > R$ from all other objects X then it forms an inclusive jet. With the choice $p = -1$ this provides a simple, infrared and collinear safe way of obtaining a cone-like algorithm for e^+e^- collisions, since hard well-separated jets have a circular profile on the 3D sphere, with opening half-angle R. To use this form of the algorithm, define

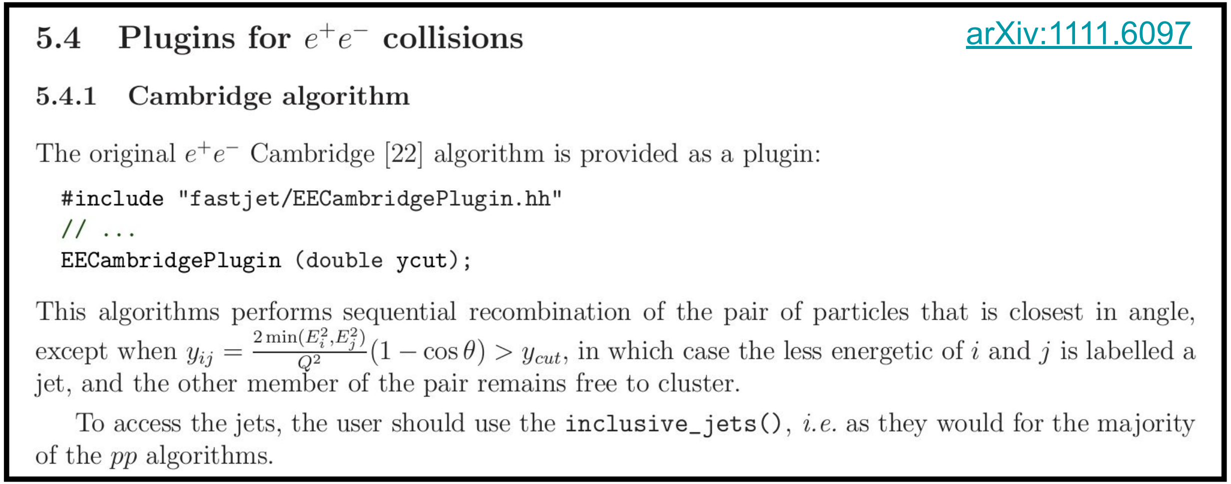
```
JetDefinition jet_def(ee_genkt_algorithm, R, p);
```



Better m_z reconstructions with R=1.5

Angular order-based jet declustering in (θ, ϕ) plane

- Use ee-dedicated Cambridge algorithm (**EECambridgePlugin**); Implemented in code with help from fastjet experts (link)
- Setup is in place; only tested for few **ee→ Z→ light jets** events



5.4 Plugins for e^+e^- collisions

arXiv:1111.6097

5.4.1 Cambridge algorithm

The original e^+e^- Cambridge [22] algorithm is provided as a plugin:

```
#include "fastjet/EECambridgePlugin.hh"
// ...
EECambridgePlugin (double ycut);
```

This algorithms performs sequential recombination of the pair of particles that is closest in angle, except when $y_{ij} = \frac{2\min(E_i^2, E_j^2)}{Q^2}(1 - \cos\theta) > y_{cut}$, in which case the less energetic of i and j is labelled a jet, and the other member of the pair remains free to cluster.

To access the jets, the user should use the `inclusive_jets()`, *i.e.* as they would for the majority of the *pp* algorithms.

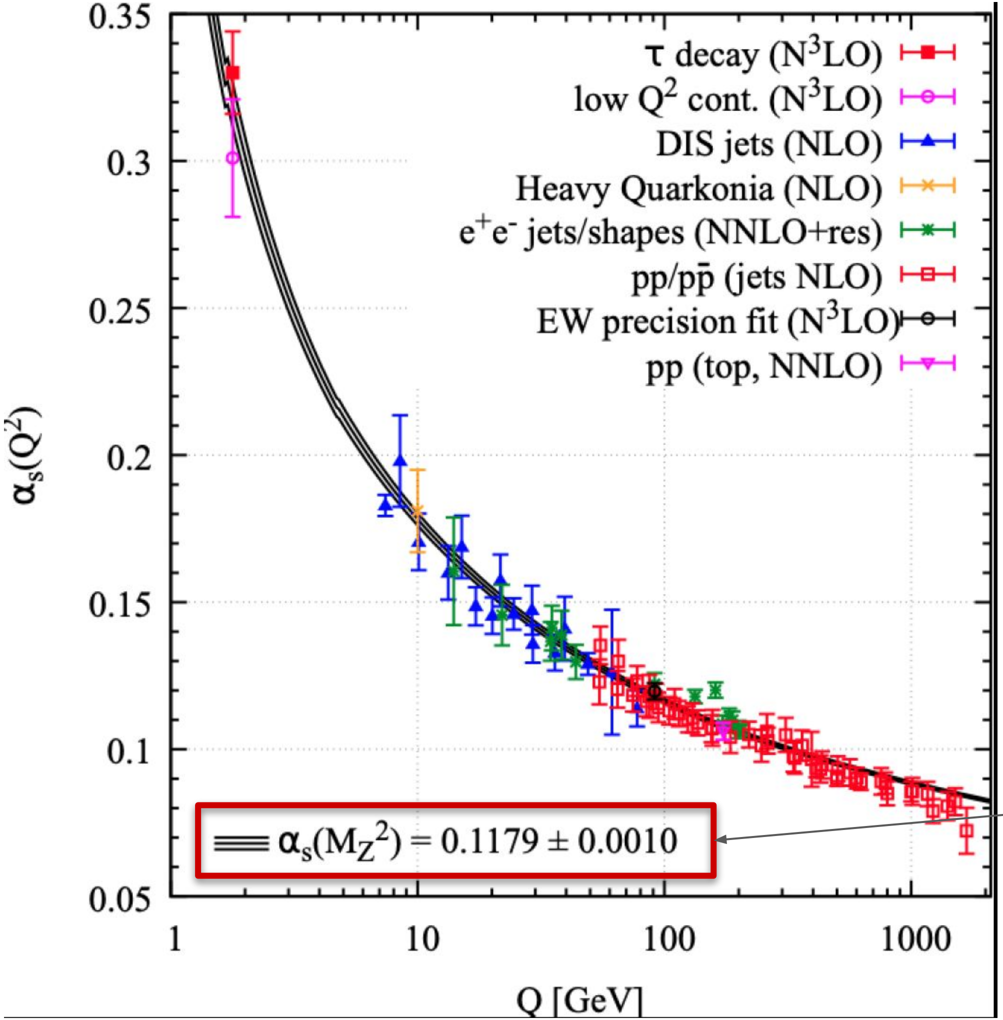
Note: Study ongoing (more in backup)

Summary and next steps

- To our knowledge it is first study which looks at jet substructure at FCC-ee
  - Motivated by the study of the sensitivity to α_s and test of RGE
  - Motivated by studying the jet tagging using LJP (For tagging heavy flavour and quark/gluon jets)
  - Plan to explore the sensitivity of the reconstructed LJP to:
    - α_s by doing α_s-scan; (explore both Primary and Secondary LJP)
    - Optimization of the detector parameters
- Present updates from feedback received from last FCC workshop in Annecy
  - Validated Madgraph sample generation with different α_s values at generator (LHE) level
  - Study ongoing for α_s impact on reconstructed jets (From Delphes)
  - Also, switched to ee-dedicated jet clustering/declustering algorithms (implemented within code and performed preliminary checks)
- Special thanks to **M. Selvaggi, P. Azzi and E. Perez** for all their support

Thank you

BACKUP



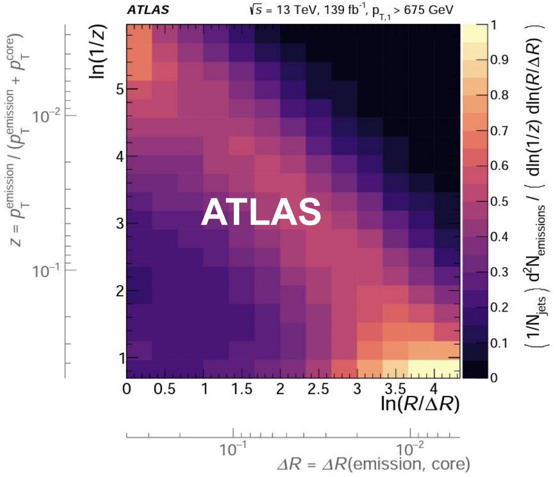
Expect factor of 10 improvement in precision with FCCee

Current precision

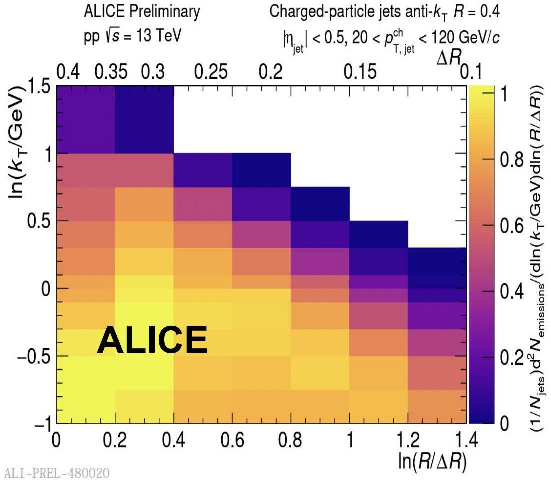
Recent Lund Jet Plane based measurements

- LJP studies at LHC √s = 13 TeV, following recent theoretical proposal (*JHEP 12 (2018) 064*)
- These studies measure the lund plane density for charged particles jets
- We are interested in following the same for FCC-ee environment

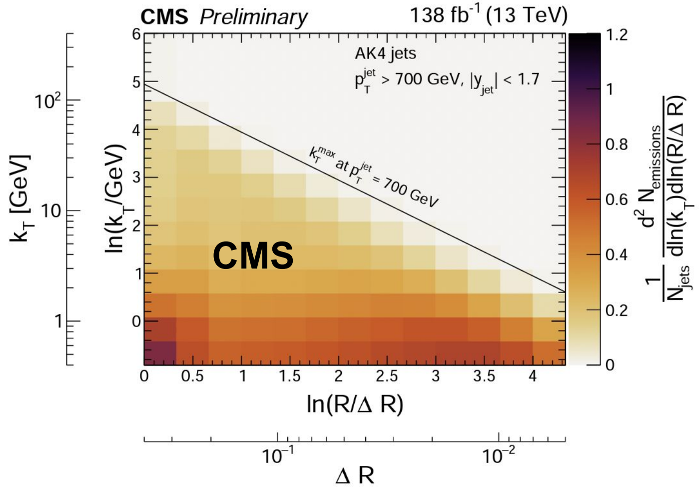
arXiv 2004.03540



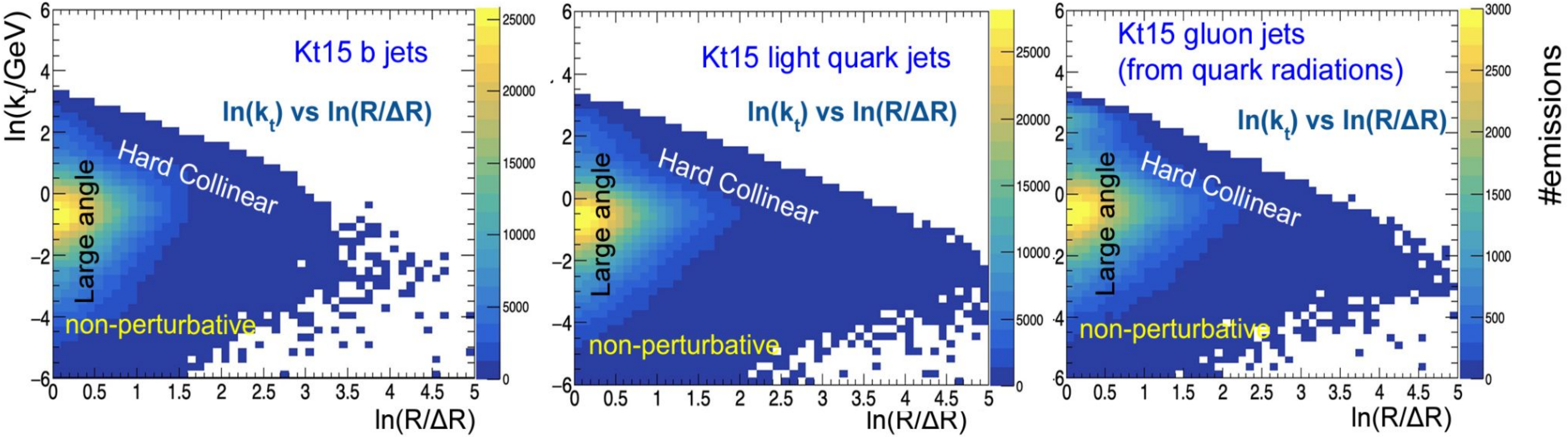
arXiv 2111.00020



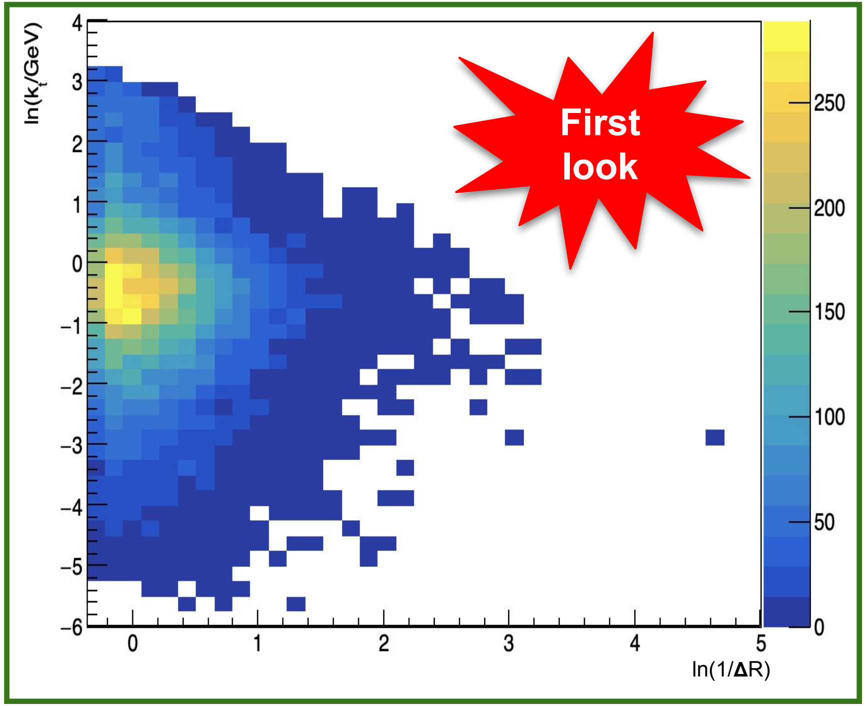
CMS-PAS-SMP-22-007

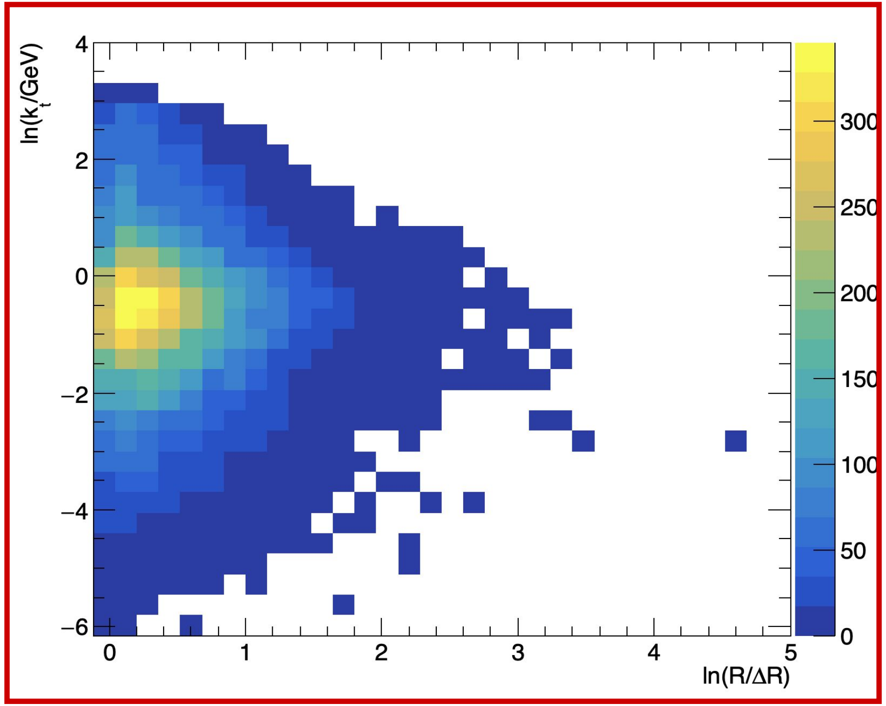


LJP representation for light and heavy flavor jets at higher energy



- For process ee→ Z→bb and ee→Z→uu/dd (+ gluons from quark radiation) @91 GeV ; #events = 1 M
- Selection: E_{jet} > 10 GeV, N_{jets} >= 2; selection efficiency > 99%; use two leading p_T jets
- Difference in the large ln(R/ΔR) region shows that light jets are more collimated than heavy flavour jets; working on further understanding in more detail





$k_t = p_{T,\mathrm{emission}} * \sin(\Delta R(\theta, \phi))$

Declustering with EECambridgePlugin

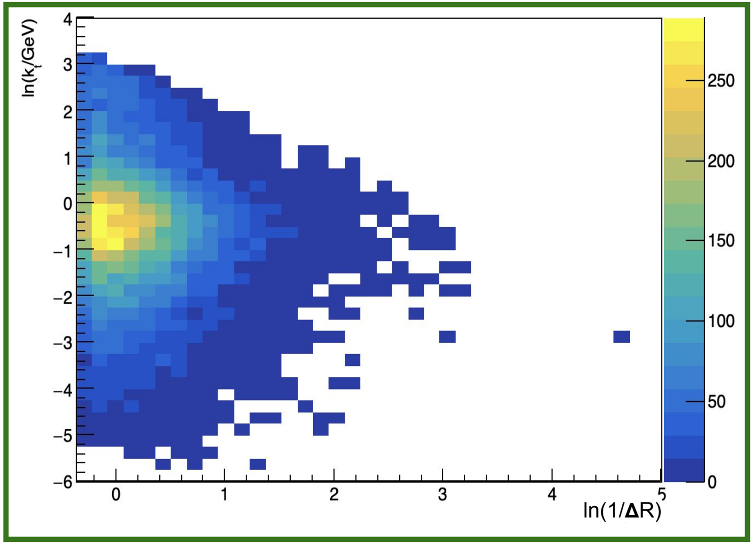
$k_t = p_{T,\mathrm{emission}} * \sin(\Delta R(y, \phi))$

Declustering with C/A

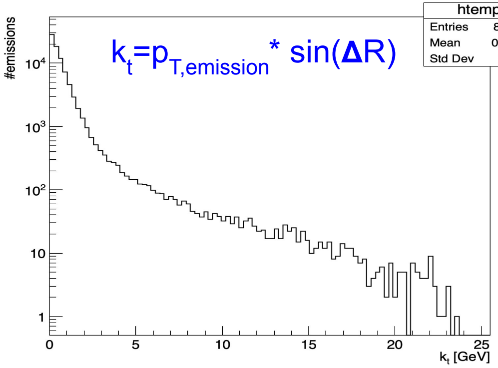
- ee_genkt clustered jets with R=1.5, $E_{jet} > 10$ GeV and #jets > 1 (for leading jets)

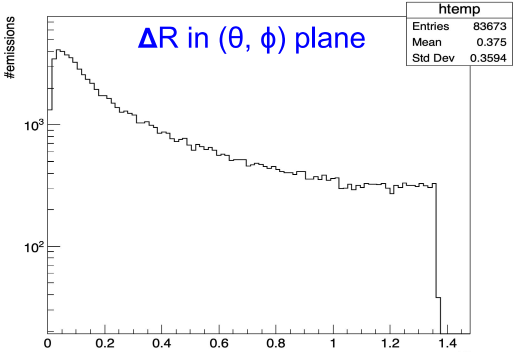
Angular order-based jet declustering in (θ, ϕ) plane

First look



$\ln(k_t)$ vs $\ln(1/\Delta R)$; ΔR in (θ, ϕ) plane





Samples generation with Madgraph

```
./bin/mg5_aMC
define q = u d u~ d~
generate e+ e- > Z > q q [QCD]
display processes
display diagrams
output myDir
launch
```

w/o pythia fragmentation and hardronisation

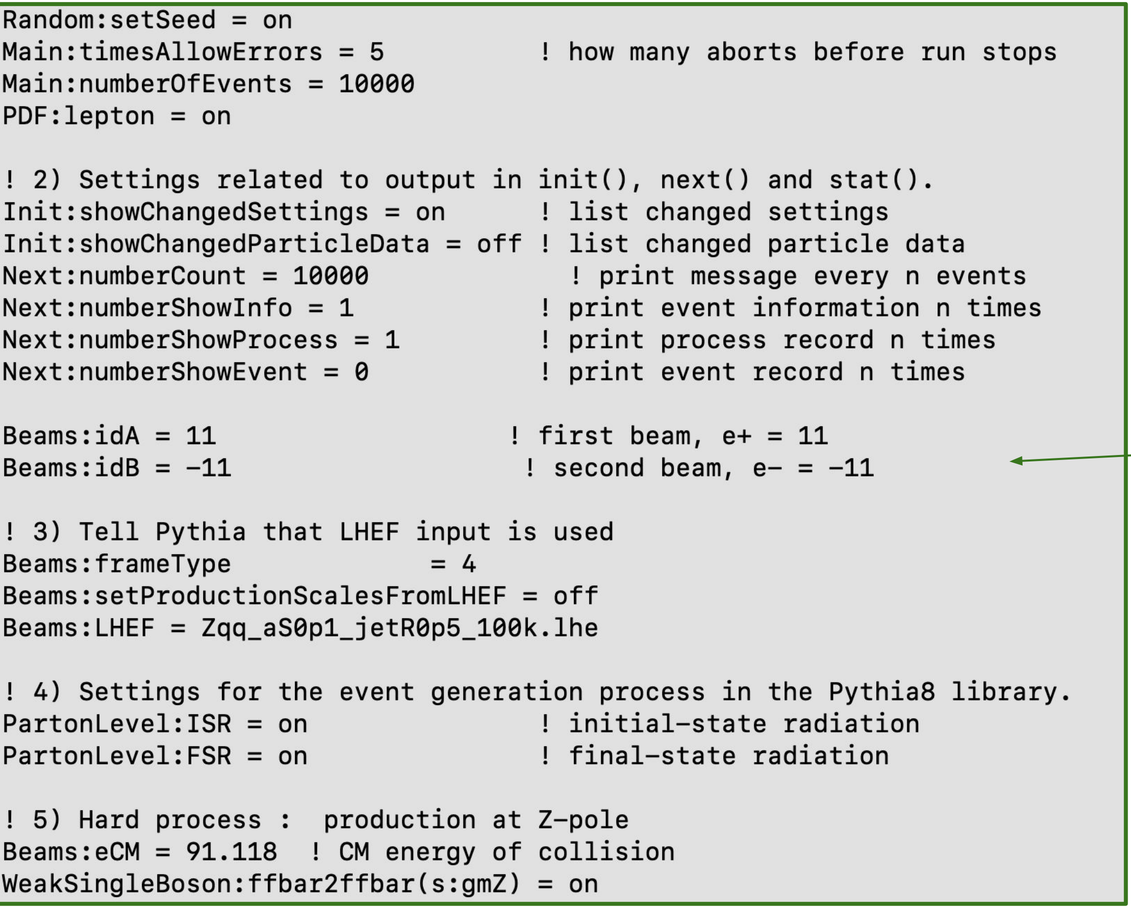
Config file to change the alphaS value:
madgraph/interface/common_run_interface.py
(Set alphaS for nn23nlo pdf, nn23nlo is default pdf in the cards)

Process card:
/afs/cern.ch/user/l/lpanwar/public/Zqq_aS0p117/Cards/proc_card_mg5.dat

Run card:
/afs/cern.ch/user/l/lpanwar/public/Zqq_aS0p117/Cards/run_card.dat
Change beam energy 45.559 GeV and set lpp1 and lpp2 as “0” for ee collision

Example Output of launch command:
/afs/cern.ch/user/l/lpanwar/public/Zqq_aS0p117/out.log

Pythia+Delphes simulation



```
Random:setSeed = on
Main:timesAllowErrors = 5          ! how many aborts before run stops
Main:numberOfEvents = 10000

PDF:lepton = on

! 2) Settings related to output in init(), next() and stat().
Init:showChangedSettings = on      ! list changed settings
Init:showChangedParticleData = off ! list changed particle data
Next:numberCount = 10000             ! print message every n events
Next:numberShowInfo = 1            ! print event information n times
Next:numberShowProcess = 1         ! print process record n times
Next:numberShowEvent = 0           ! print event record n times

Beams:idA = 11                   ! first beam, e+ = 11
Beams:idB = -11                   ! second beam, e- = -11

! 3) Tell Pythia that LHEF input is used
Beams:frameType             = 4
Beams:setProductionScalesFromLHEF = off
Beams:LHEF = Zqq_aS0p1_jetR0p5_100k.lhe

! 4) Settings for the event generation process in the Pythia8 library.
PartonLevel:ISR = on               ! initial-state radiation
PartonLevel:FSR = on               ! final-state radiation

! 5) Hard process :  production at Z-pole
Beams:eCM = 91.118  ! CM energy of collision
WeakSingleBoson:ffbar2ffbar(s:gmZ) = on
```

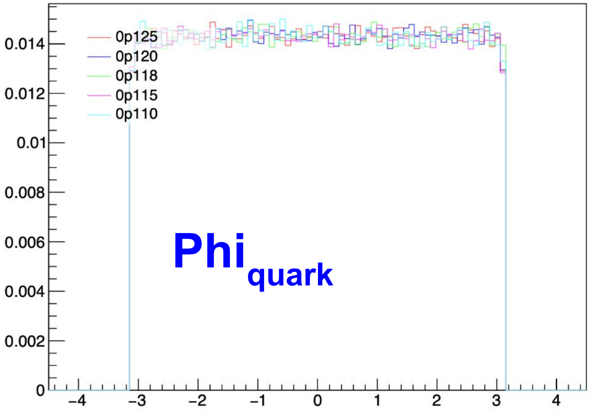
Within FCCAnalysis framework:

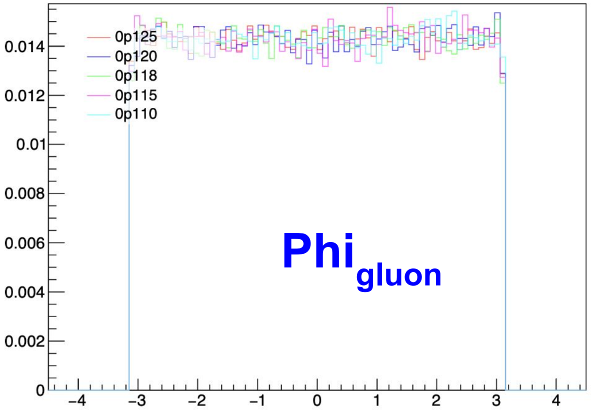
```
source /cvmfs/fcc.cern.ch/sw/latest/setup.sh
```

```
DelphesPythia8_EDM4HEP delphes-3.5.1pre05/cards/delphes_card_IDEA.tcl edm4hep_output_config.tcl p8_ee_Z_Zqq_ecm91_LHE.cmd out.root
```

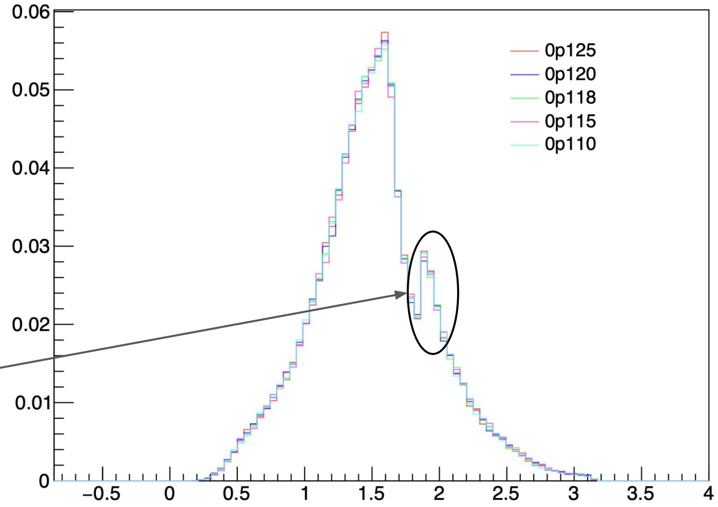
Note: Jet clustering setting is changed in delphes card

Jet algorithm numbering labels differ from fastjet





Angle bw g and the closest q



Events in second peak region comes due to gluon recoils against both quarks

Peaked structure is still under investigation

$p_Z = p_{q'} + p_q + p_g$ ## 3-momentum conservation

Since Z is produced at rest

$0 = p_{q'} + p_q + p_g$

$\Rightarrow -p_{q'} = p_q + p_g$

$\Rightarrow p_{q'}^2 = (p_q + p_g)^2$

$\Rightarrow p_{q'}^2 = p_q^2 + p_g^2 + 2p_q p_g \cos\theta$

$\Rightarrow p_{q'}^2 - p_q^2 - p_g^2 = 2p_q p_g \cos\theta$

$\Rightarrow \cos\theta = (p_{q'}^2 - p_q^2 - p_g^2) / 2p_q p_g$

When gluon is very soft $p_g \sim 0$ and $p_{q'}^2 \sim p_q^2 \Rightarrow \theta \sim \pi/2$