Physics Analysis with Jets at CMS

Kalanand Mishra
Fermilab

JTerm IV, August 3, 2009

Outline of this talk

- Concepts
  - What is a Jet?
  - Jets in CMS
  - Jet calibration
  - Jet Id
  - Jet cleaning
- Practical Instructions for Beginners
- Jets in PAT
- Performance
  - Jet Reconstruction Efficiency
  - Jet Resolution
- Links to More Information

Why should you care about jets ?

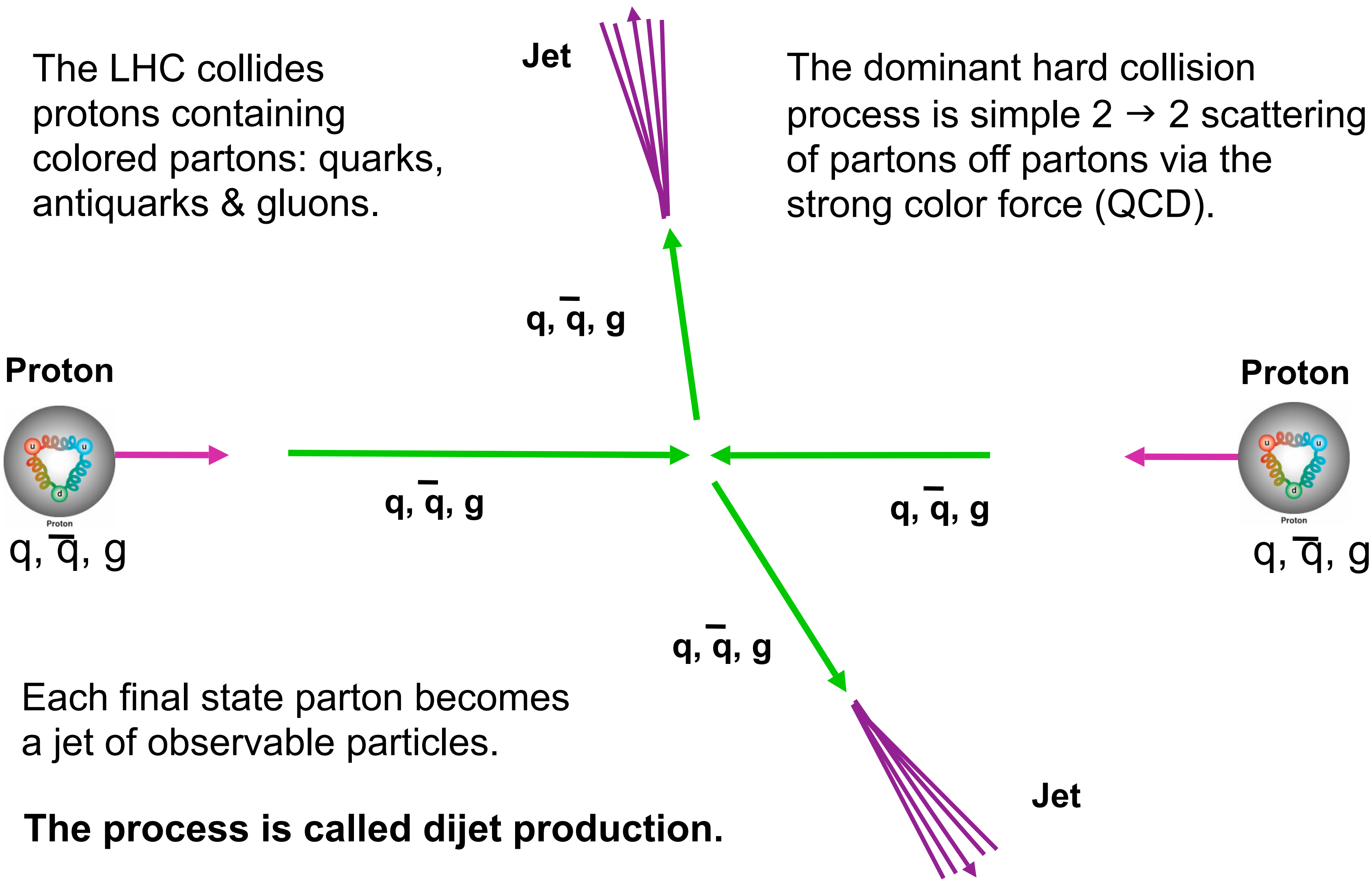
- Jets are everywhere; their cross section is orders of magnitude higher than most other processes.
- Jets can fake as γ, e, μ, τ
  - Probability of jet faking a γ ~10^{-4}
  - Probability of faking e/μ ~10^{-5}, but some jets have real lepton, e.g., b-jets
  - Probability of faking a τ ~10^{-3}
- Light quark or gluon jets can fake b-quark jet at the % level
- Missing Transverse Energy must be corrected for jet energy measurements.

If jets are not your signal they are most certainly your background !

Jets at LHC

The LHC collides protons containing colored partons: quarks, antiquarks & gluons.

The dominant hard collision process is simple 2 → 2 scattering of partons off partons via the strong color force (QCD).

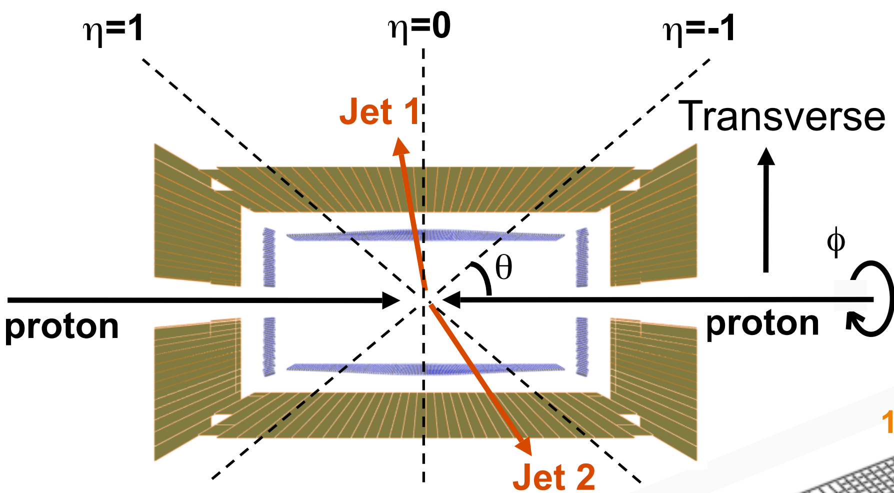


Each final state parton becomes a jet of observable particles.

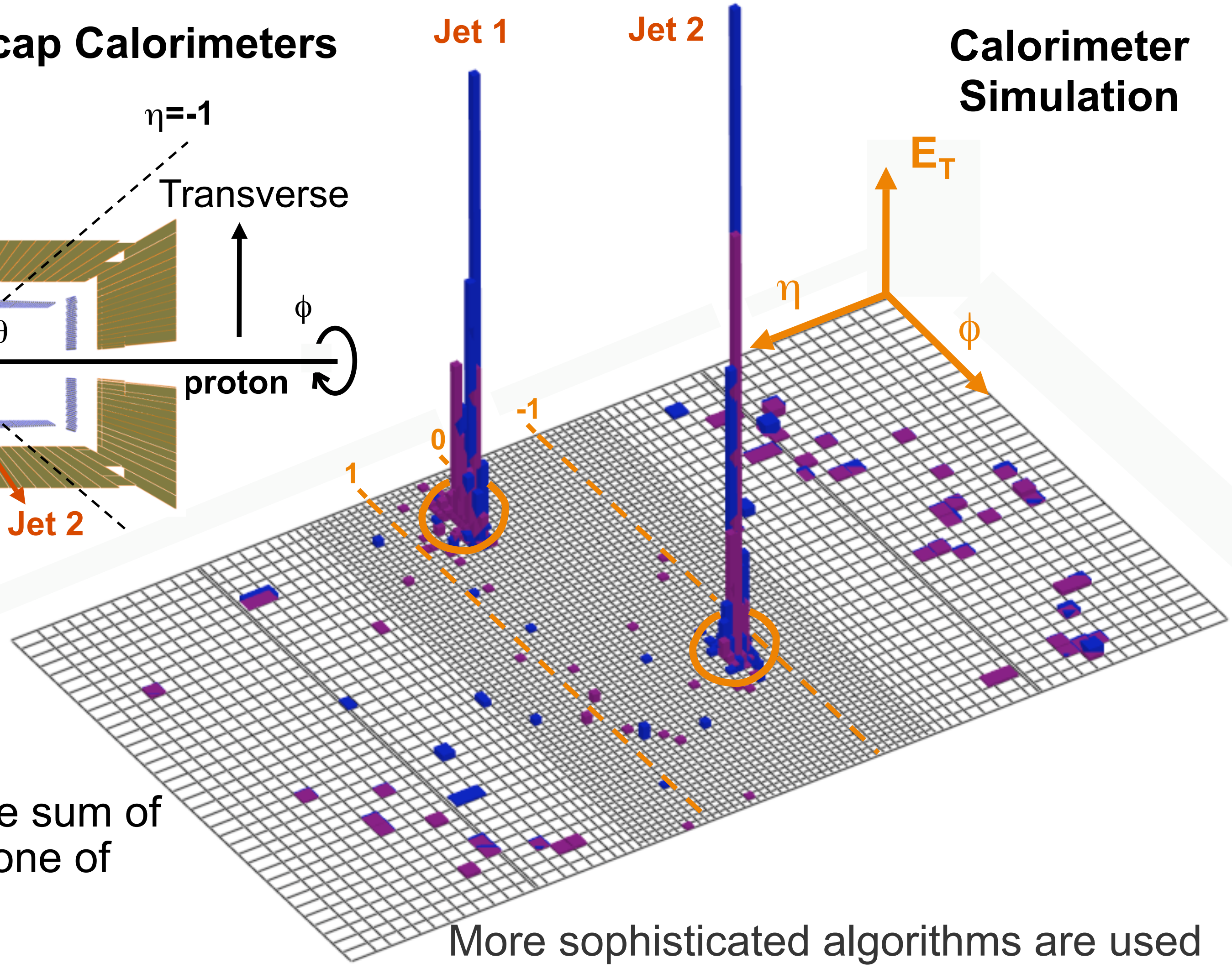
The process is called dijet production.

Experimental observation of jets

CMS Barrel & Endcap Calorimeters



Calorimeter Simulation

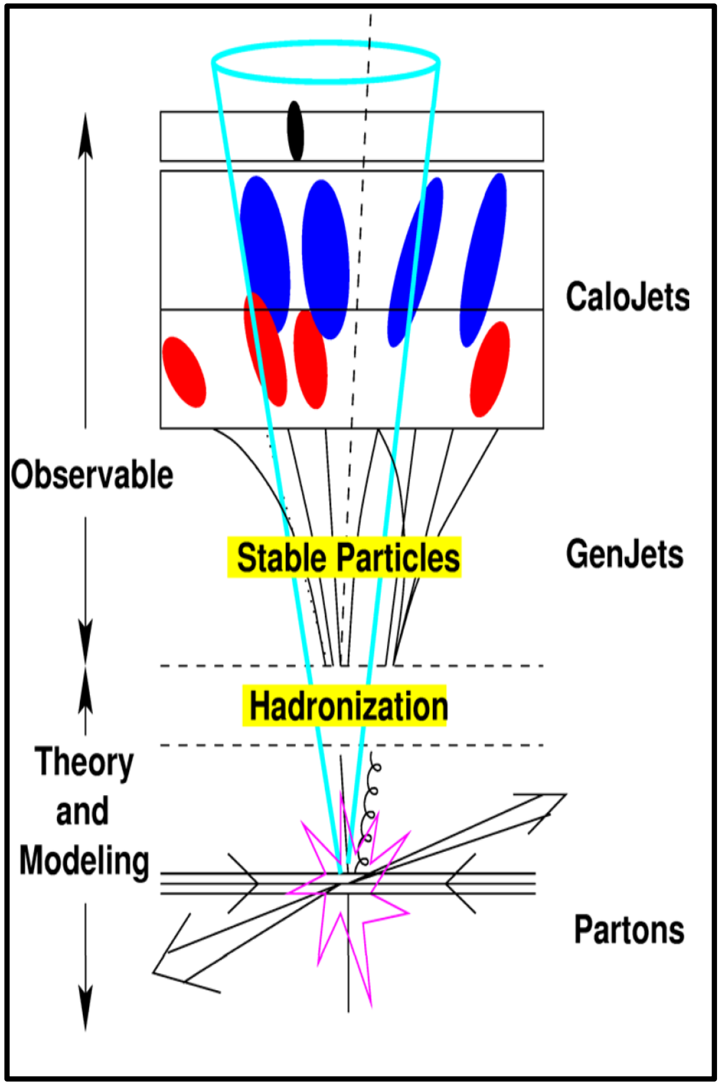


In CMS trigger, a jet is the sum of calorimeter towers in a cone of radius

$$R = \sqrt{\Delta\eta^2 + \Delta\phi^2} = 0.5$$

More sophisticated algorithms are used in offline analysis to reconstruct jets.

Jets in CMS



Official Jet Algorithms @ CMS

1. **S**eedless **I**nfrared **S**afe Cone

Fixed size cone of radius = 0.5, 0.7. Infrared and collinear safe.

2. KT

Successive recombination with resolution parameter =0.4, 0.6. Infrared and collinear safe.

3. Iterative Cone

Fixed size cone with radius =0.5. NOT Infrared and collinear safe but fast!!! Used by HLT.

For details on jet algo: arXiv:0906.1833 by Gavin Salam

The jet algorithms take as input a set of 4-vectors:

- CaloJets: Take CaloTowers as input.
- GenJets: Stable MC truth particles as input.
- TrackJets:Tracks as input.
- PFJets: Reco. stable particles as input.
- JPT: CaloJets corrected with track of p_T measurement in the tracker.

Getting hands dirty: How to access jets in CMSSW ?

https://twiki.cern.ch/twiki/bin/view/CMS/WorkBookJetAnalysis#JetAna

Example code: RecoJets/JetAnalyzers/src/JetValidation.cc

```
#include "DataFormats/JetReco/interface/CaloJetCollection.h"
#include "DataFormats/JetReco/interface/CaloJet.h"
............................
calAlgo = cfg.getParameter<string> ("calAlgo");
............................
Handle<CaloJetCollection> caljets;
event.getByLabel(calAlgo,caljets);
CaloJetCollection::const_iterator i_jet;
for(i_jet = caljets->begin(); i_jet != caljets->end(); ++i_jet)
 {
  e = i_jet->energy();
  pt = i_jet->pt();
  phi = i_jet->phi();
  eta = i_jet->eta();
  ............................
  emEB = i_jet->emEnergyInEB();
  hadHB = i_jet->hadEnergyInHB();
  ............................
}
```

Configurable jet collection name (via python config script)

Handle the jet collection

Define an iterator for the jet collection

BasicJet quantities

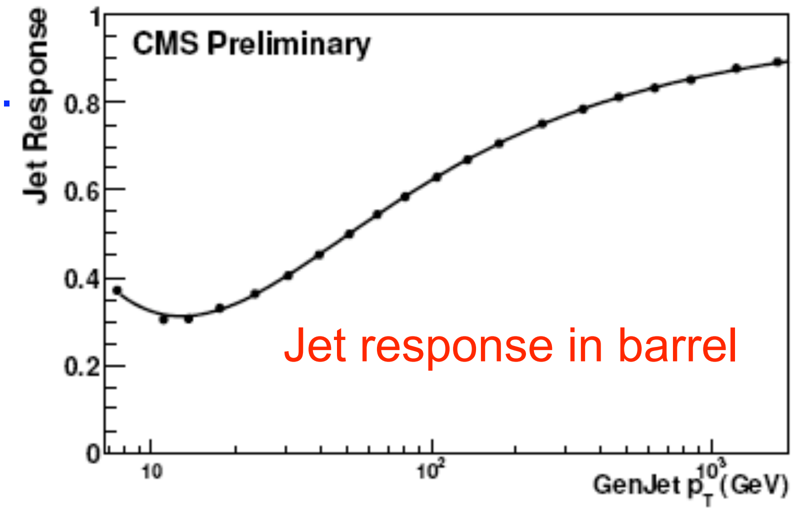
CaloJet specific quantities

- iterativeCone5CaloJets
- sisCone5CaloJets
- sisCone7CaloJets
- kt4CaloJets
- kt6CaloJets

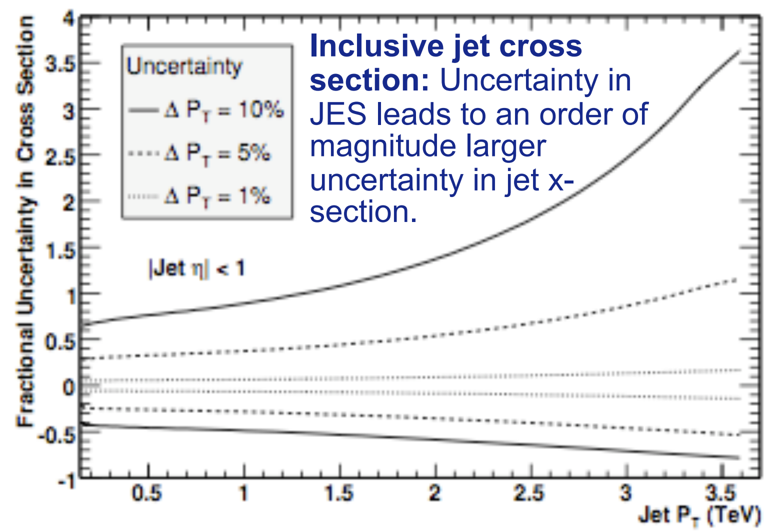
GenJets* can be accessed in a similar way: *Calo* → *Gen

Jet energy scale & why does it matter?

- ✦We do not see quarks and gluons
  - -We do not see all stable particles: π,K,η,...
  - -How do we go from raw inputs (calorimeter/ track energy) to the particle level energy? ⇨ Jet Energy Scale



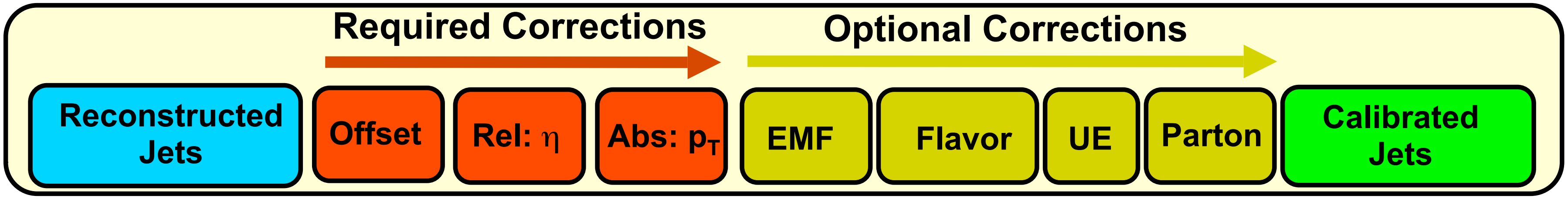
- ✦Factors impacting the JES include
  - - Calorimeter response
  - - Effect of B field (sweeps particles away)
  - - Energy offset (*i.e.*, energy not from the hard scattering process)
  - - Material in front of the calorimeter
  - - Out-of-cone showering
  - - Resolution ⇒ unsmearing



An example of SM measurement

- ✦JES uncertainties typically are the largest systematic errors in jet measurements.
  - -Depending on p_T, we expect ~10% initial uncertainty in JES.

Jet corrections are factorized



- **Plan:** the jet corrections will be factorized
  - Correcting for each factor in a fixed sequence up to a level chosen by the user.
- Factorization facilitates the use of data-driven corrections
  - Breaking the correction into pieces that are naturally measured in collider data:
    - **Offset**: pile-up and noise measured in zero-bias events.
    - **Relative**: jet response vs. η relative to barrel found using dijet balance.
    - **Absolute**: jet response vs. P_T found in barrel using γ / Z + jet.
  - Allows data-driven corrections as they emerge to easily replace MC truth

$$p_T^{corrected} = Abs(p_T \cdot Rel(\eta, p_T)) \times Rel(\eta, p_T) \times (p_T - \text{offset})$$

Absolute correction is applied to the jets which have already been corrected for η dependence

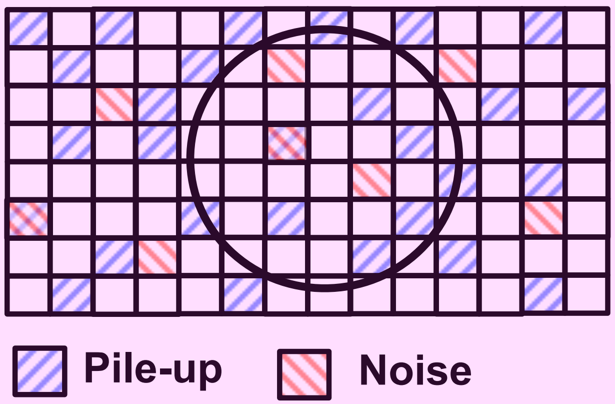
Relative correction is applied to the jets which have already been “offset” corrected

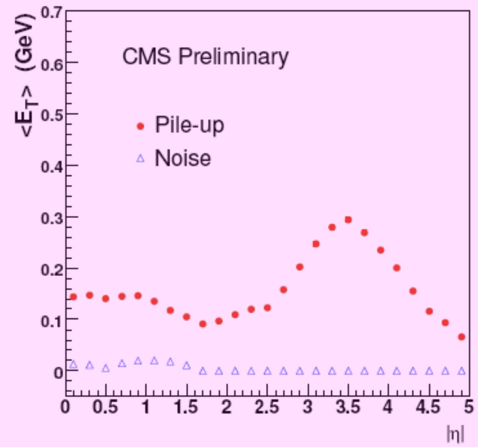
Offset correction is applied to the uncorrected jets

Combined correction brings back the jet to the particle level

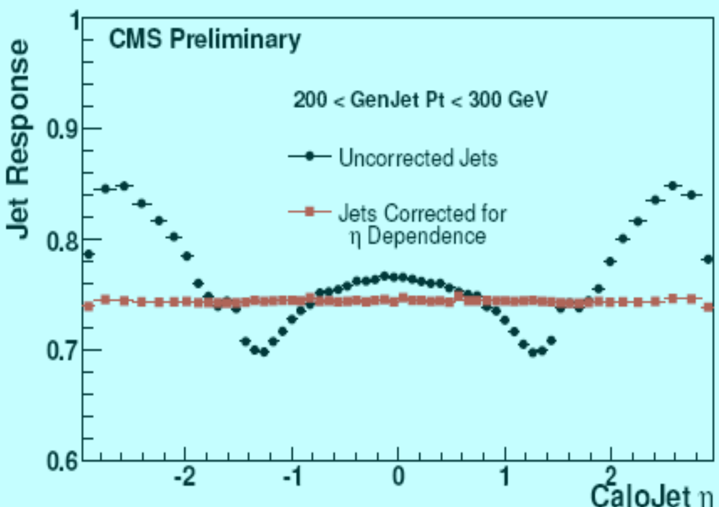
Required corrections

Offset in Jet Area In Calorimeter





Relative Correction in η



The relative correction provides answer to the question: if a particle jet with p_T^{gen} at a given η is measured in CMS with p_T^{cal}, what would be measured in the control region for the same input p_T^{gen} ?

Absolute Correction in p_T

The absolute correction flattens the jet response in p_T, corrects the energy of the jet back to the particle level in control region (|η| <1.3)

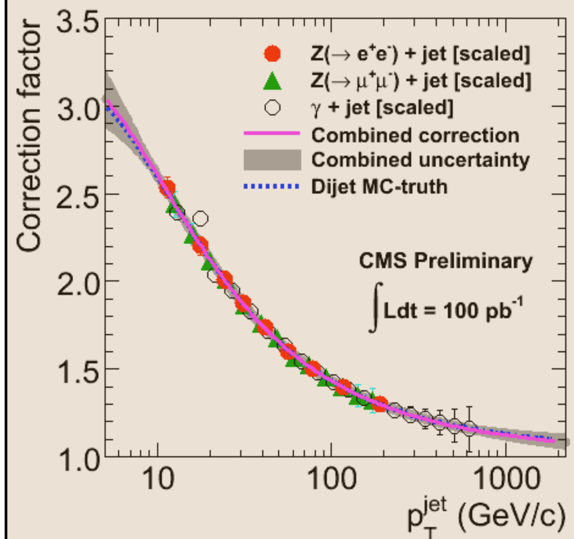


Fig: Jet energy absolute scale correction obtained by combining results from γ+jet and Z+jet samples, and extrapolation using dijet MC.

How to apply jet corrections in CMSSW ?

- The "jet correction service" is the software that delivers the correction factor.
- The "correction module" delivers the corrected, re-ordered, jet collection.
- Currently available jet corrections are derived from MC truth.

Example configuration: RecoJets/JetAnalyzers/test/runL2L3JetCorrectionExample_cfg.py

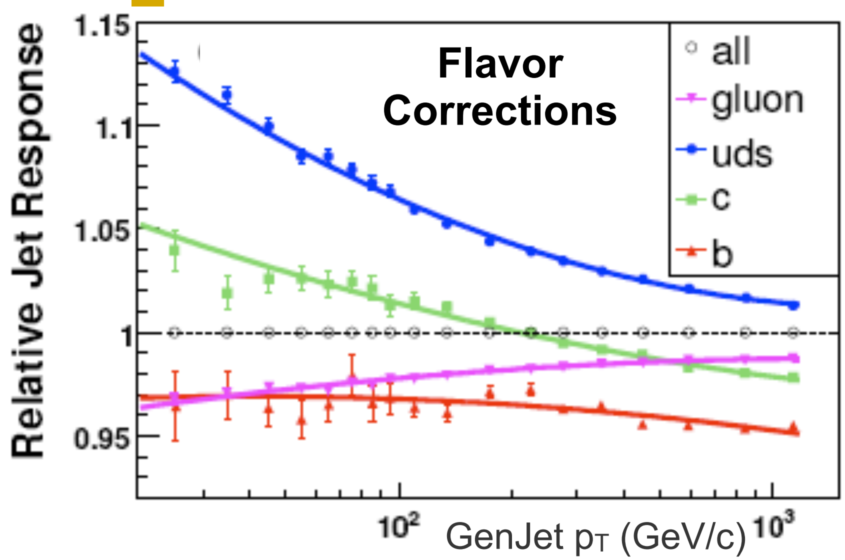
```
..............................................
process.load("JetMETCorrections.Configuration.L2L3Corrections_Summer08_cff")
process.prefer("L2L3JetCorrectorIC5Calo")
required by CMSSW but IRRELEVANT to the actual corrections applied. ANY
correction service defined in L2L3Corrections_Summer08_cff can be used.
..............................................
process.corrected = cms.EDAnalyzer("CaloJetPlotsExample",
    JetAlgorithm  = cms.string('L2L3CorJetIC5Calo'),
    HistoFileName = cms.string('CorJetHisto.root'),
    NJets         = cms.int32(2)
)
process.p = cms.Path(process.L2L3CorJetIC5Calo * process.corrected)
```

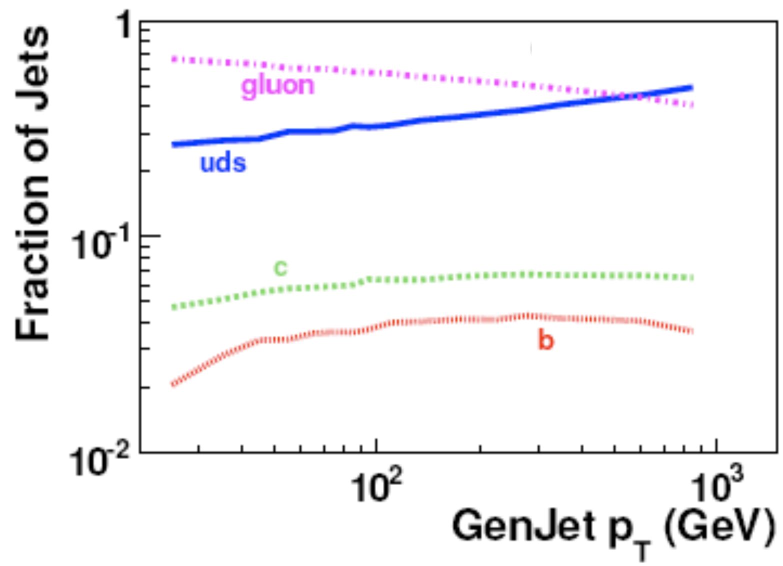
Step 1: Include the file defining the default jet correction services and modules

Step 2: Use the corrected jet collection in your EDAnalyzer

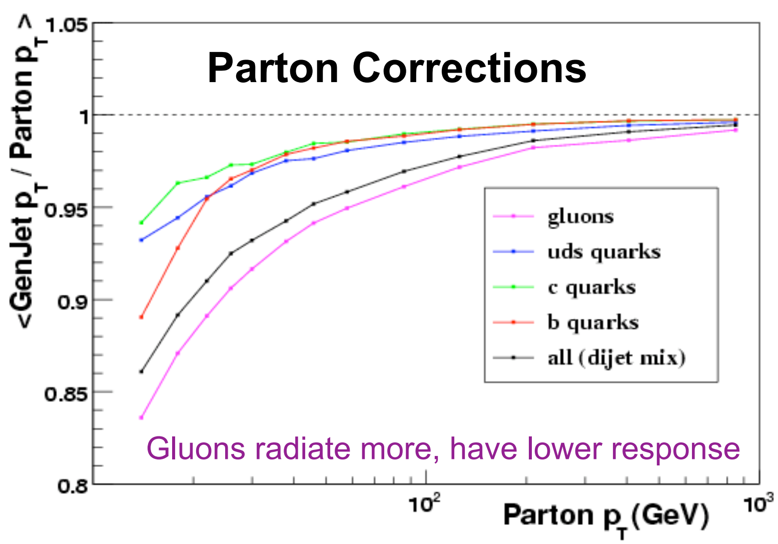
Step 3: Include the correction module in the path BEFORE your EDAnalyzer

Optional corrections: Flavor and Parton





Fraction of jets of each flavor in the QCD dijet sample



- ✦ Light quarks have higher response than gluons as they fragment into higher p_T particles
- –QCD dijet events have mostly gluons
- – γ/Z+jet events are rich in quarks, have higher jet response
- ✦ The parton correction is intended to provide correction between the GenJet and parton level jet for parton showering and hadronization.

Where to get information on jet corrections ?

How can I learn more about jet energy corrections?

https://twiki.cern.ch/twiki/bin/view/CMS/WorkBookJetAnalysis#JetCorApplication

In the above TWiki you can find examples for applying default/optional corrections.

Where can I find details on the derivation of jet corrections?

- JME-07-002: "Plans for Jet Energy Corrections at CMS"
- JME-09-003: Offset correction
- JME-08-003: Relative correction using dijet balance
- JME-09-005: Absolute correction from γ/Z p_T balance (also JME-09-004, JME-09-009)
- JME-08-002: Parton correction

Where can I find details on jet reconstruction, efficiency, and resolution?

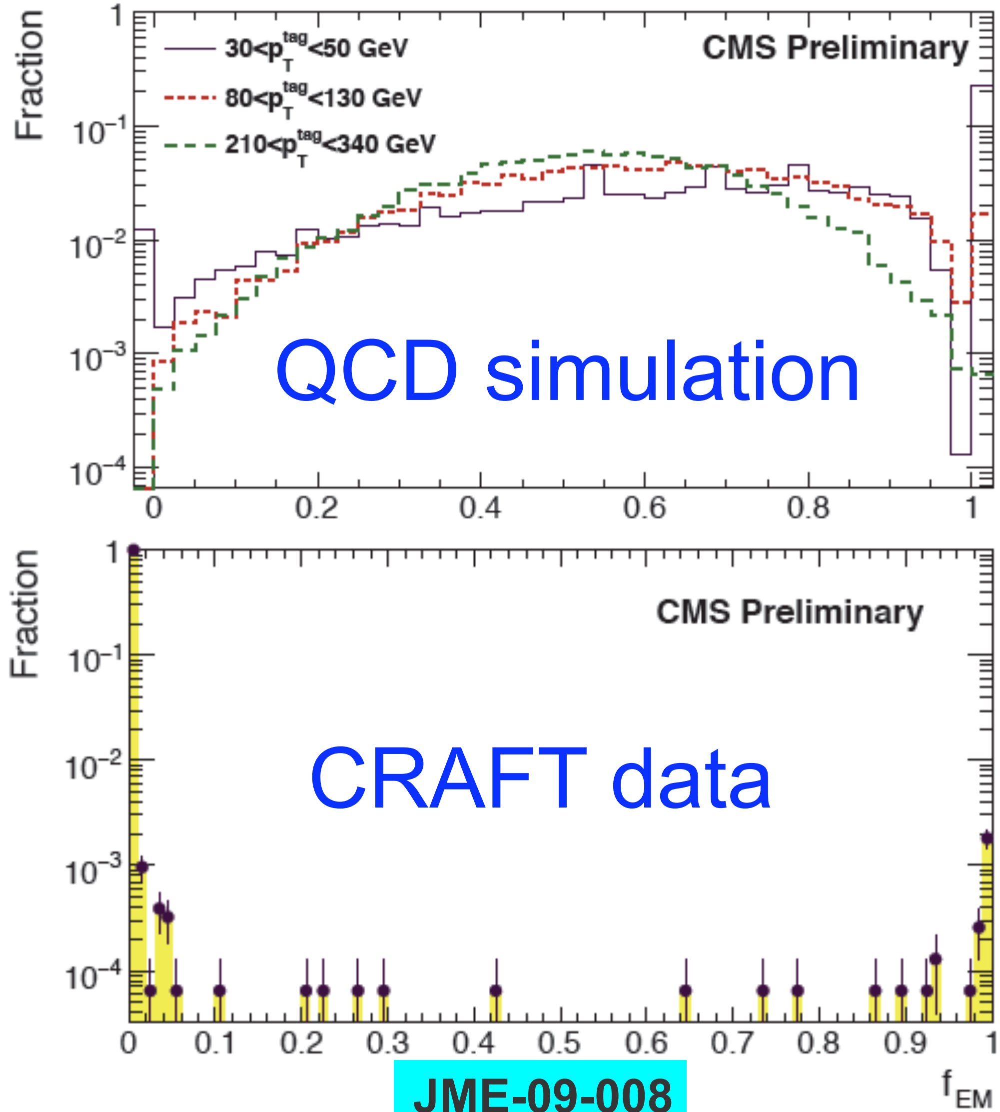
- JME-07-003: Jet algorithms
- JME-09-007: Jet resolution & jet reconstruction efficiency
- JME-09-002: Jet-plus-tracks algorithm

All the above documents are available publicly from the *CMS Physics Results* web page:
https://twiki.cern.ch/twiki/bin/view/CMS/PhysicsResults

IMPORTANT: Don't hesitate to ask the experts!!!!!

Jet Id

- Studies, based on CRAFT data and MC simulation, indicate that the electromagnetic energy fraction (EMF or f_{EM}) is a powerful criterion to reject fake jets.
- Other quantities are being examined too:
  - #CaloTowers
  - #CaloTowers containing 90% of the jet energy
  - RMS of the E_T weighted ϕ distribution
  - fraction of energy from the hottest HPD/ RBX
- Data-driven method methods to measure the jet id efficiency are being developed
- Software implementation of the jet ID is being examined.



JME-09-008

Jet cleaning & matching

Why do I need to clean jets ?

- Any object which deposits energy in HCAL can get reconstructed as a jet !
- This implies that jet collections also contain electrons, muons, photons, ...
- Before doing analysis with jets, these objects need to be removed from the jet collection.

Jet matching

- For Monte Carlo based analysis one may be interested to know what fraction of the reconstructed jets is matched to generator level quantities: GenJets or partons.
- In data-driven analyses of certain types of events (e.g., dijet, Z+jet, photon +jet) one may want to use p_T balance between a jet and a reference object by doing back-to-back matching in ϕ.

We have standard tools in CMSSW to perform these recurring tasks and to compute efficiency for each step ➜ see next slide

Standard tools for jet cleaning & matching

CMSSW/JetMETAnalysis/JetUtilities

```
import FWCore.ParameterSet.Config as cms

ic5CaloJetsClean = cms.EDFilter("JetViewCleaner",
    srcJets = cms.InputTag("iterativeCone5CaloJets"),
    module_label = cms.string('ic5CaloJetsClean'),
    srcObjects = cms.VInputTag(cms.InputTag("gsfElectrons")),
    deltaRMin = cms.double(0.3)
)
```

The *JetViewCleaner* loops over the object collection and removes all jets within a specified ΔR of the object. Useful for removing electrons, muons, photons from your jet collection.

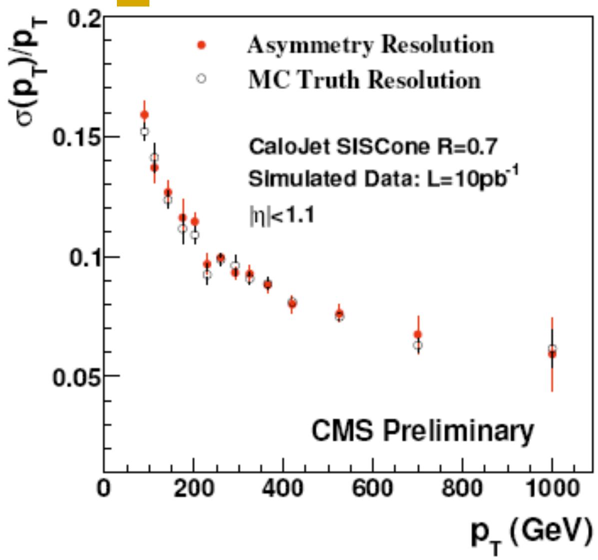
```
import FWCore.ParameterSet.Config as cms

ic5MatchedJets = cms.EDFilter("MatchRecToGen",
    srcRec = cms.InputTag("iterativeCone5CaloJets"),
    srcGen = cms.InputTag("iterativeCone5GenJets"),
    module_label = cms.string('ic5MatchedJets'),
)
```

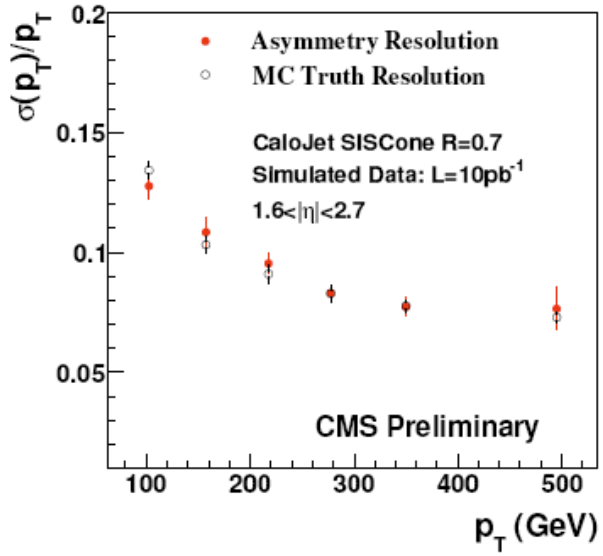
The *MatchRecToGen* loops over the CaloJet and GenJet collections and produces two reco::*CandViewMatchMap* objects - "rec2gen" and "gen2rec". These contain the association map and ΔR.

There is also another ED filter called *"MatchBackToBack"* which can be useful for sometimes.

These filters also print out the cleaning/matching efficiency at the end of your job.

Jet resolution and reconstruction efficiency

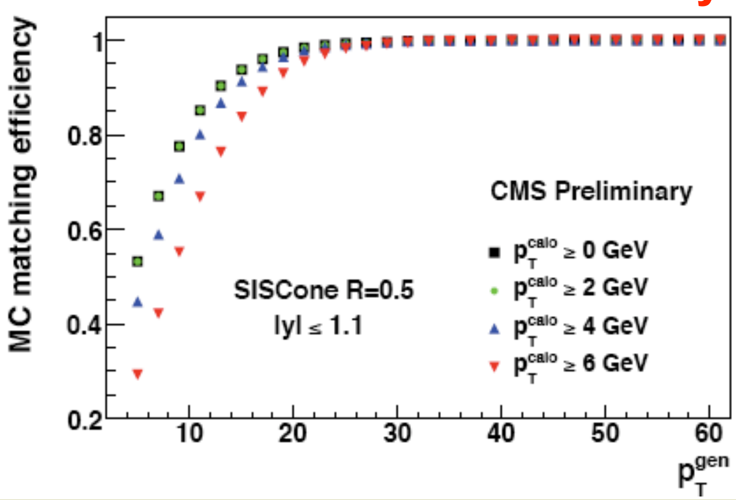




Jet p_T resolution

- Jet resolution improves at higher p_T.
- Resolution is slightly better in endcap than in the barrel !
- Can be measured in data using p_T balance in the QCD dijet events (asymmetry method).

Jet reconstruction efficiency



MC truth matching efficiency for several choices of CaloJet p_T threshold. The matching criteria is $\Delta R < 0.5$.

- The jet reconstruction efficiency is practically 100% for jets with p_T > 20 GeV/c.
- BUT this does not let you off the hook for syst. uncertainty:
  - -A 15 GeV jet can easily fluctuate to 30 GeV and efficiency at 15 GeV ≠ 100%.

Jets in Physics Analysis Toolkit

- The pat::Jet is the basic jet object
- Stores internally the jet correction factors that can be retrieved:
  jetCorrFactor(string &step, string &flavor="")
- In the default PAT configuration, some jet corrections - relative and absolute - applied (based on MC). Can do correction on the fly.
- Stores jet flavor (from MC) and jet Id vars.
- Can do MC matching with GenJets/partons.
- Can use any kind jet collection: pFlow, JPT.

PAT: jetProducer_cff

```
import FWCore.ParameterSet.Config as cms

# prepare reco information
from PhysicsTools.PatAlgos.recoLayer0.jetTracksCharge_cff import *
from PhysicsTools.PatAlgos.recoLayer0.jetMETCorrections_cff import *

# add PAT specifics
from PhysicsTools.PatAlgos.mcMatchLayer0.jetFlavourId_cff import *
from PhysicsTools.PatAlgos.mcMatchLayer0.jetMatch_cfi import *

# produce object
from PhysicsTools.PatAlgos.producersLayer1.jetProducer_cfi import *

makeAllLayer1Jets = cms.Sequence(
    # reco pre-production
    patJetCharge *
    patJetCorrections *
    # pat specifics
    jetPartonMatch *
    jetGenJetMatch *
    jetFlavourId *
    # object production
    allLayer1Jets
    )
```

Many thanks to Salvatore Rappoccio & Roger Wolf for providing these information.

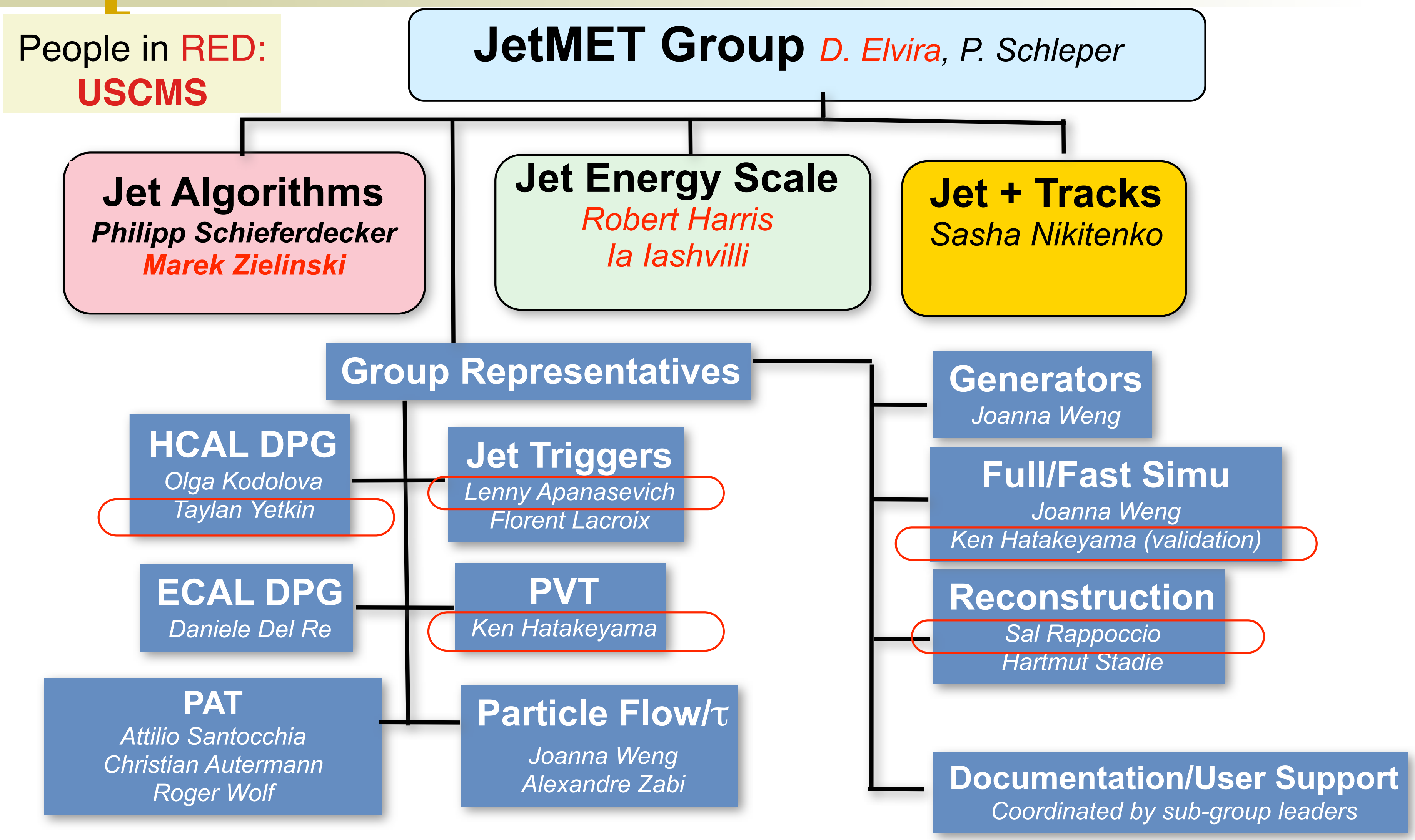
Be careful when using PAT jets !

- Jets should be ordered in decreasing p_T. BUT PAT::Jet collections in 1.6.X and 2.0.X were ordered by E_T.
- Sometimes stored correction factors are negative. <u>One should use the absolute value.</u>

For more information: PAT twiki & Roger Wolf's talks

https://twiki.cern.ch/twiki/bin/view/CMS/SWGuidePATLayer1
http://indico.cern.ch/contributionDisplay.py?contribId=47&confId=62064
http://indico.cern.ch/contributionDisplay.py?contribId=18&confId=55313

CMS Jet Physics Object Group



Summary

✓**Jet cross section at LHC is several orders of magnitude higher than any other process**

-Jets will be the first objects to be observed and most frequent at CMS!!!

-We need to understand their performance as early as possible.

✓**The baseline jet object at CMS is the calorimeter jet (CaloJet)**

-Jets need to be calibrated and cleaned before they can be used for physics analysis.

-Default calibration corrects the measured jet energy to the particle level.

-Depending on your physics analysis, you may need to apply flavor and/or parton corrections on top of that.

✓**There exist documentation and examples to help you with jets**

-But a lot of things need to be done before first encounter with real data.

-Contact the group leaders and volunteer your time if you can help.

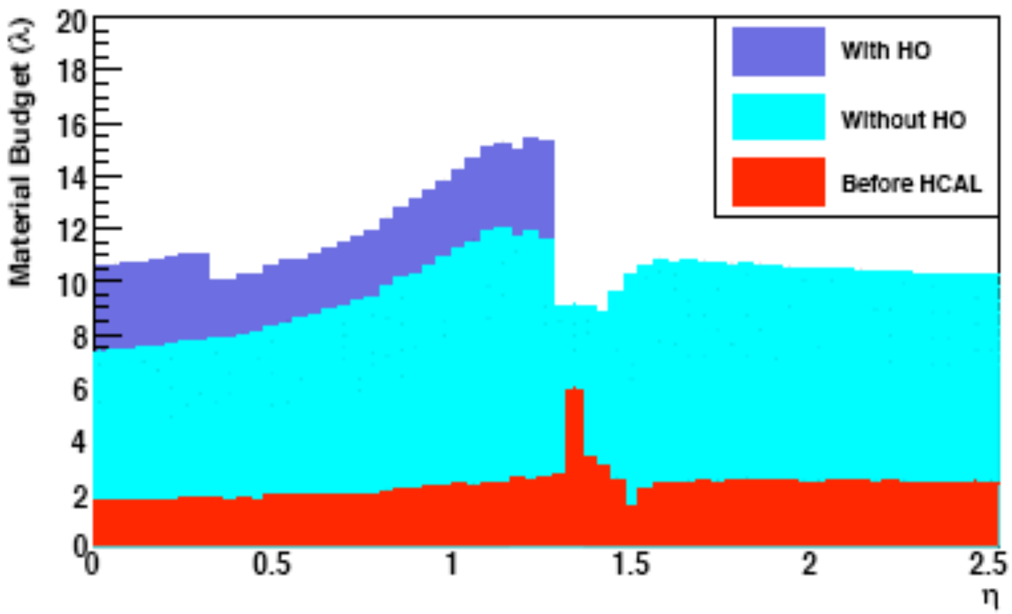
✓Many thanks to Frank Chlebana, Daniel Elvira, Robert Harris, Kostas Kousouris, and Marek Zielinski for help and feedback.

LPC is full of jet experts!!!! Take the time to speak to them!!!!

Backup

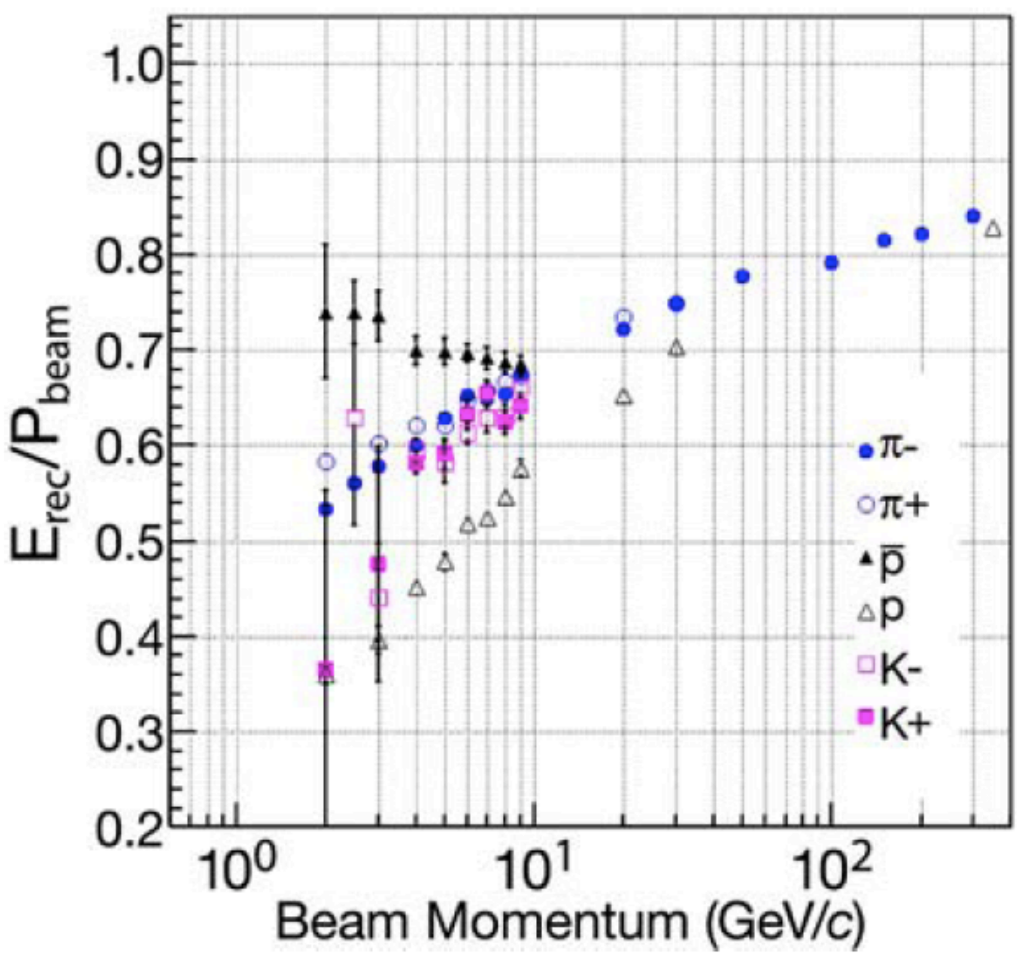
Material budget of the calorimeter

Thickness of HCAL in terms of interaction lengths



7-8 Interaction Lengths at $\eta=0$ with HCAL alone and is insufficient to fully contain the shower generated by pions above 100 GeV

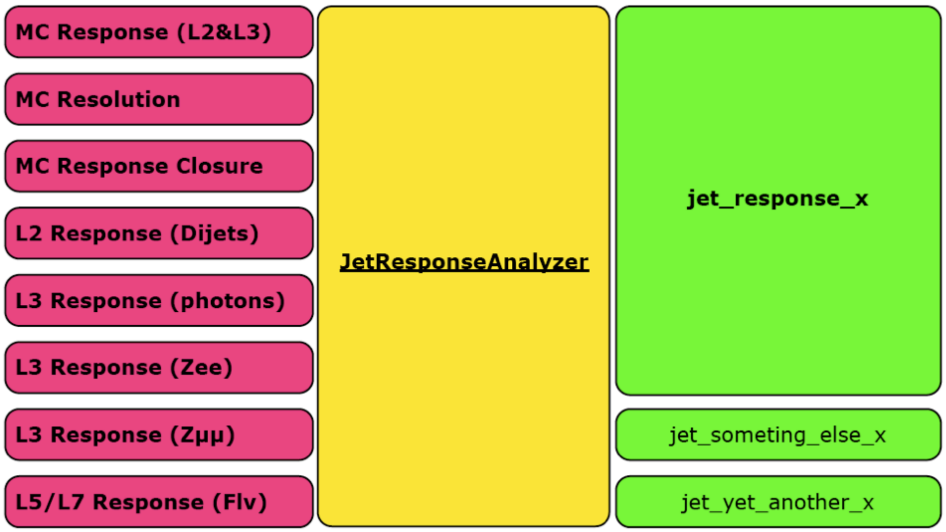
Calorimeter response in test beam data



- The figure shows the combined response of EB+HB calorimeter to different particles as a function of beam momentum.
- The response is normalized to 1 for electron.
- At 100 GeV/c, the pion response is 80 % of the electron response.
- The proton response is always lower than the pion response.
- In collision data the response is expected to be lower than in test beam because of additional material in front of the calorimeter.
- The calorimeter response is clearly non-linear.

JetResponseAnalyzer: A multi-purpose tool

https://twiki.cern.ch/twiki/bin/view/CMS/SWGuideJetResponseAnalyzer



- CMSSW framework analyzer
- records the necessary information to perform response and/or resolution measurements
- can be used for MC or data analysis
- fully configurable
- “official” tool, debugged by experts
- can handle all jet definitions (O(100)).

For more details please see the following presentations:

http://indico.cern.ch/getFile.py/access?contribId=5&resId=0&materialId=slides&confId=38581
http://indico.cern.ch/getFile.py/access?contribId=2&resId=0&materialId=slides&confId=46778