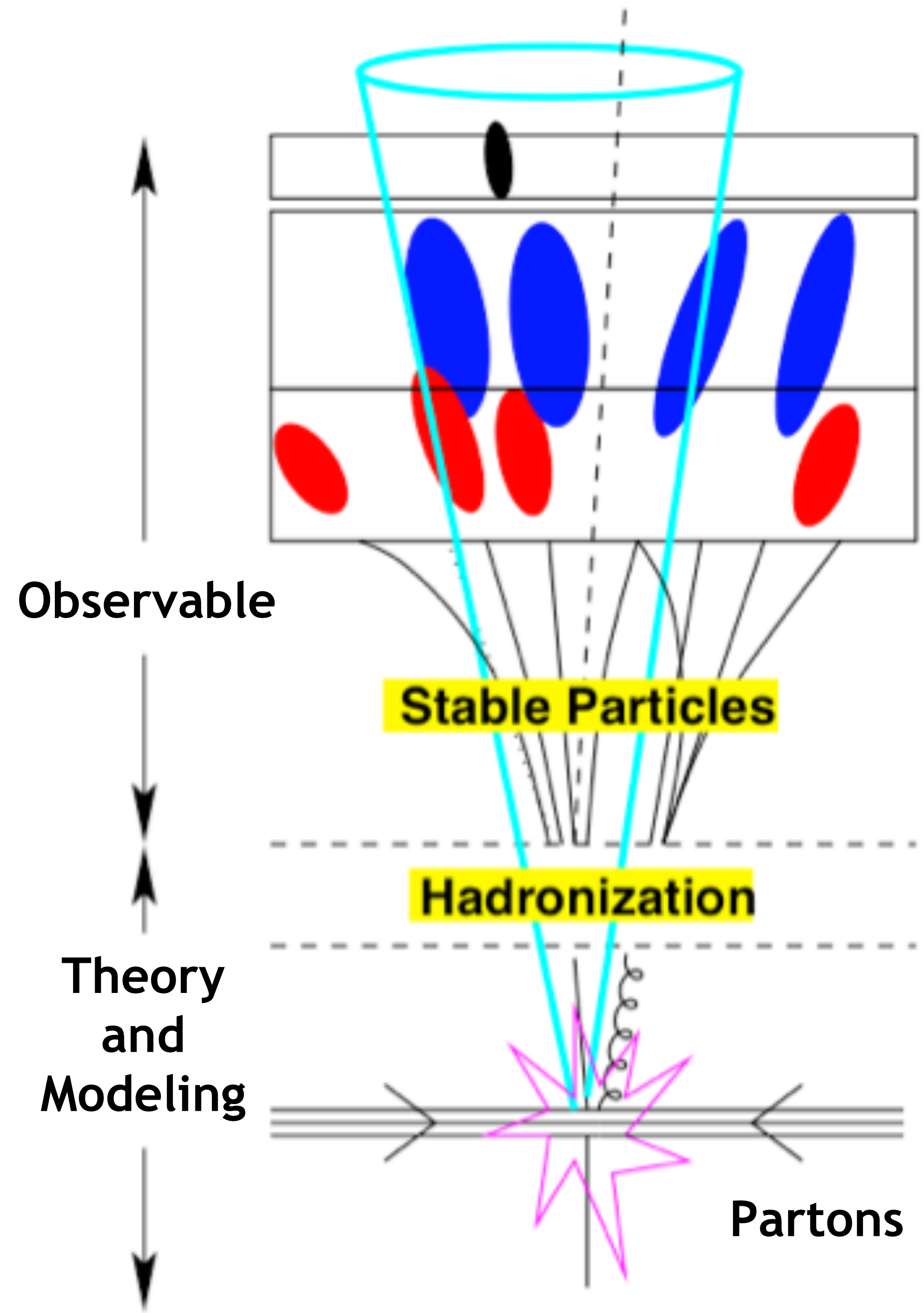
Jet Reconstruction and Particle Flow

Philipp Schieferdecker (KIT)

Outline

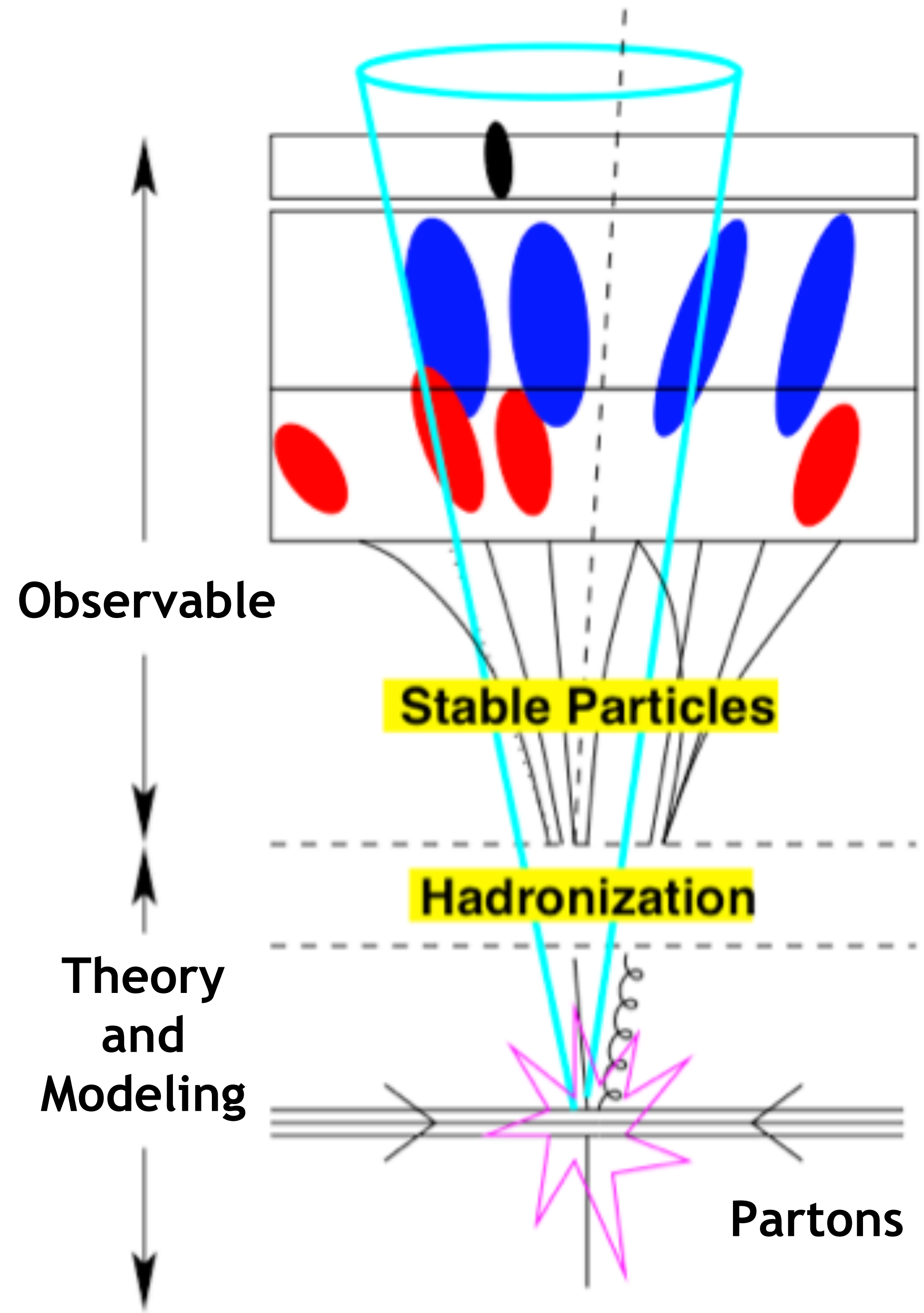
- What are Jets?
- Formation of Jets
- What is a Jet Algorithms?
- Theoretical and Experimental Jet Algorithm Requirements
- Cone Algorithms
- Sequential Clustering Algorithms
- Making everybody happy: Anti-kT
- Jet Areas & Pile-Up Subtraction
- Calorimeter Towers, Topological Clusters & Particle Flow
- Jet Performance in first CMS Collision Data

What are Jets?



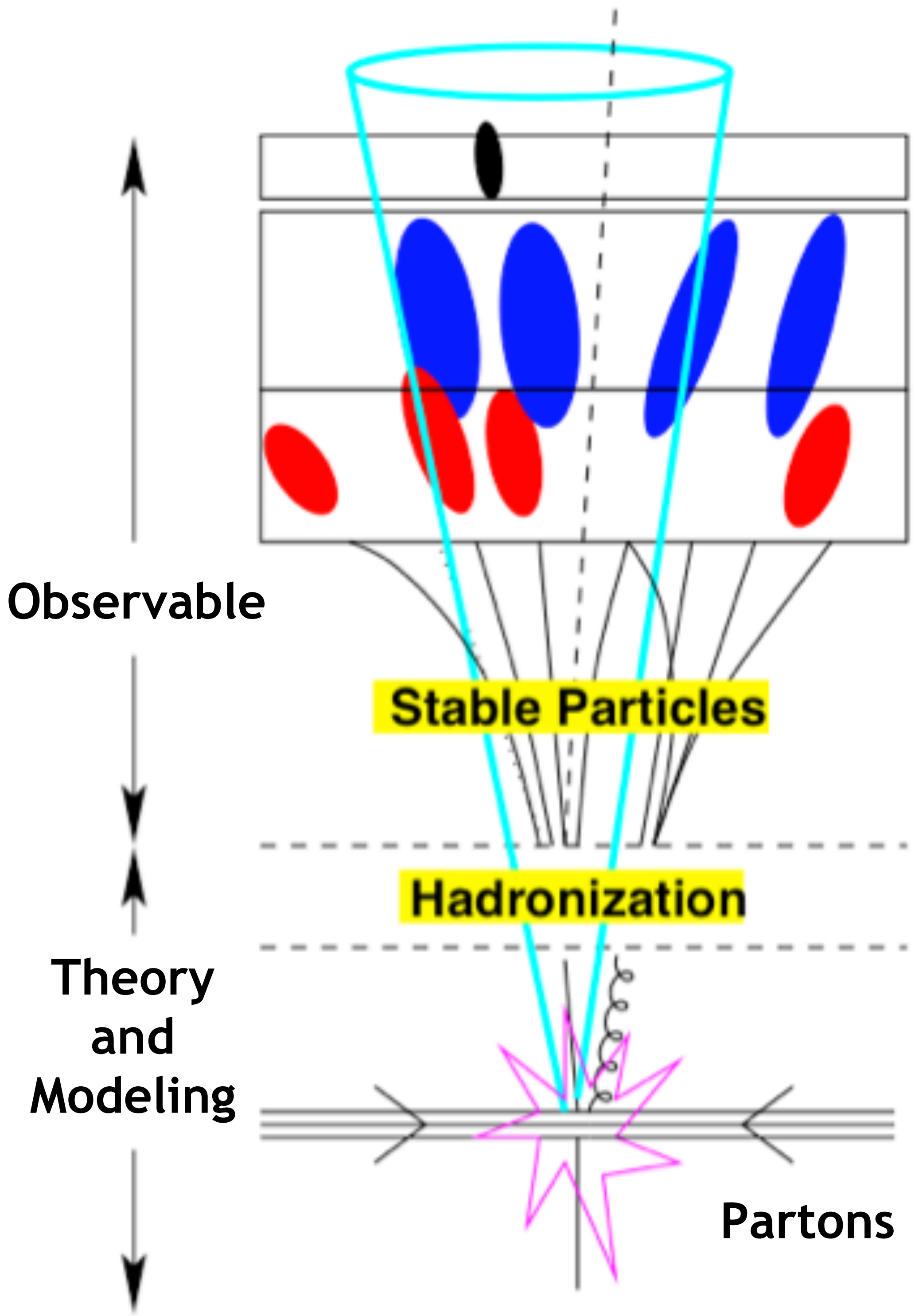
What are Jets?

- Due to the special nature of the strong force ("confinement"), **quarks & gluons** (partons) can not be observed directly



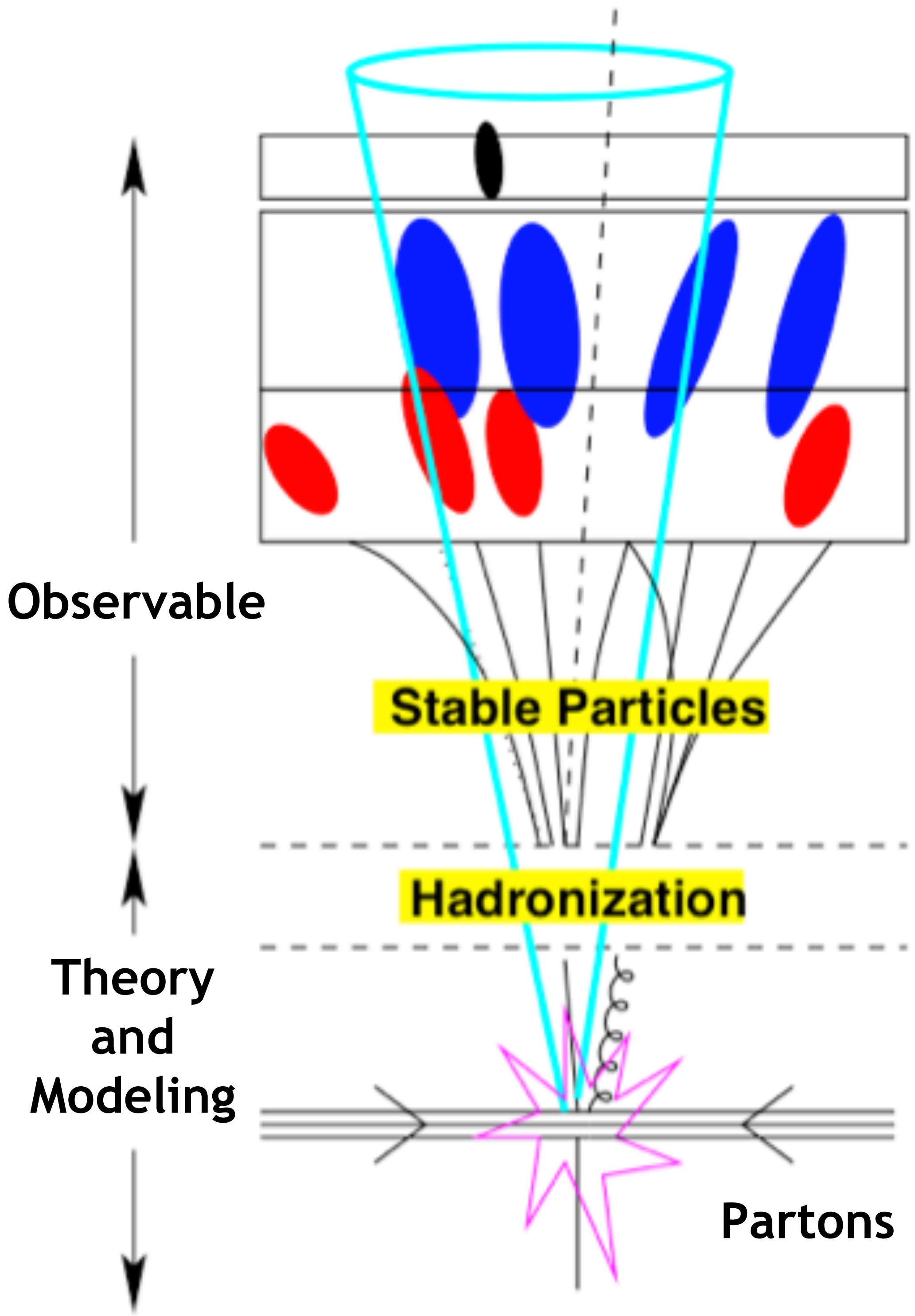
What are Jets?

- Due to the special nature of the strong force (“confinement”), **quarks & gluons** (partons) can not be observed directly
- Instead we observe collimated bunches of stable colorless hadrons, originating from partons after **fragmentation** and **hadronization**: **Jets!**



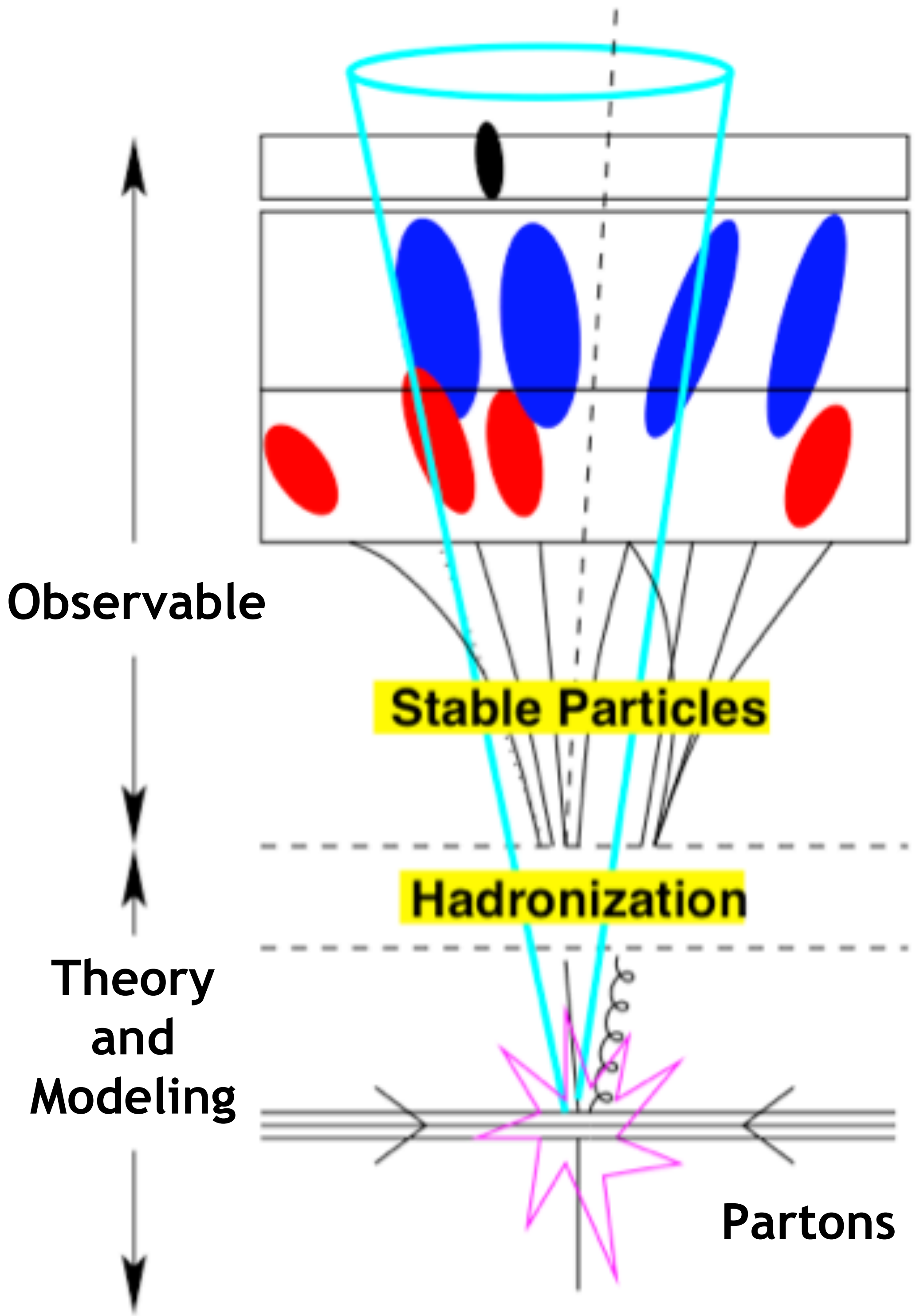
What are Jets?

- Due to the special nature of the strong force ("confinement"), **quarks & gluons** (partons) can not be observed directly
- Instead we observe collimated bunches of stable colorless hadrons, originating from partons after **fragmentation** and **hadronization**: **Jets!**
- Jet Finding is the **approximate** attempt to reverse-engineer the quantum mechanical processes of fragmentation and hadronization



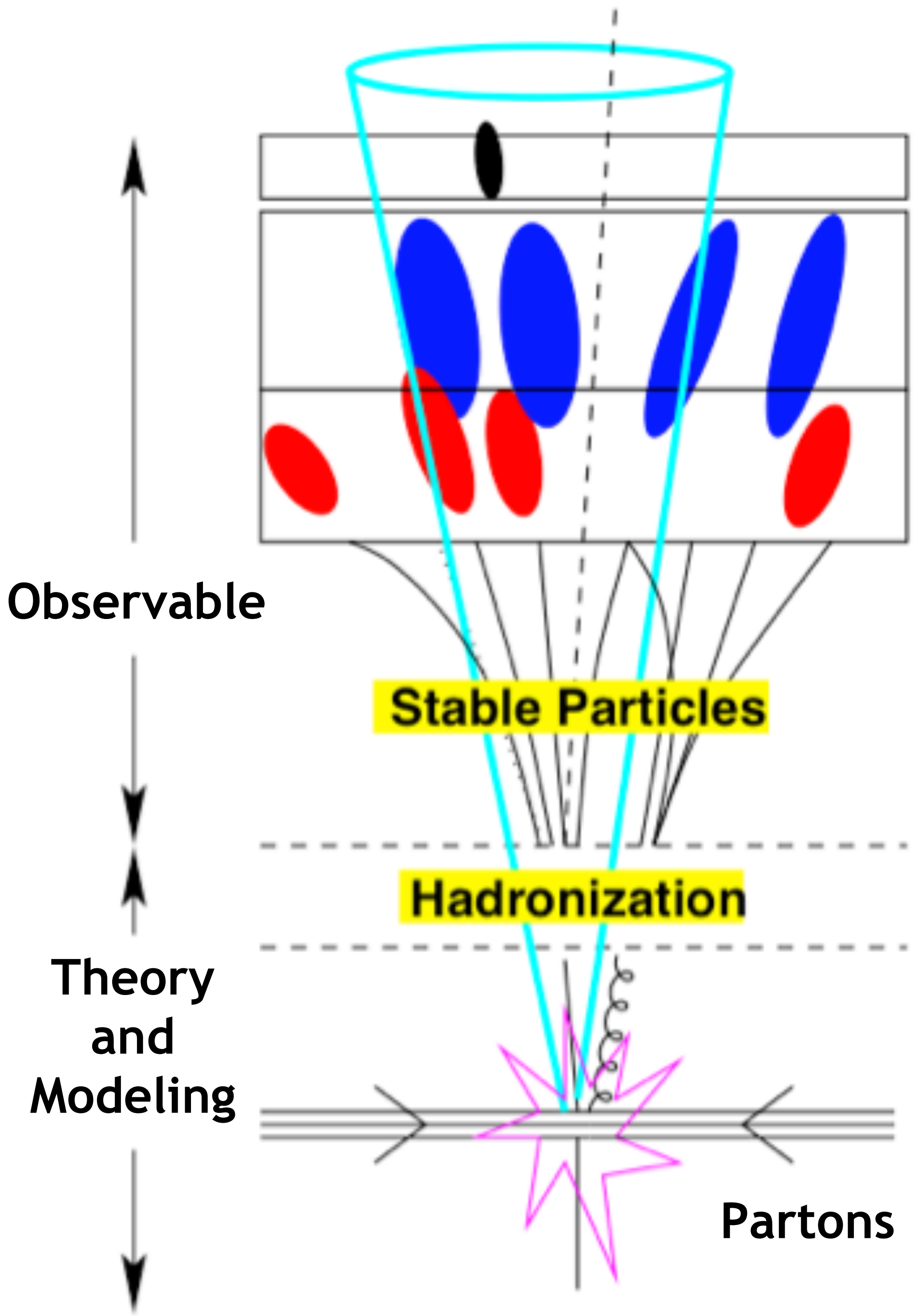
What are Jets?

- Due to the special nature of the strong force ("confinement"), **quarks & gluons** (partons) can not be observed directly
- Instead we observe collimated bunches of stable colorless hadrons, originating from partons after **fragmentation** and **hadronization**: **Jets!**
- Jet Finding is the **approximate** attempt to reverse-engineer the quantum mechanical processes of fragmentation and hadronization
- Fundamentally **ambigous** procedure!

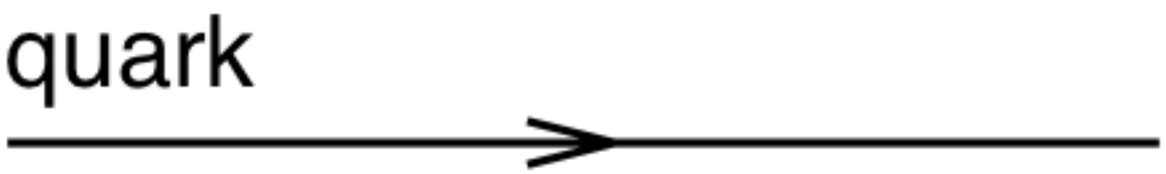


What are Jets?

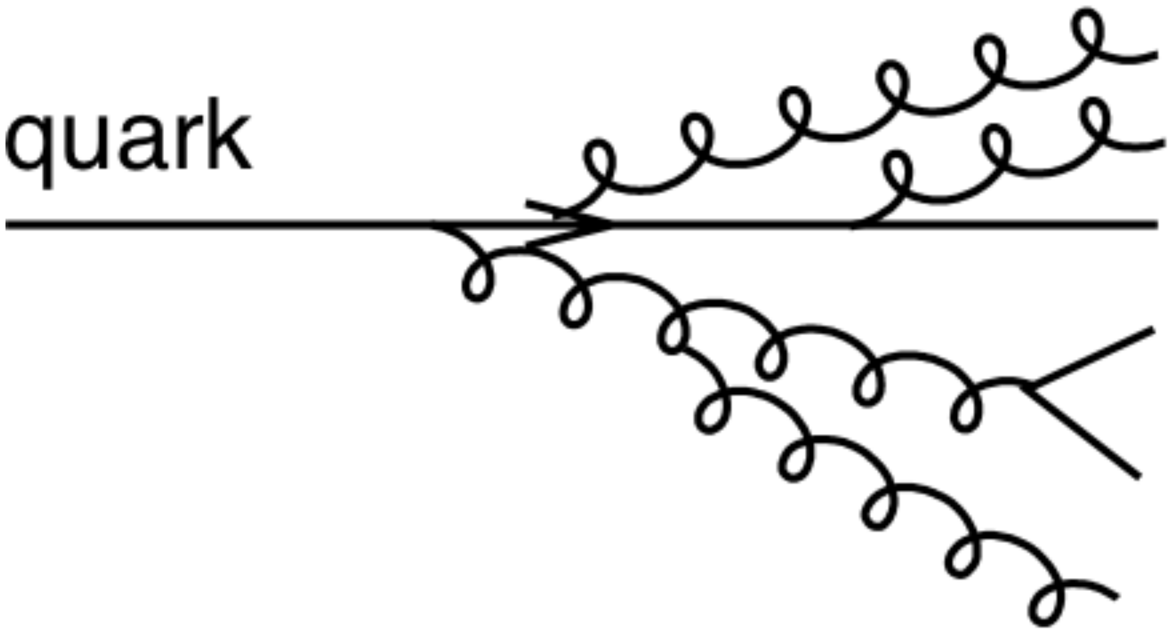
- Due to the special nature of the strong force ("confinement"), **quarks & gluons** (partons) can not be observed directly
- Instead we observe collimated bunches of stable colorless hadrons, originating from partons after **fragmentation** and **hadronization**: **Jets!**
- Jet Finding is the **approximate** attempt to reverse-engineer the quantum mechanical processes of fragmentation and hadronization
- Fundamentally **ambigous** procedure!
- Jets are the observable objects to relate experimental **observations** to theory **predictions** formulated in terms of quarks and gluons



Formation of Jets



Formation of Jets

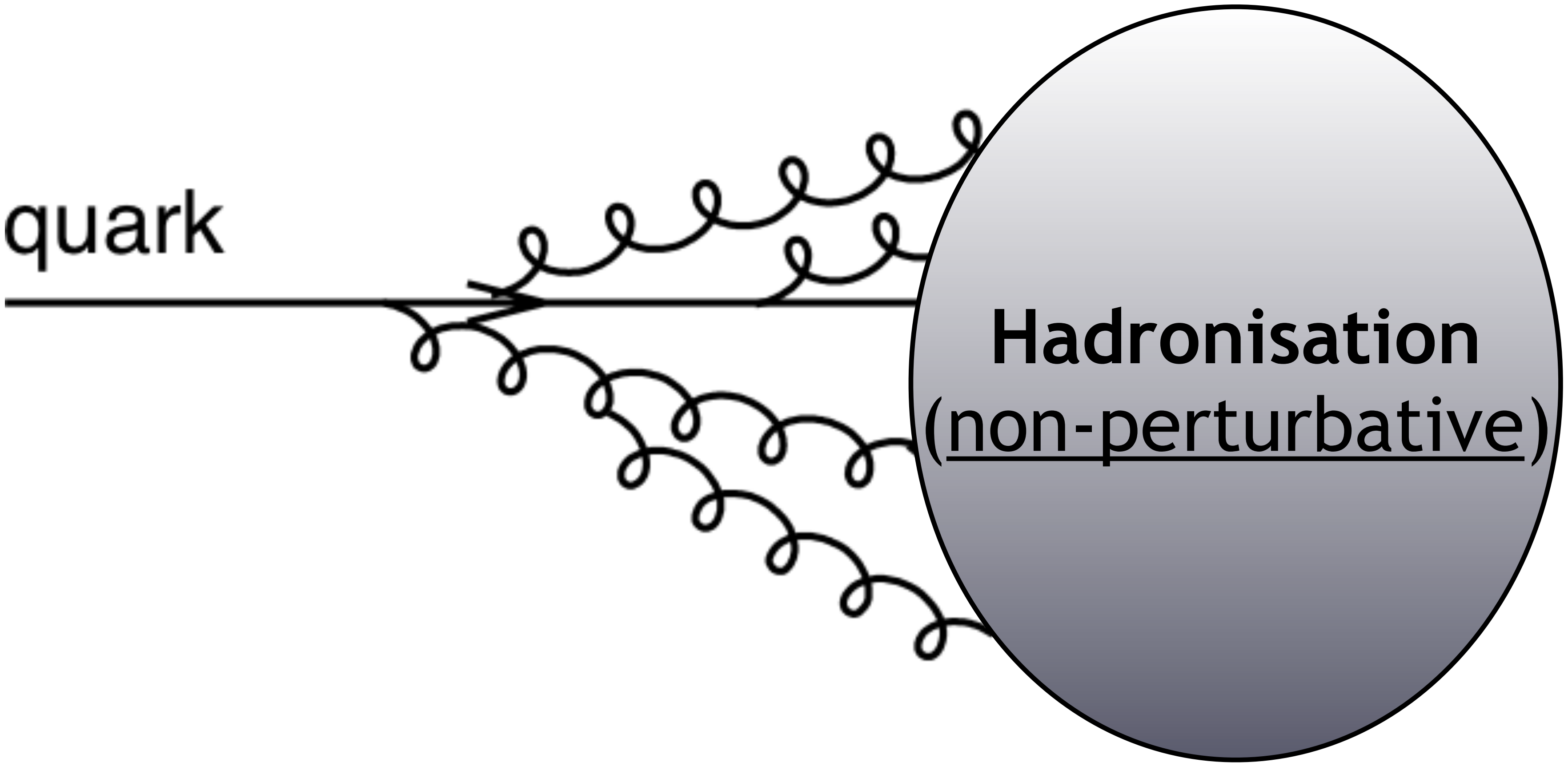


Fragmentation

perturbative gluon emissions

"*Parton Shower*"

Formation of Jets

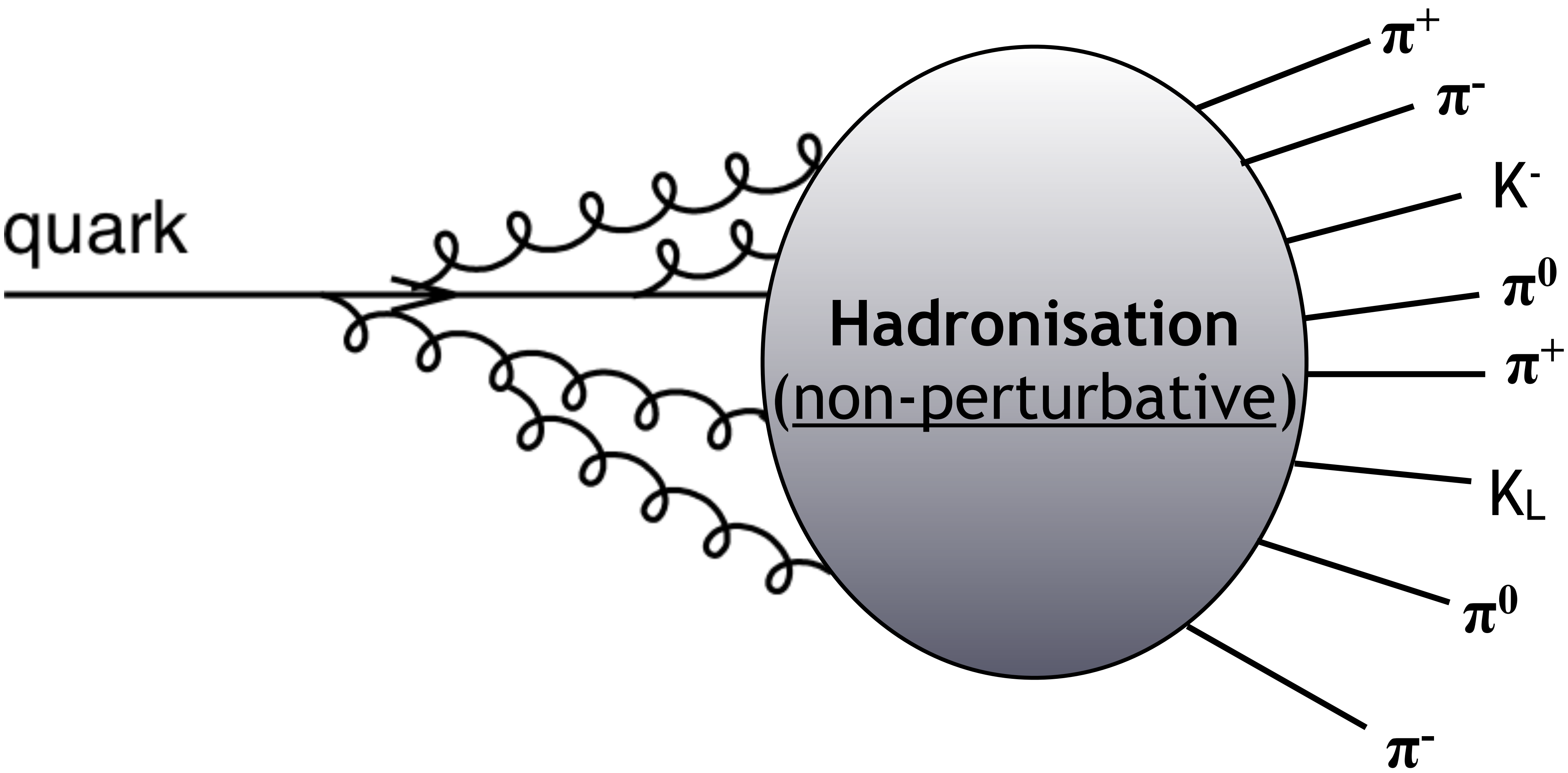


Fragmentation

perturbative gluon emissions

"Parton Shower"

Formation of Jets



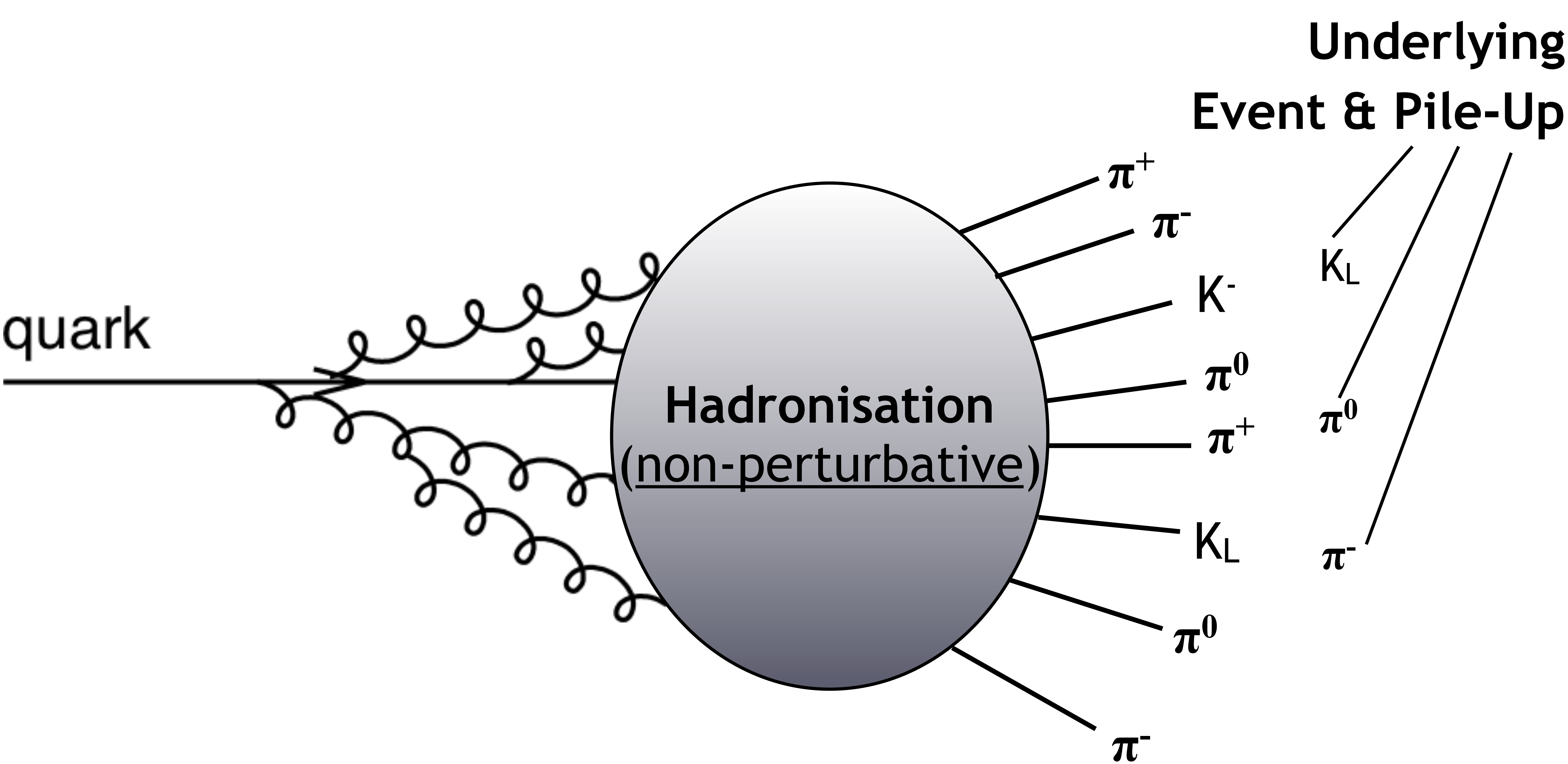
Fragmentation

perturbative gluon emissions

"*Parton Shower*"

Stable Particles

Formation of Jets



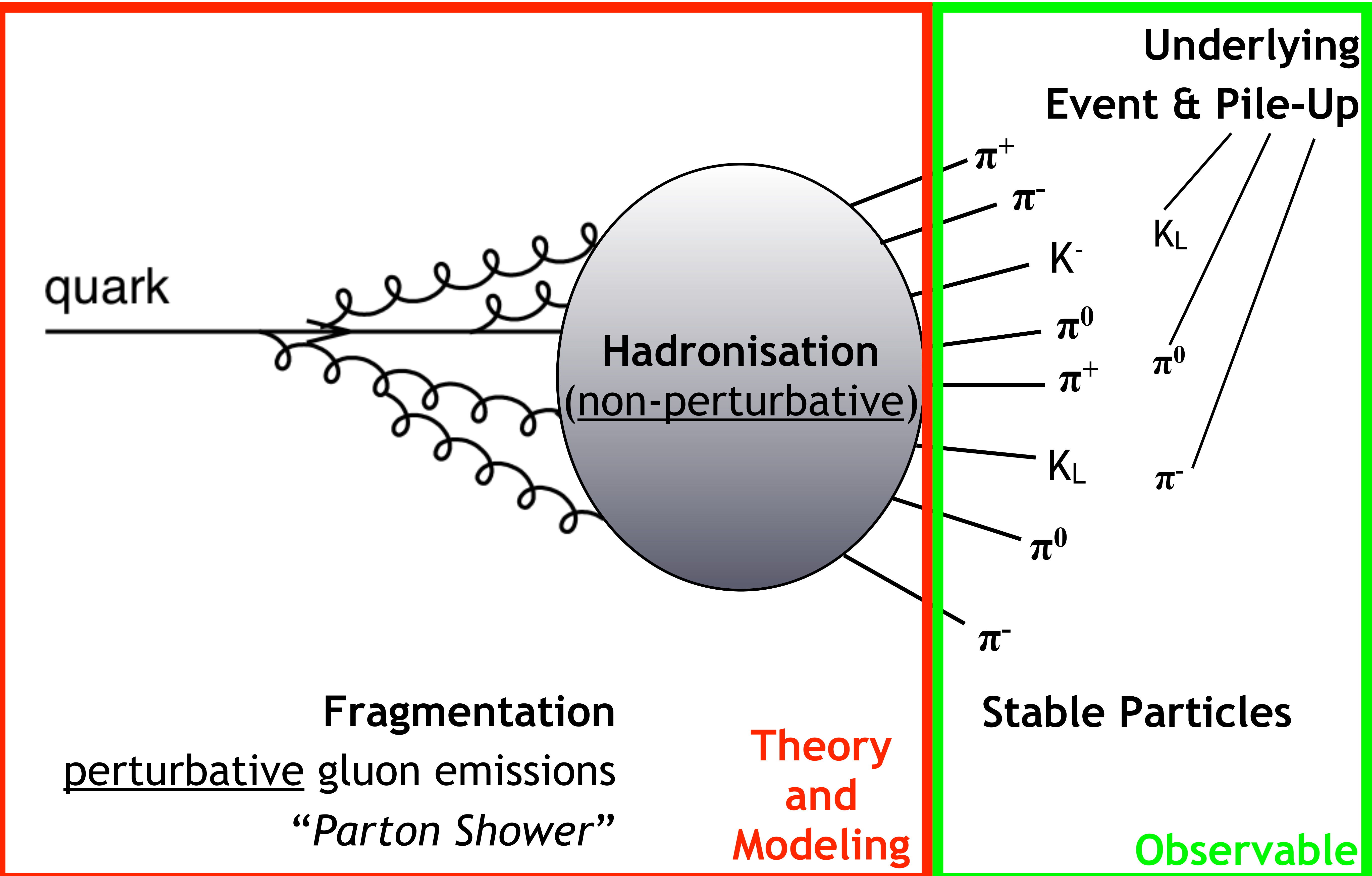
Fragmentation

perturbative gluon emissions

"Parton Shower"

Stable Particles

Formation of Jets



Underlying Event & Pile-Up

Underlying Event (UE)

- Multiple low-pT interactions which occur between the two hadron remnants in a pp-collisions
- Hadronisation and UE can not be unambigously separated, but affect jets differently
- Underlying Event activity at LHC **~15GeV per unit rapidity**
- MC generators simulate UE based on phenomenological models, which are tuned based on observations in real collisions

Pile-Up Interactions (PU)

- Additional minbias collisons overlayed with primary interaction
- In-Time PU: PU collisisons occur in same bunch crossing
- Out-of-Time PU: PU collisions occur in neighbouring bunches
- Depends on instantanous luminosity and bunch-spacing
- Easily **100GeV per unit rapidity** at high-lumi / low bunch-spacing

Jet Algorithms

A **Jet Algorithm** is a well-defined procedure which transforms a set of input particles $\{p_i\}$ into a set of jets $\{j_k\}$

$$\{p_i\} \xrightarrow[\text{algorithm}]{\text{jet}} \{j_k\}$$

Particles:

four-vectors from different stages of the formation of jets are transformed into corresponding **jet types**:

- partons -> **parton jets**
- stable particles -> **generator jets**
- calorimeter towers
  PF particles
  topological clusters -> **reco jets**
  tracks
  ...

Jet Algorithm Parameters:

- Typically at least **"Radius" R**
- Additional Parameters e.g. for Split/Merge procedures

Recombination Scheme:

- Defines how the jet four-momentum is calculated from its constituent particles
- E-Scheme: jet four-momentum defined to be four-vector sum of constituent particles

Theoretical & Experimental Jet Algorithms Requirements

Parton-/Hadron-/Detector-Level

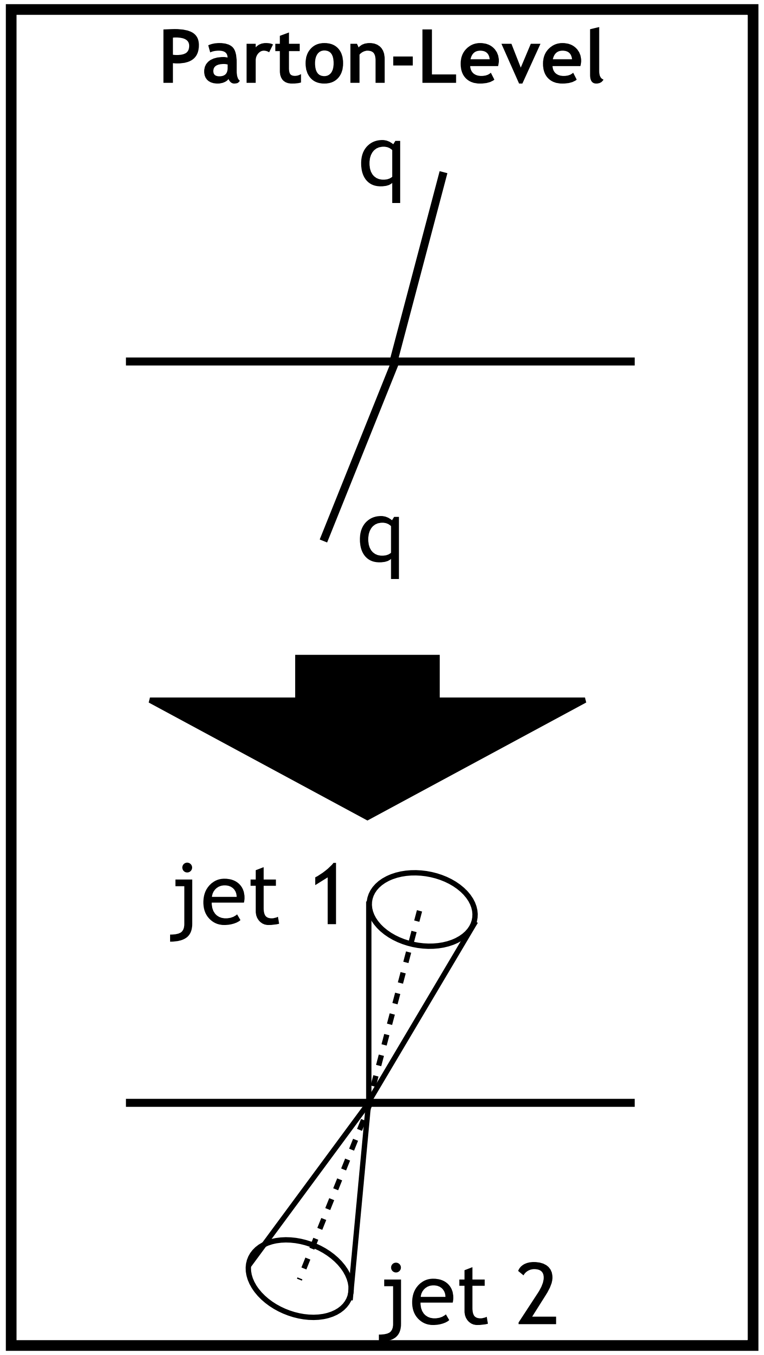
The projections of an event at different stages of the jet formation should be coherent with each other

(Reminder: *"Jets are the observable objects to relate experimental observations to theory predictions formulated in terms of quarks and gluons"*)

Parton-/Hadron-/Detector-Level

The projections of an event at different stages of the jet formation should be coherent with each other

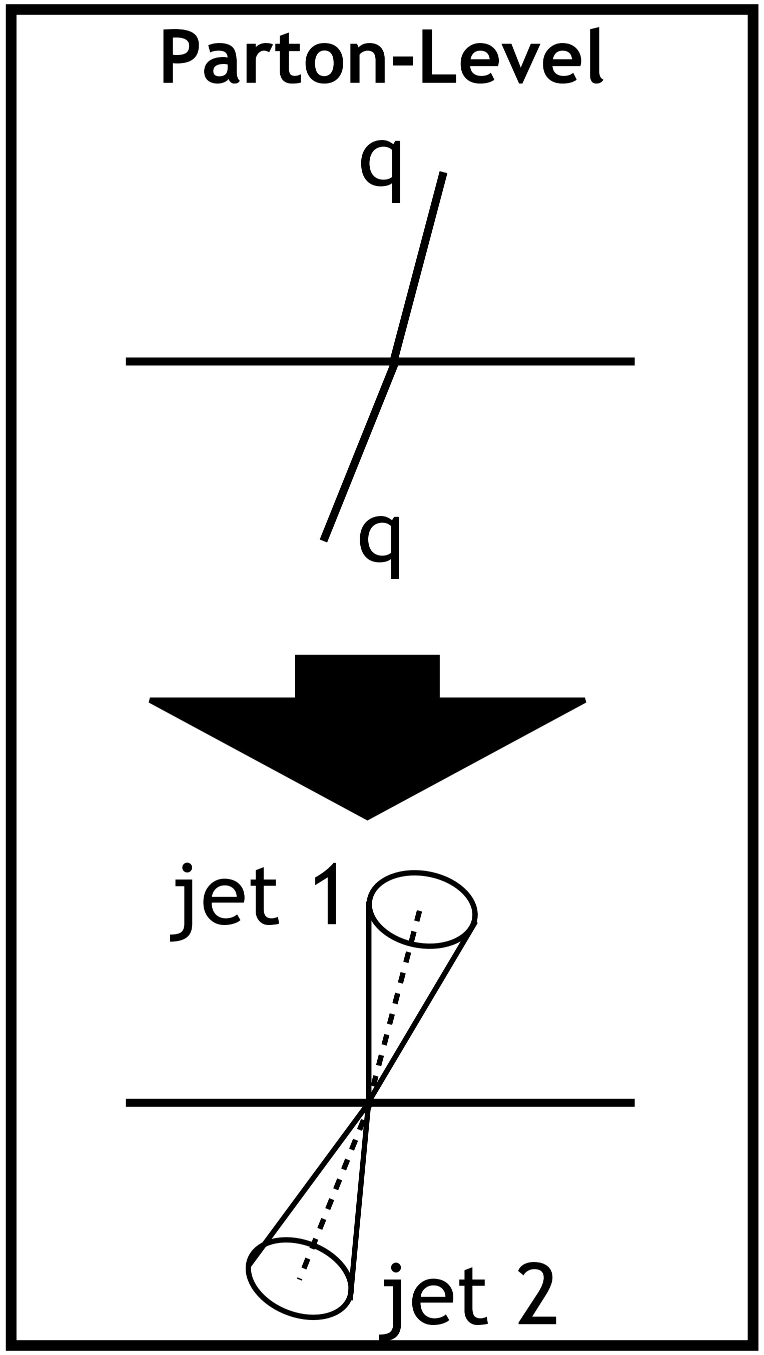
(Reminder: *"Jets are the observable objects to relate experimental observations to theory predictions formulated in terms of quarks and gluons"*)

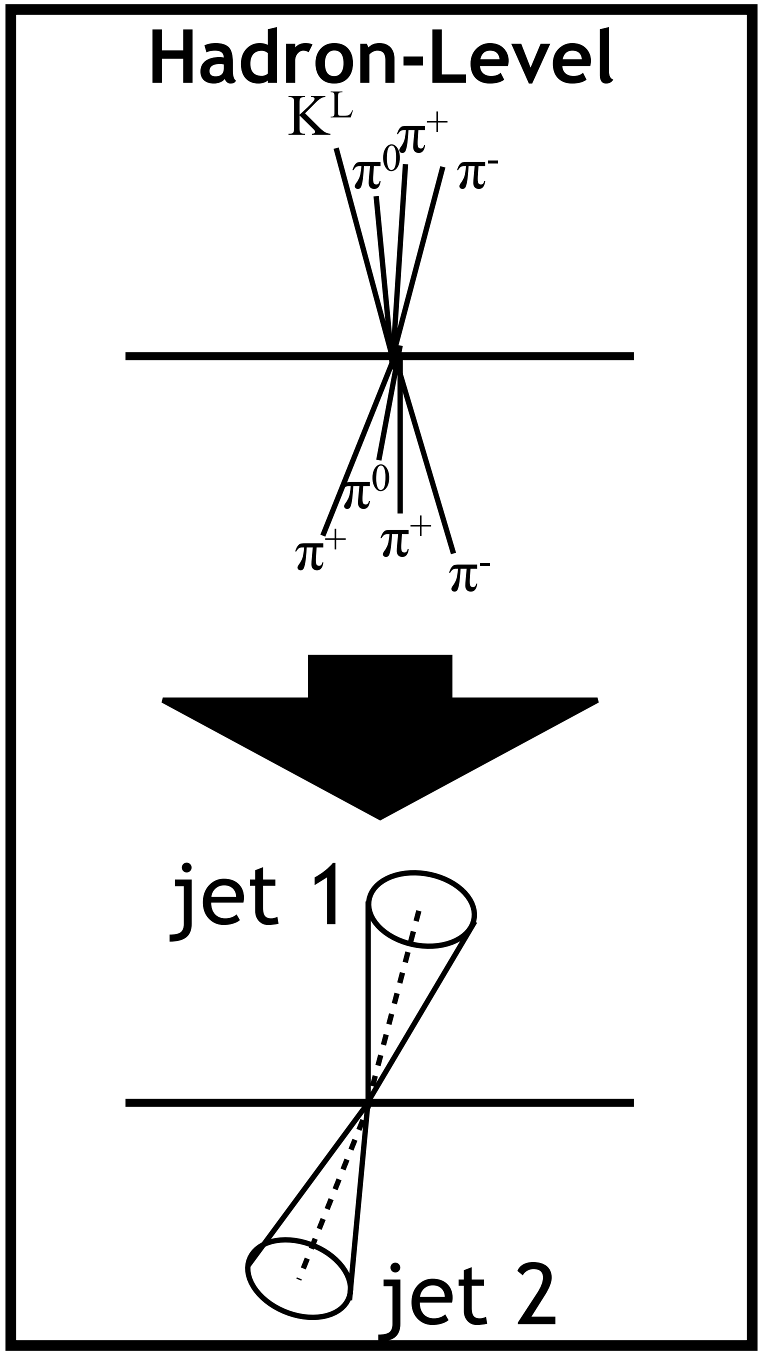


Parton-/Hadron-/Detector-Level

The projections of an event at different stages of the jet formation should be coherent with each other

(Reminder: *"Jets are the observable objects to relate experimental observations to theory predictions formulated in terms of quarks and gluons"*)



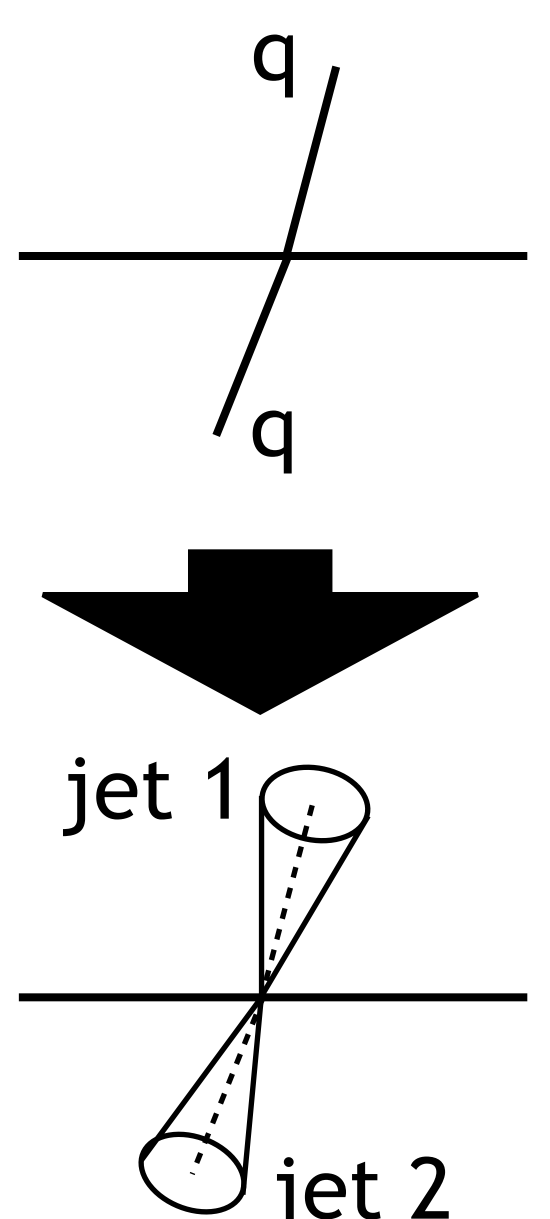


Parton-/Hadron-/Detector-Level

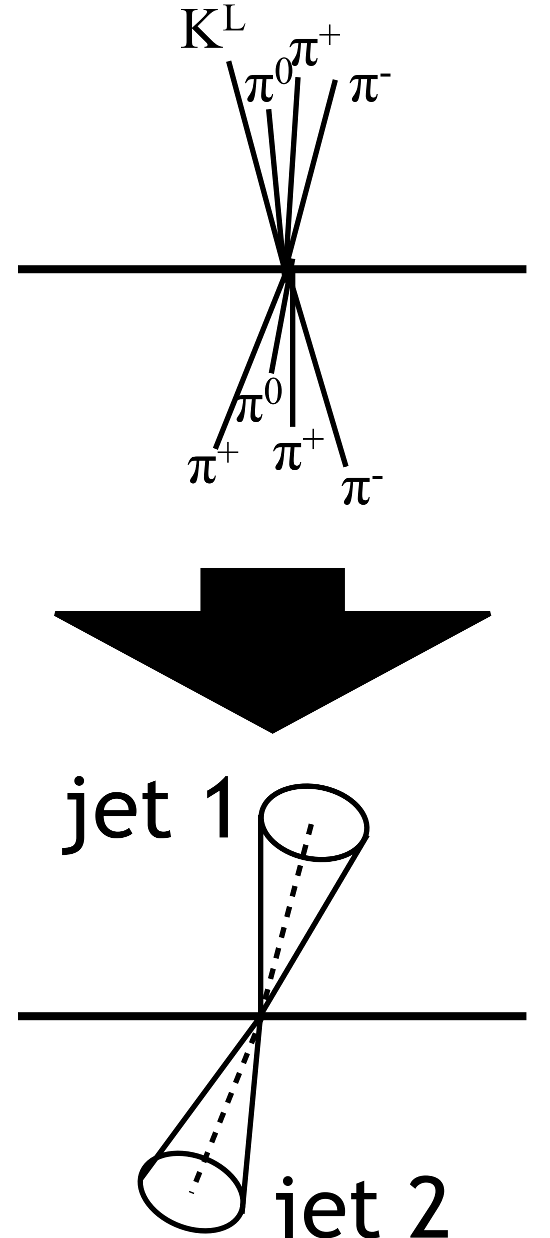
The projections of an event at different stages of the jet formation should be coherent with each other

(Reminder: *"Jets are the observable objects to relate experimental observations to theory predictions formulated in terms of quarks and gluons"*)

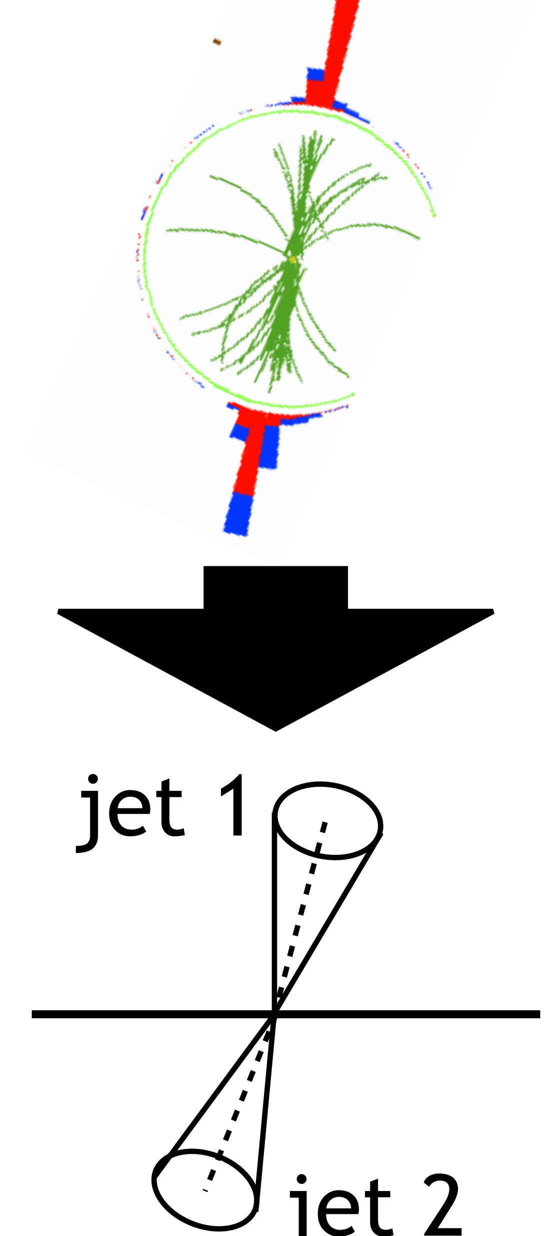
Parton-Level



Hadron-Level



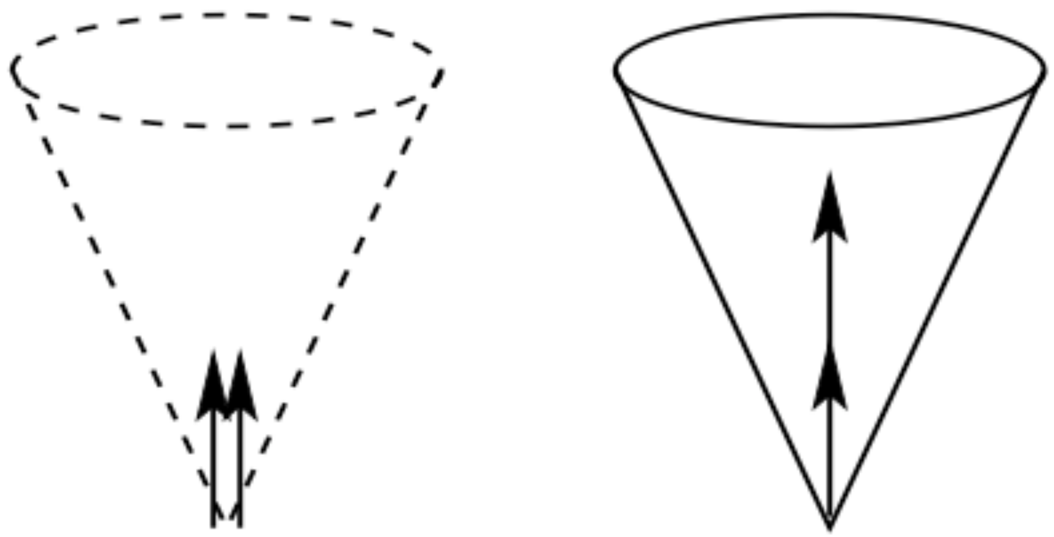
Detector-Level



Infrafred- and Collinear-Safety

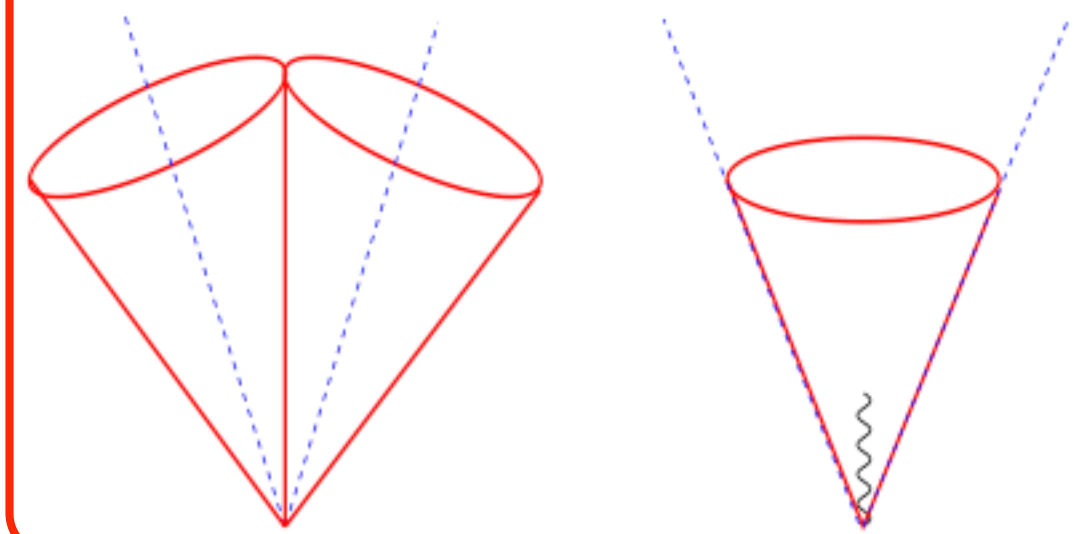
Collinear Safety

Collinear Splittings
should not change jets



Infrared-Safety

Soft gluon emissions
should not change jets



arXiv:0704.0292

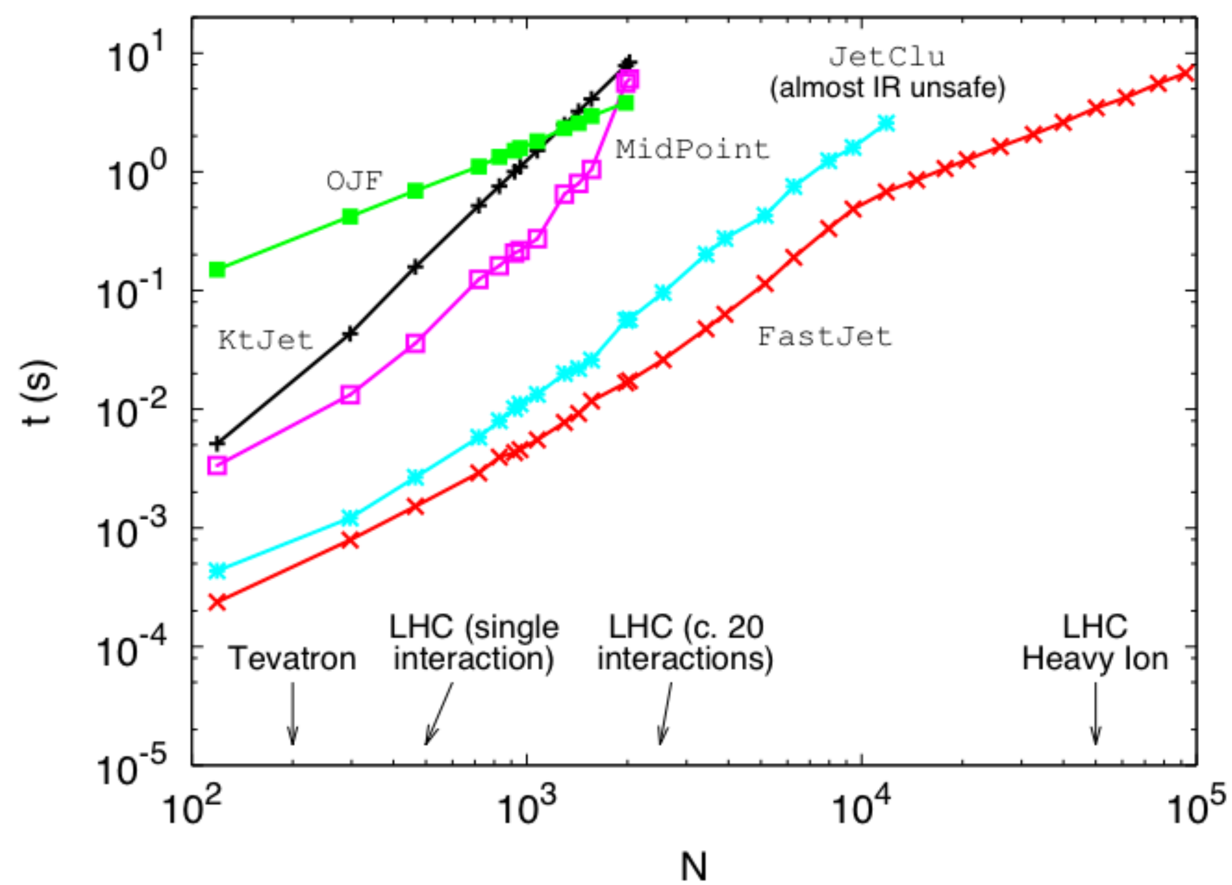
"Infrared unsafety is a serious issue, not just because it makes impossible to carry out meaningful (finite) perturbative calcuations, but also because it breaks the whole relation betweeb the (Born or low-order) partonic structure of the event and the jets that one observes, and it is precisly this relation that a jet algorithm is supposed to codify: it makes no sense for the structure of multi-hundred GeV jets to change radically just because hadronisation, the underlying event or pileup threw a 1 GeV particle in between them."

Experimental Considerations

- **Good Jet Energy Resolution**
- **Resiliancy to Detector Effects, like**
  - input particle (e.g. calorimeter tower) thresholds
  - bending of charged particles in the magnetic field
  - detector noise

  ➡ **These effects are very related to and partially addressed by IR&C-Safety!**

- **Good computational performance**
  - Sequential clustering algorithms were originally disfavored at hadron colliders because of poor computational performance for large numbers of particles (N^3)
  - This concern was once and for all addressed by the fastjet [1,2] package
  - Both ATLAS & CMS rely on fastjet for (almost) all their jet clustering needs



[1] http://www.lpthe.jussieu.fr/~salam/fastjet/

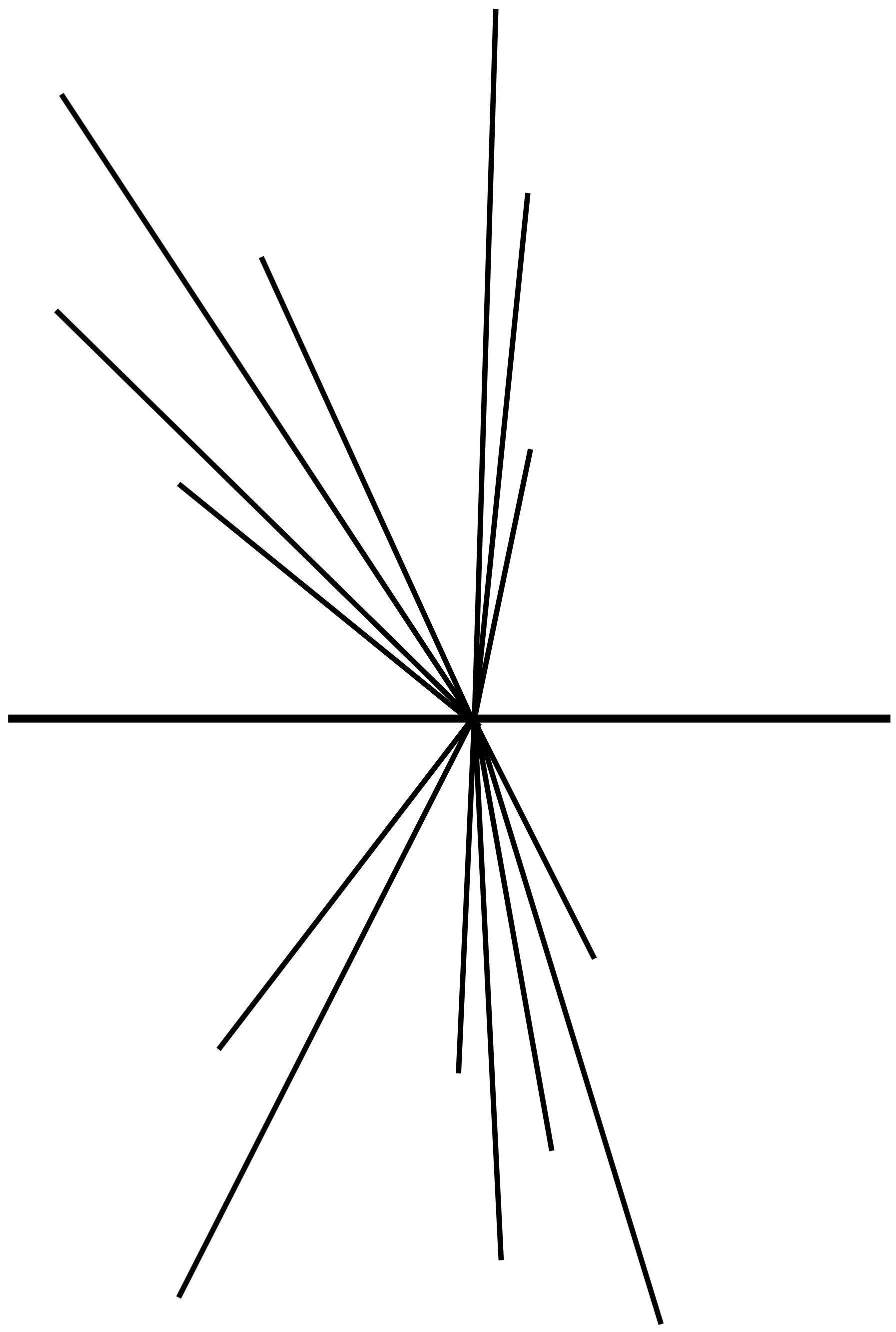
[2] http://arxiv.org/abs/hep-ph/0512210

Two classes of jet algorithms:

(1) Cone Algorithms
(2) Sequential Clustering Algorithms

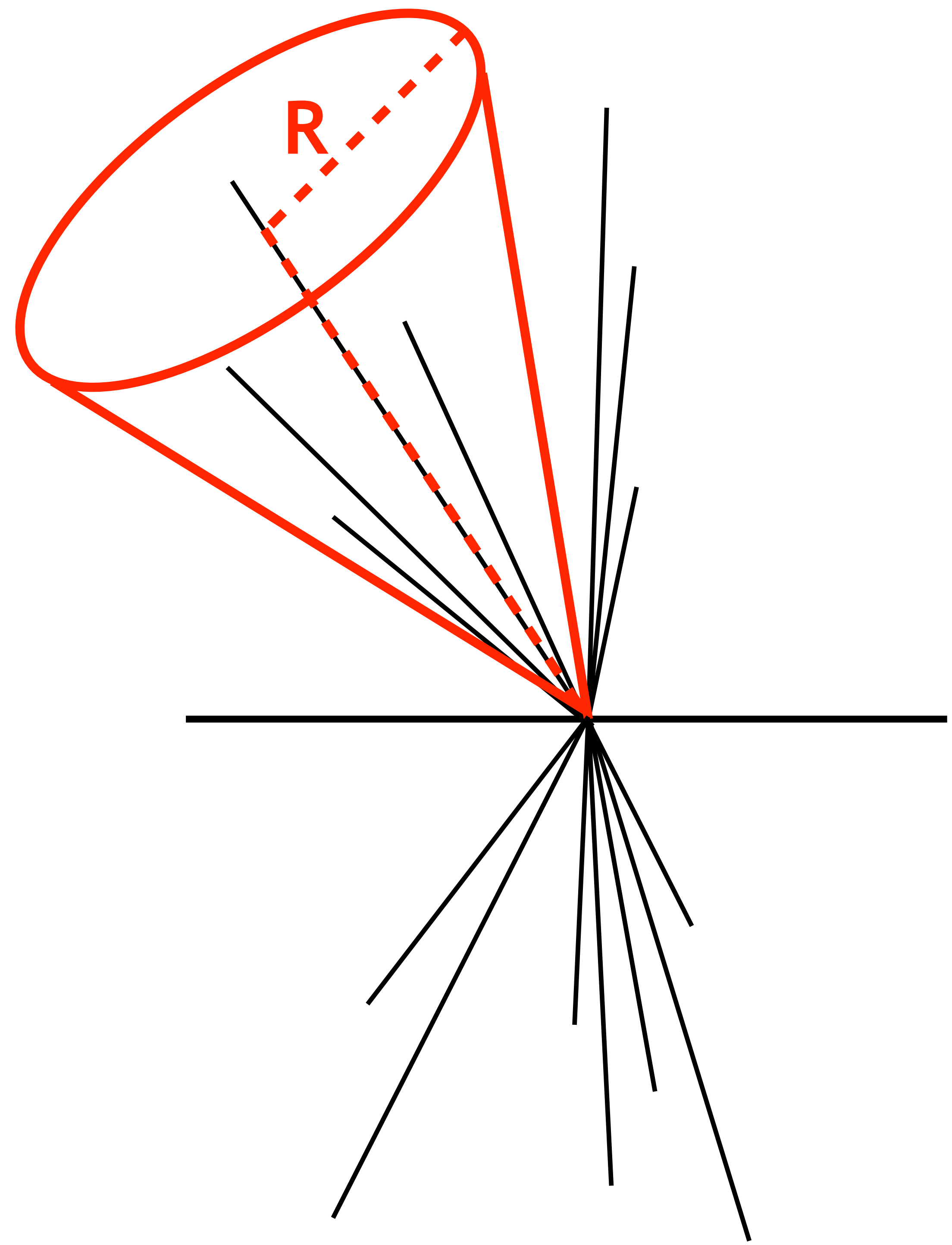
Cone Algorithms

- **“top-down”** approach



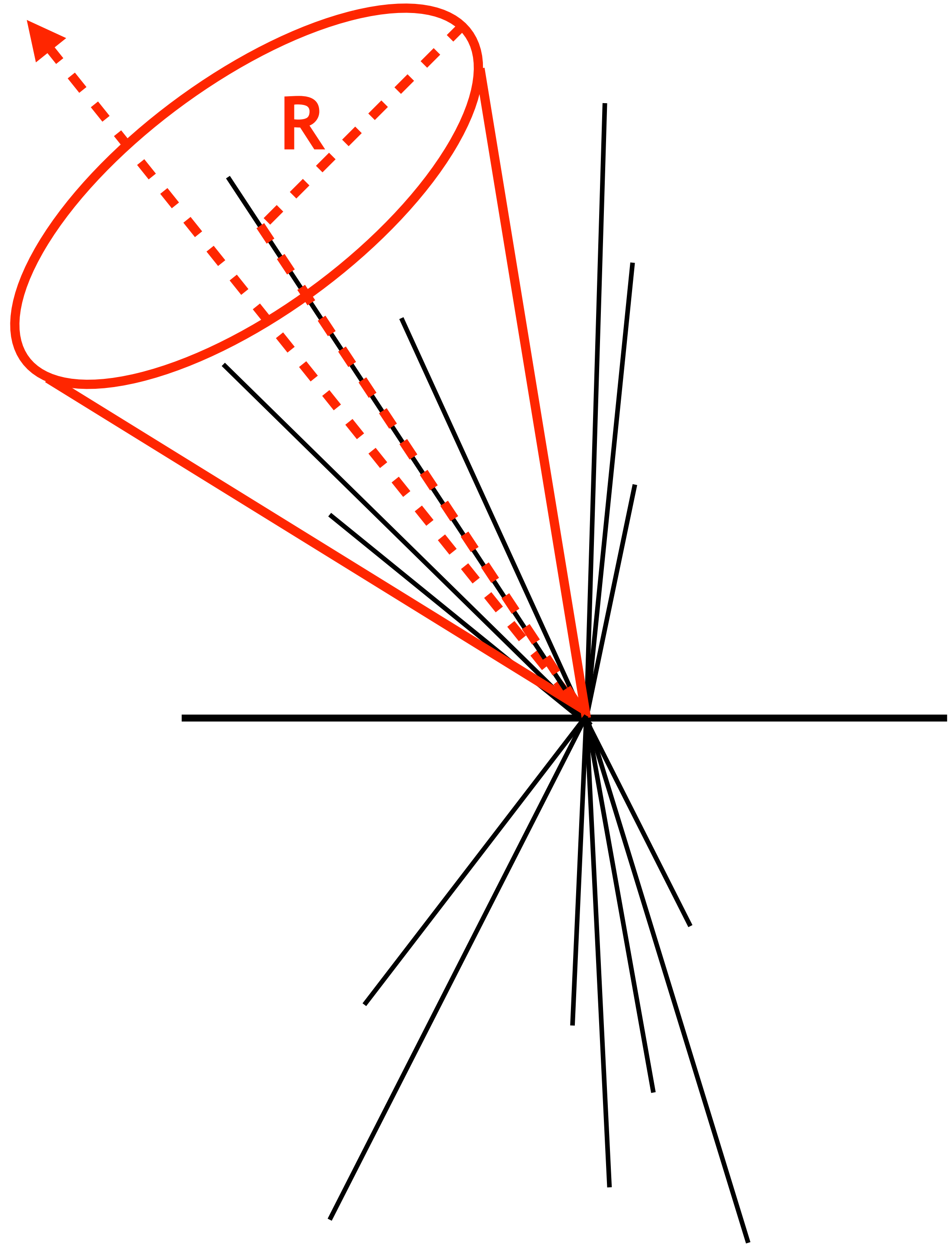
Cone Algorithms

- **“top-down”** approach
- Typically **Seeded:** consider only a subset of particles (above some pT threshold) as seeds



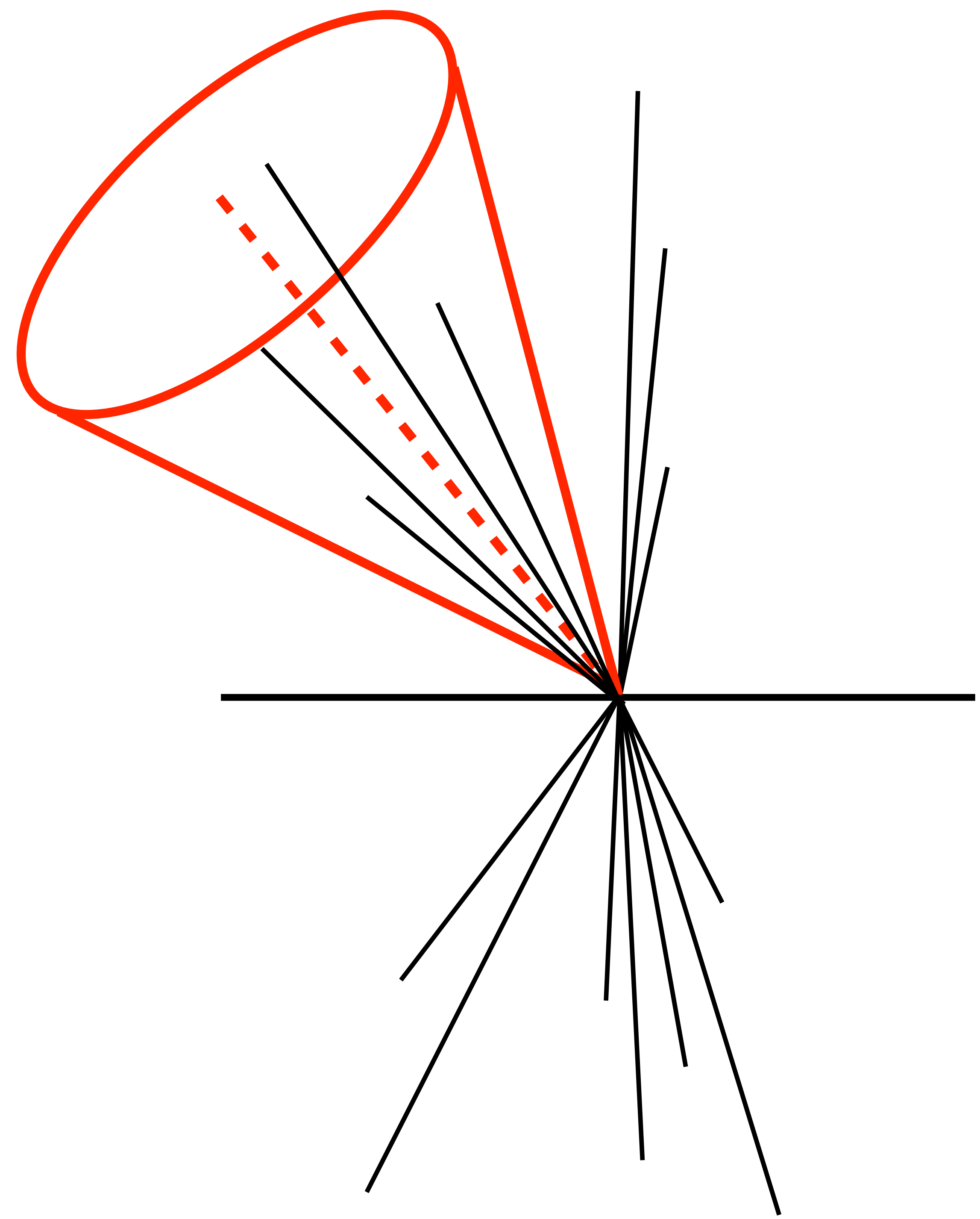
Cone Algorithms

- **"top-down"** approach
- Typically **Seeded:** consider only a subset of particles (above some pT threshold) as seeds
- Sum up all particle four-vectors contained in a cone of **radius R = $\sqrt{\Delta y^2+\Delta\varphi^2}$** around the seed direction



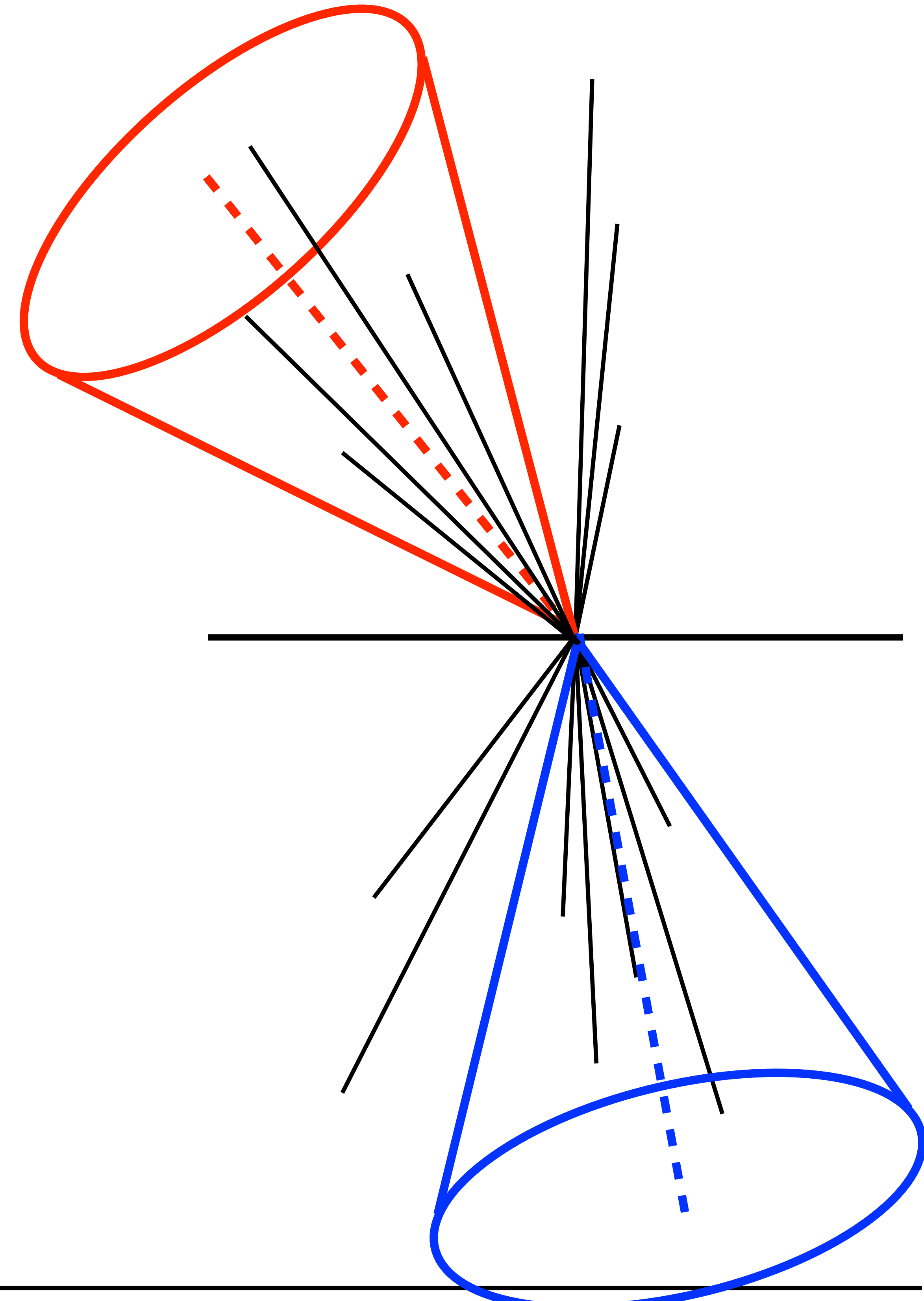
Cone Algorithms

- **"top-down"** approach
- Typically **Seeded:** consider only a subset of particles (above some pT threshold) as seeds
- Sum up all particle four-vectors contained in a cone of **radius** $\mathbf{R = \sqrt{\Delta y^2+\Delta\varphi^2}}$ around the seed direction
- Iterate until **"stable cone"** is found: four-vector sum direction of contained particles corresponds to cone center



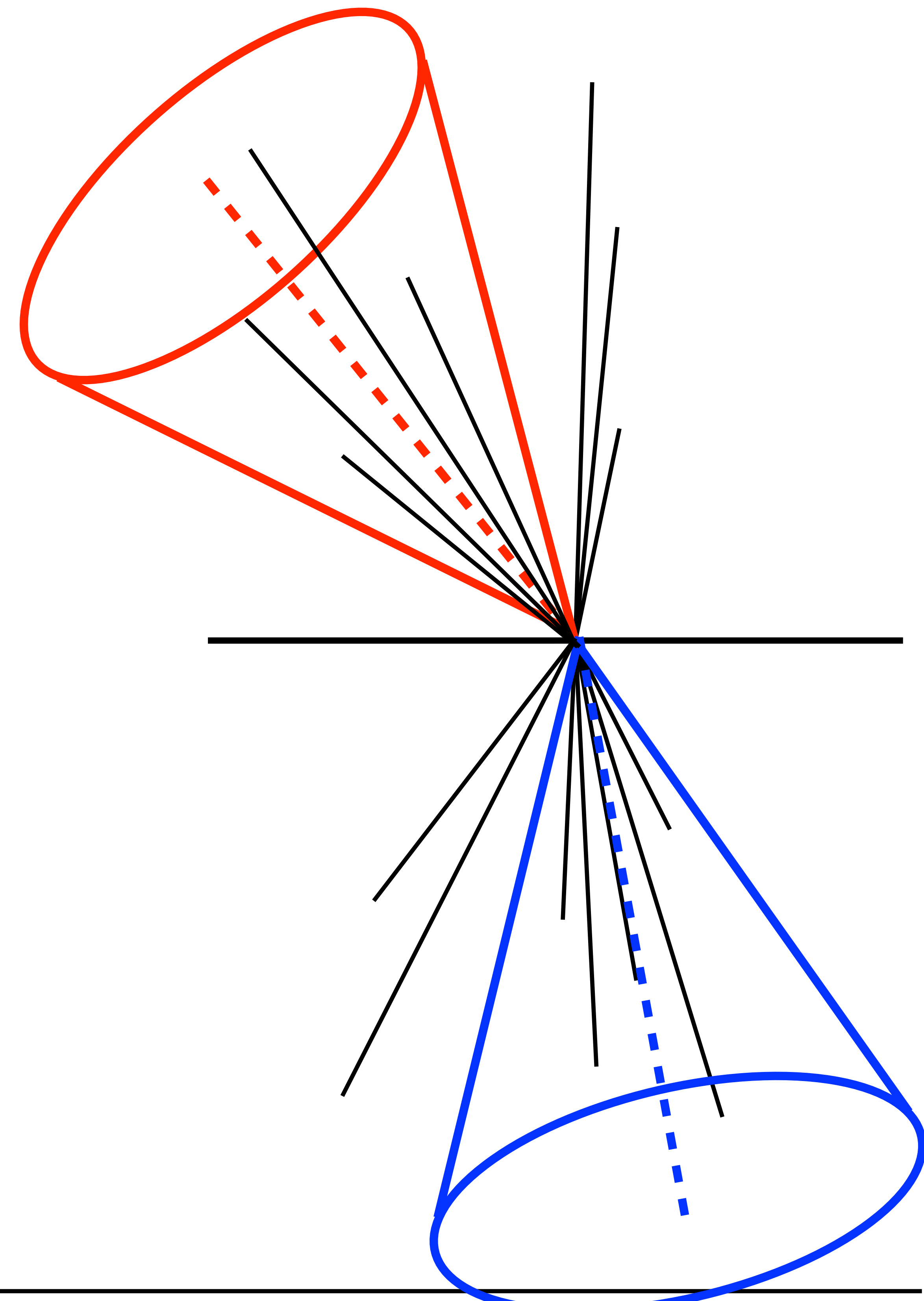
Cone Algorithms

- **"top-down"** approach
- Typically **Seeded:** consider only a subset of particles (above some pT threshold) as seeds
- Sum up all particle four-vectors contained in a cone of **radius** $\mathbf{R = \sqrt{\Delta y^2 + \Delta\varphi^2}}$ around the seed direction
- Iterate until **"stable cone"** is found: four-vector sum direction of contained particles corresponds to cone center



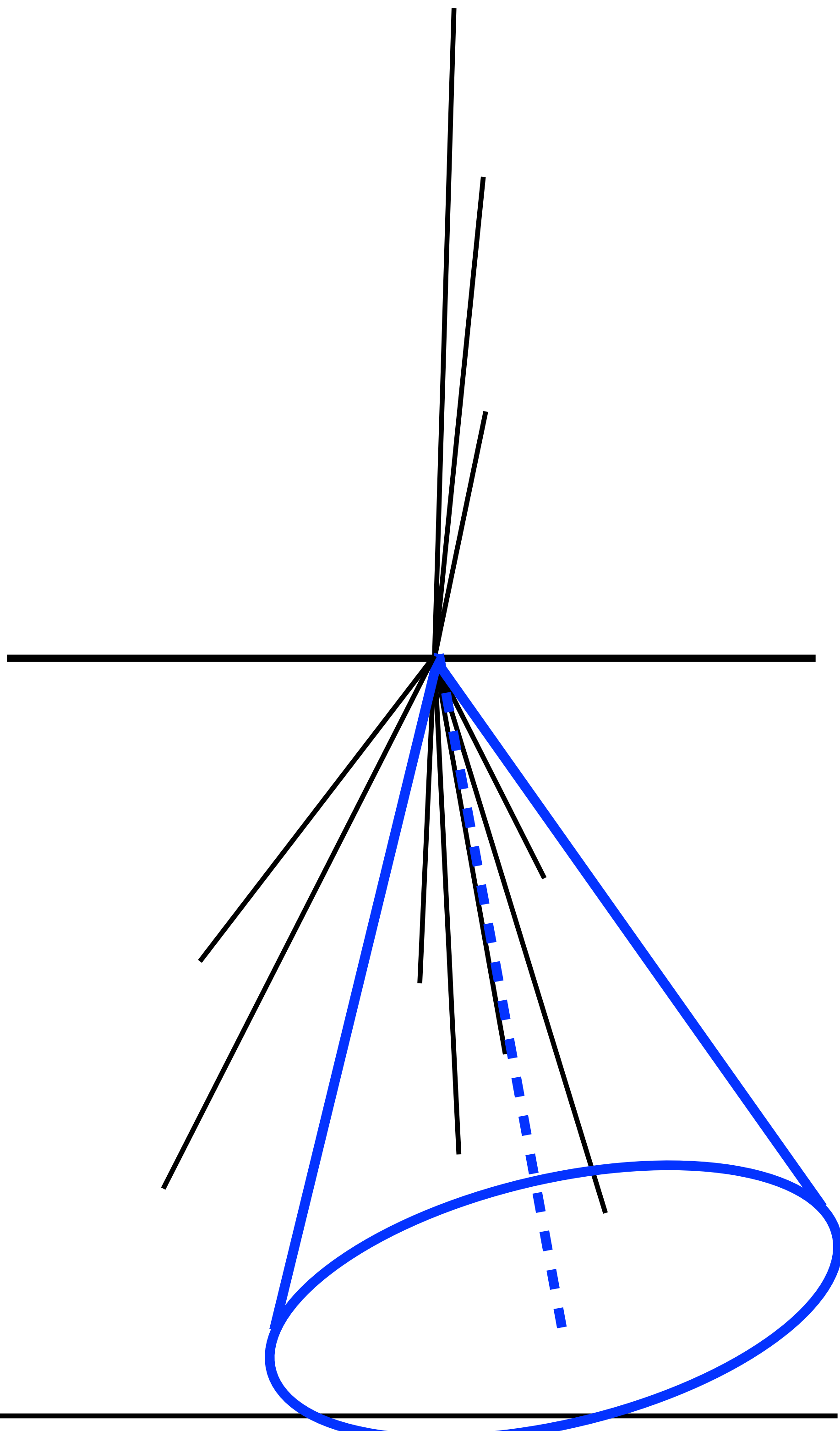
Cone Algorithms

- "**top-down**" approach
- Typically **Seeded**: consider only a subset of particles (above some pT threshold) as seeds
- Sum up all particle four-vectors contained in a cone of **radius** $\mathbf{R = \sqrt{\Delta y^2+\Delta\varphi^2}}$ around the seed direction
- Iterate until "**stable cone**" is found: four-vector sum direction of contained particles corresponds to cone center
- Different Strategies to handle **overlaps**:



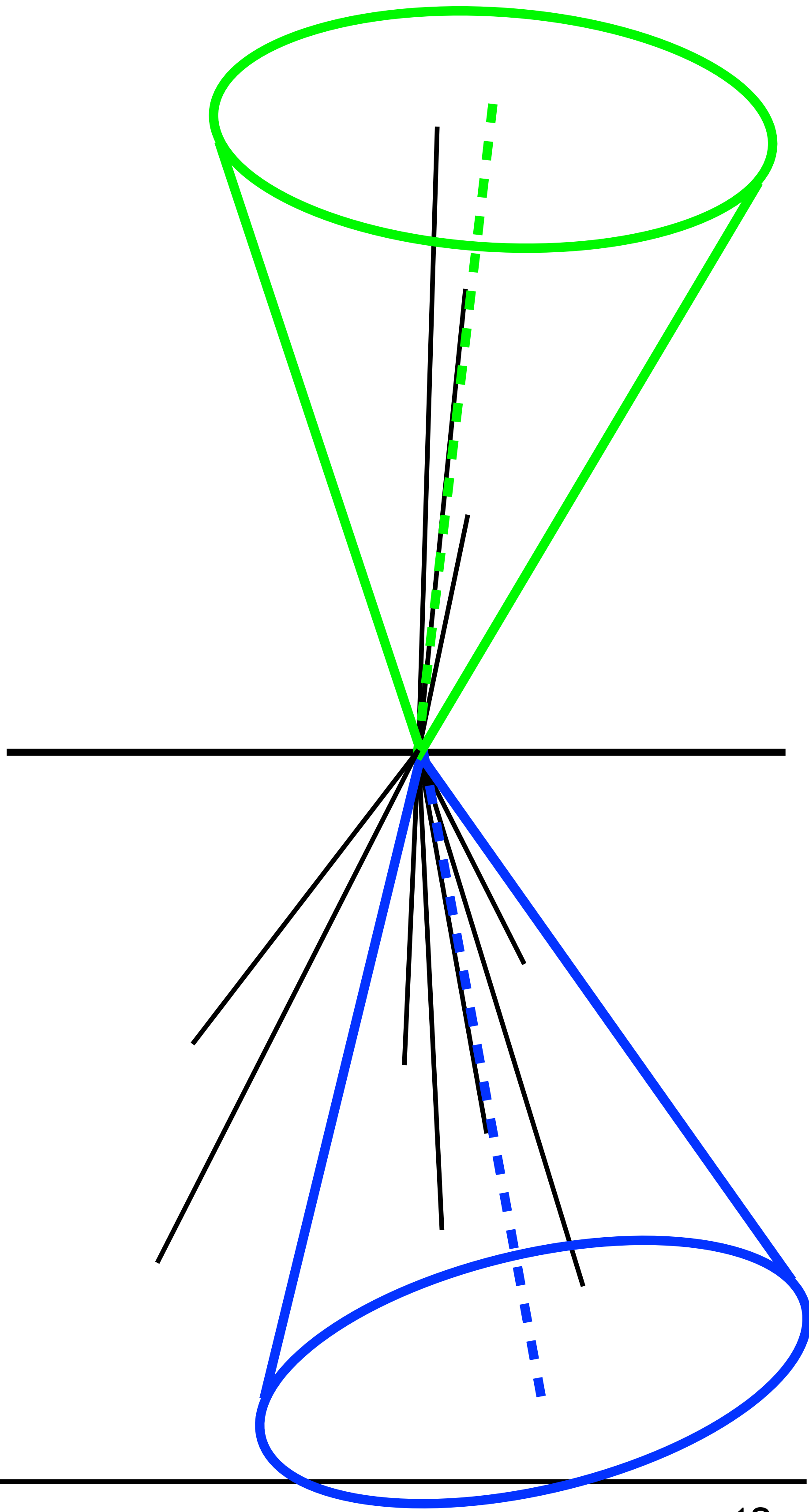
Cone Algorithms

- "**top-down**" approach
- Typically **Seeded**: consider only a subset of particles (above some pT threshold) as seeds
- Sum up all particle four-vectors contained in a cone of **radius** $\mathbf{R = \sqrt{\Delta y^2+\Delta\varphi^2}}$ around the seed direction
- Iterate until "**stable cone**" is found: four-vector sum direction of contained particles corresponds to cone center
- Different Strategies to handle **overlaps**:
  - "cooky cutter": particles belonging to reconstructed jet removed (*CMS IterativeCone*)



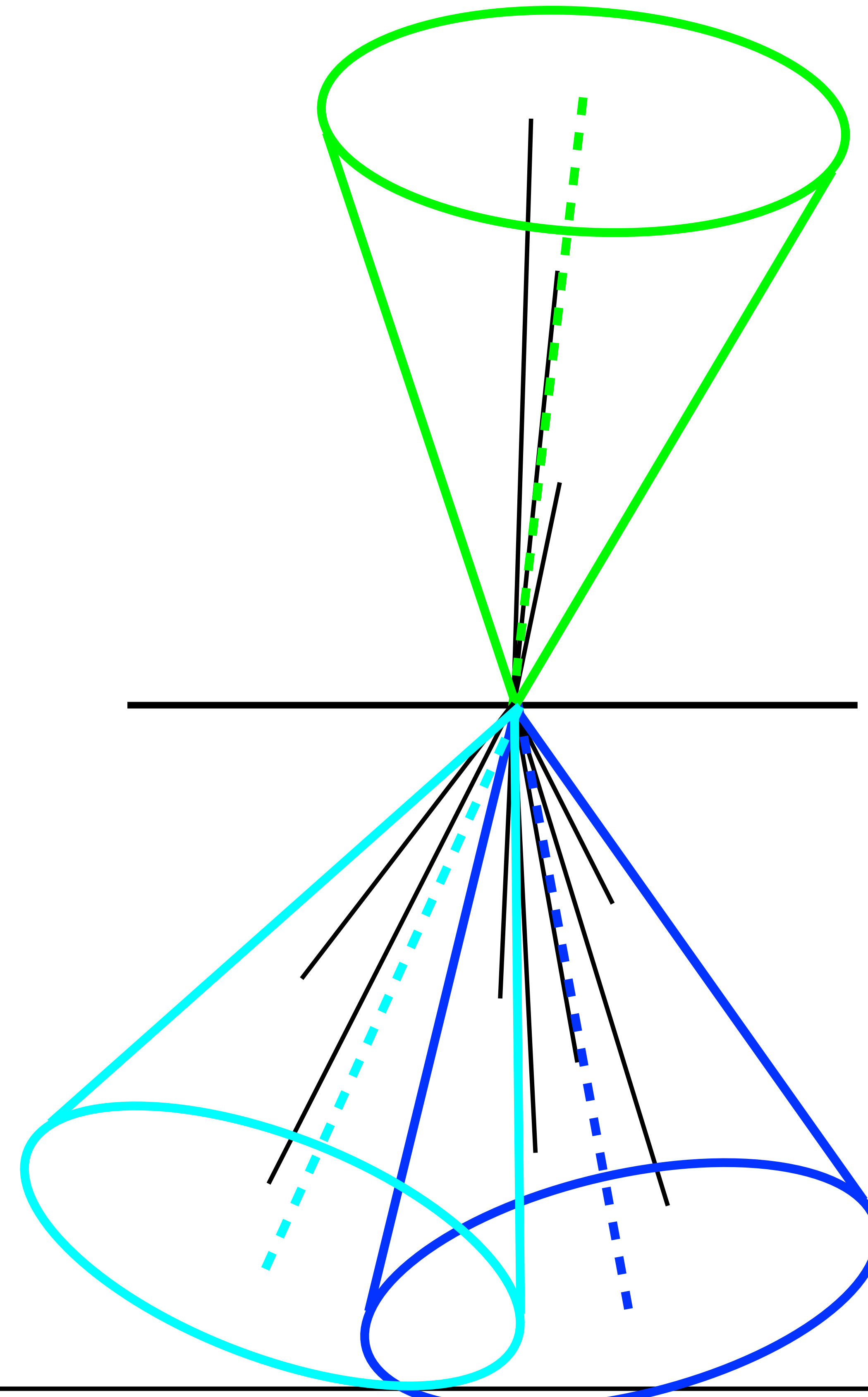
Cone Algorithms

- "**top-down**" approach
- Typically **Seeded**: consider only a subset of particles (above some pT threshold) as seeds
- Sum up all particle four-vectors contained in a cone of **radius R = $\sqrt{\Delta y^2+\Delta\varphi^2}$** around the seed direction
- Iterate until "**stable cone**" is found: four-vector sum direction of contained particles corresponds to cone center
- Different Strategies to handle **overlaps**:
  - "cooky cutter": particles belonging to reconstructed jet removed (*CMS IterativeCone*)



Cone Algorithms

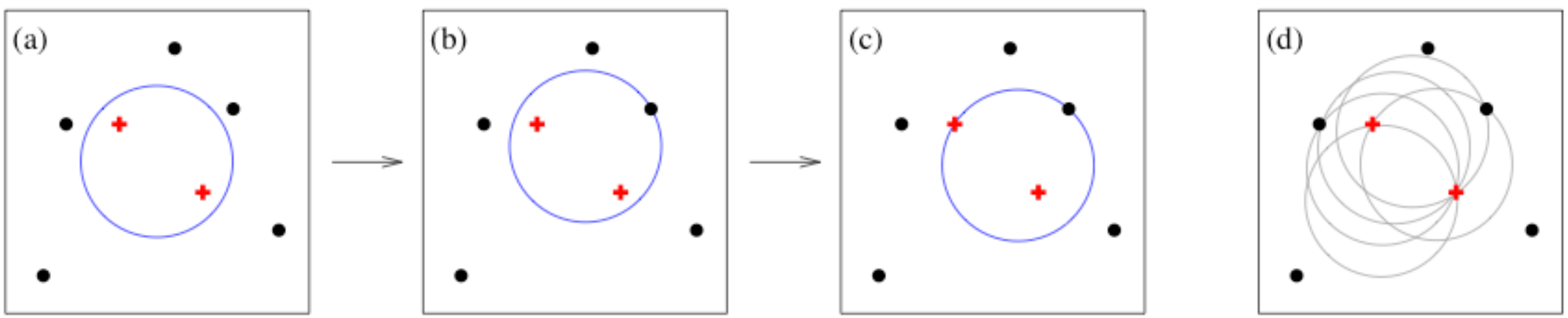
- "**top-down**" approach
- Typically **Seeded**: consider only a subset of particles (above some pT threshold) as seeds
- Sum up all particle four-vectors contained in a cone of **radius** $\mathbf{R = \sqrt{\Delta y^2+\Delta\varphi^2}}$ around the seed direction
- Iterate until "**stable cone**" is found: four-vector sum direction of contained particles corresponds to cone center
- Different Strategies to handle **overlaps**:
  - "cooky cutter": particles belonging to reconstructed jet removed (*CMS IterativeCone*)
  - "split&merge": particles are allowed to appear in several jets, split & merge procedure to determine afterwards which particle belongs to which jet or whether two jets are to be merged (*Midpoint*, *ATLAS*)



SISCone

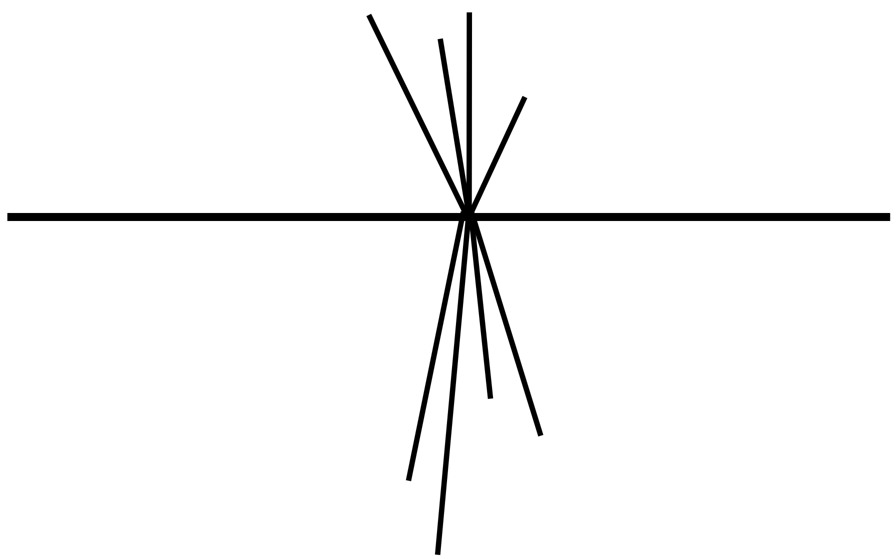
- "Seedless Infrared-Safe Cone" Algorithms
- **Exact seedless** cone algorithm which provably finds all stable cones
- Only existing **IR&C-safe** cone algorithm!
- Employs **Split&Merge** to resolve overlaps
- Reasonable computational performance
  - previous seedless exact cone algorithm formulation: $\sim N2^N$ (10^{17} years for N=100!!)
  - SISCone: $\mathbf{N^2 \ln(N)}$

➡ **still: removed from CMS standard reco-chain due to excessive processing time**



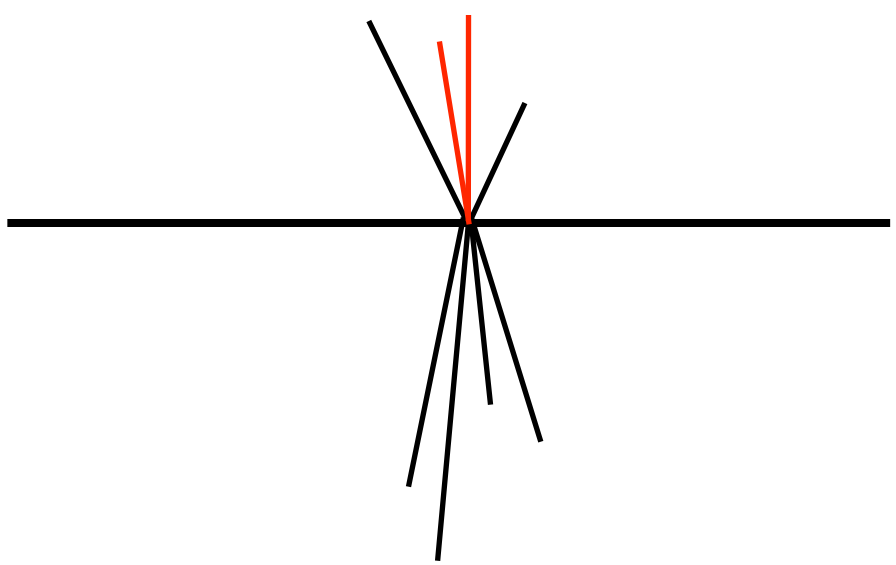
2D Simplification: Moving (a) initial circular enclosure in a random direction until some **particle** (b) touches the circle, then pivot the circle around that edge point until (c) a second point touches the edge. (d) all circles defined by pairs of edge points are all stable cones

Sequential Clustering Algorithms



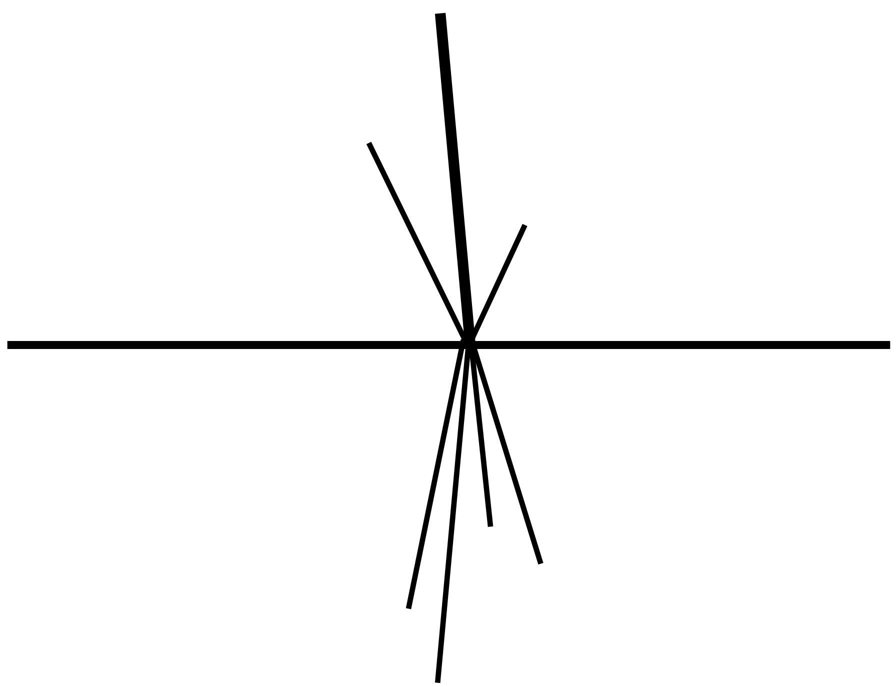
- "**bottom-up**" approach
- Based on the following distance measures:
  - $d_{ij} = \min\left(k_{Ti}^{2p}, k_{Tj}^{2p}\right)\frac{\Delta R_{ij}}{R}$
  - $d_{iB} = k_{Ti}^{2p}$
  - Find smallest of all d_{ij} and d_{iB}
  - If d_{ij}: recombine particles i and j
  - If d_{iB}: declare particle i a jet
  - Iterate until no particles are left
- p=1: kT-Algorithm
  - attempt to reverse QCD branching
- "Radius" Parameter R: any two jets will be separated at least by ΔR>R
- Sequential clustering algorithms are by construction **IRC-safe** for al finite p!

Sequential Clustering Algorithms



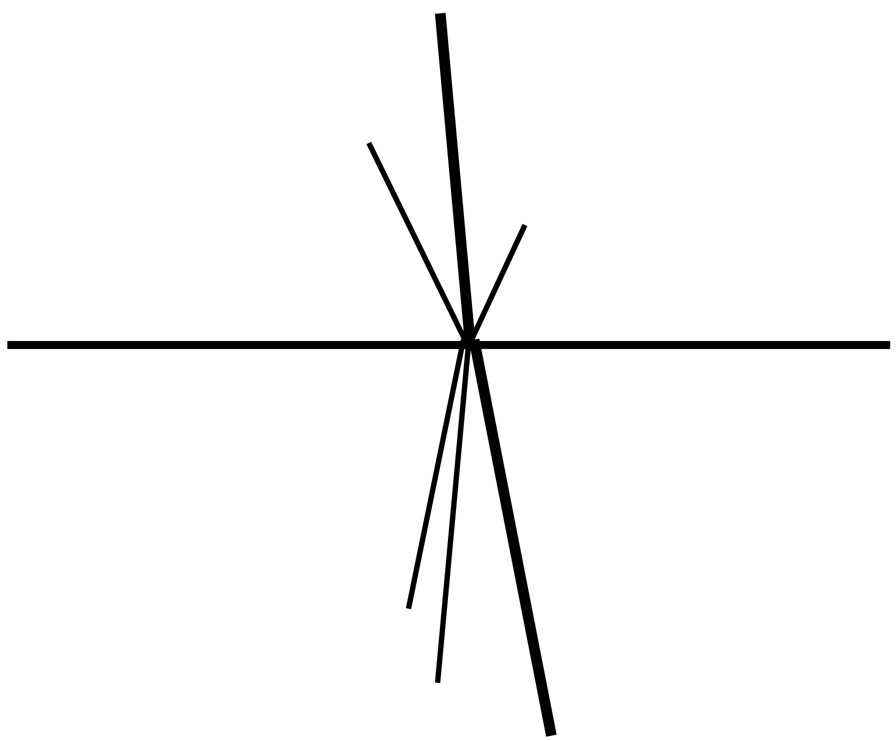
- "**bottom-up**" approach
- Based on the following distance measures:
  - $d_{ij} = \min\left(k_{Ti}^{2p}, k_{Tj}^{2p}\right) \frac{\Delta R_{ij}}{R}$
  - $d_{iB} = k_{Ti}^{2p}$
  - Find smallest of all d_{ij} and d_{iB}
  - If d_{ij}: recombine particles i and j
  - If d_{iB}: declare particle i a jet
  - Iterate until no particles are left
- p=1: kT-Algorithm
  - attempt to reverse QCD branching
- "Radius" Parameter R: any two jets will be separated at least by ΔR>R
- Sequential clustering algorithms are by construction **IRC-safe** for al finite p!

Sequential Clustering Algorithms



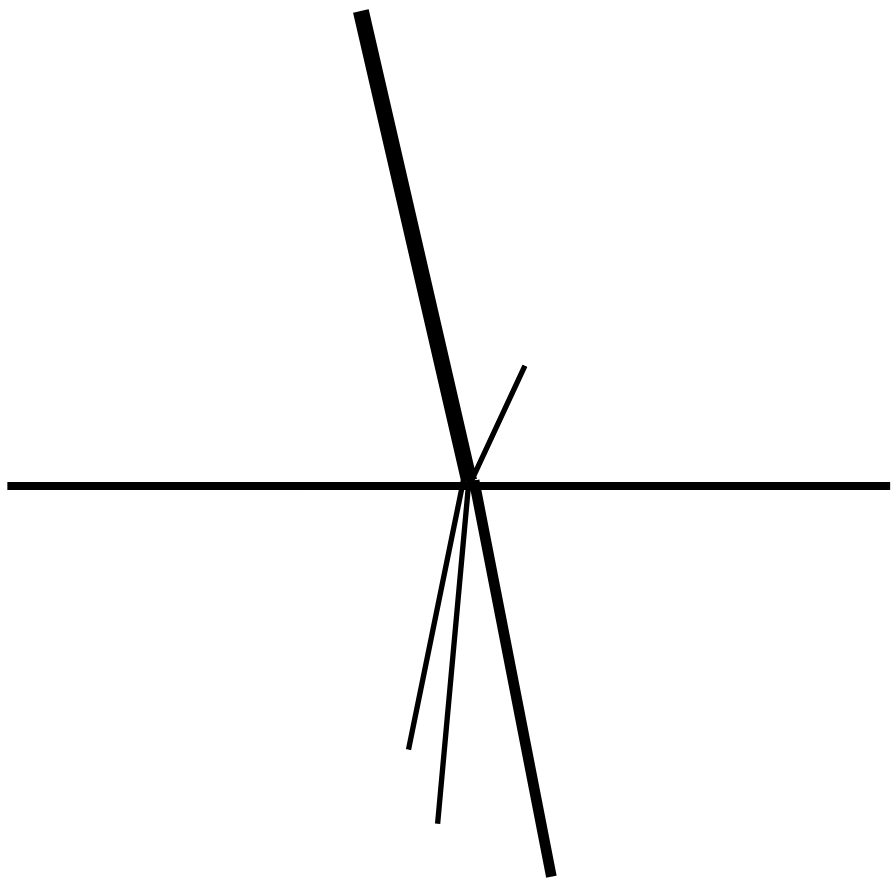
- "**bottom-up**" approach
- Based on the following distance measures:
  - $d_{ij} = \min\left(k_{Ti}^{2p}, k_{Tj}^{2p}\right)\frac{\Delta R_{ij}}{R}$
  - $d_{iB} = k_{Ti}^{2p}$
  - Find smallest of all d_{ij} and d_{iB}
  - If d_{ij}: recombine particles i and j
  - If d_{iB}: declare particle i a jet
  - Iterate until no particles are left
- p=1: kT-Algorithm
  - attempt to reverse QCD branching
- "Radius" Parameter R: any two jets will be separated at least by ΔR>R
- Sequential clustering algorithms are by construction **IRC-safe** for al finite p!

Sequential Clustering Algorithms



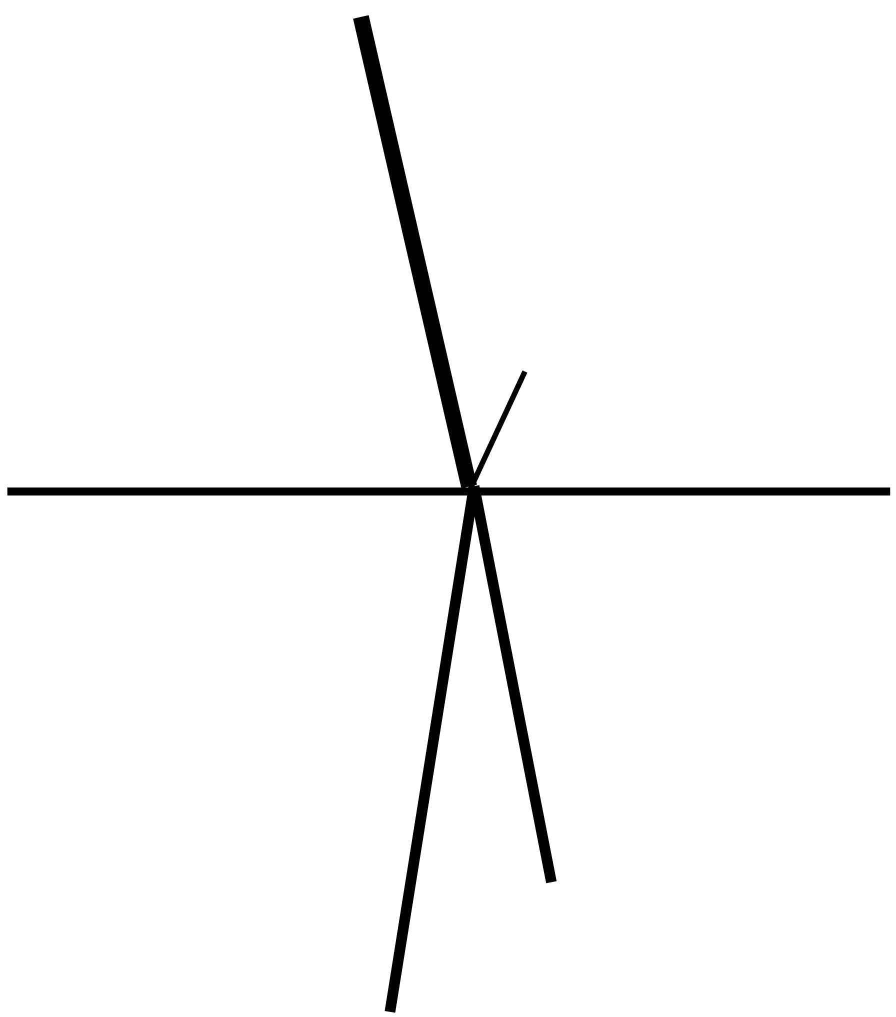
- "**bottom-up**" approach
- Based on the following distance measures:
  - $d_{ij} = \min\left(k_{Ti}^{2p}, k_{Tj}^{2p}\right) \frac{\Delta R_{ij}}{R}$
  - $d_{iB} = k_{Ti}^{2p}$
  - Find smallest of all d_{ij} and d_{iB}
  - If d_{ij}: recombine particles i and j
  - If d_{iB}: declare particle i a jet
  - Iterate until no particles are left
- p=1: kT-Algorithm
  - attempt to reverse QCD branching
- "Radius" Parameter R: any two jets will be separated at least by ΔR>R
- Sequential clustering algorithms are by construction **IRC-safe** for al finite p!

Sequential Clustering Algorithms



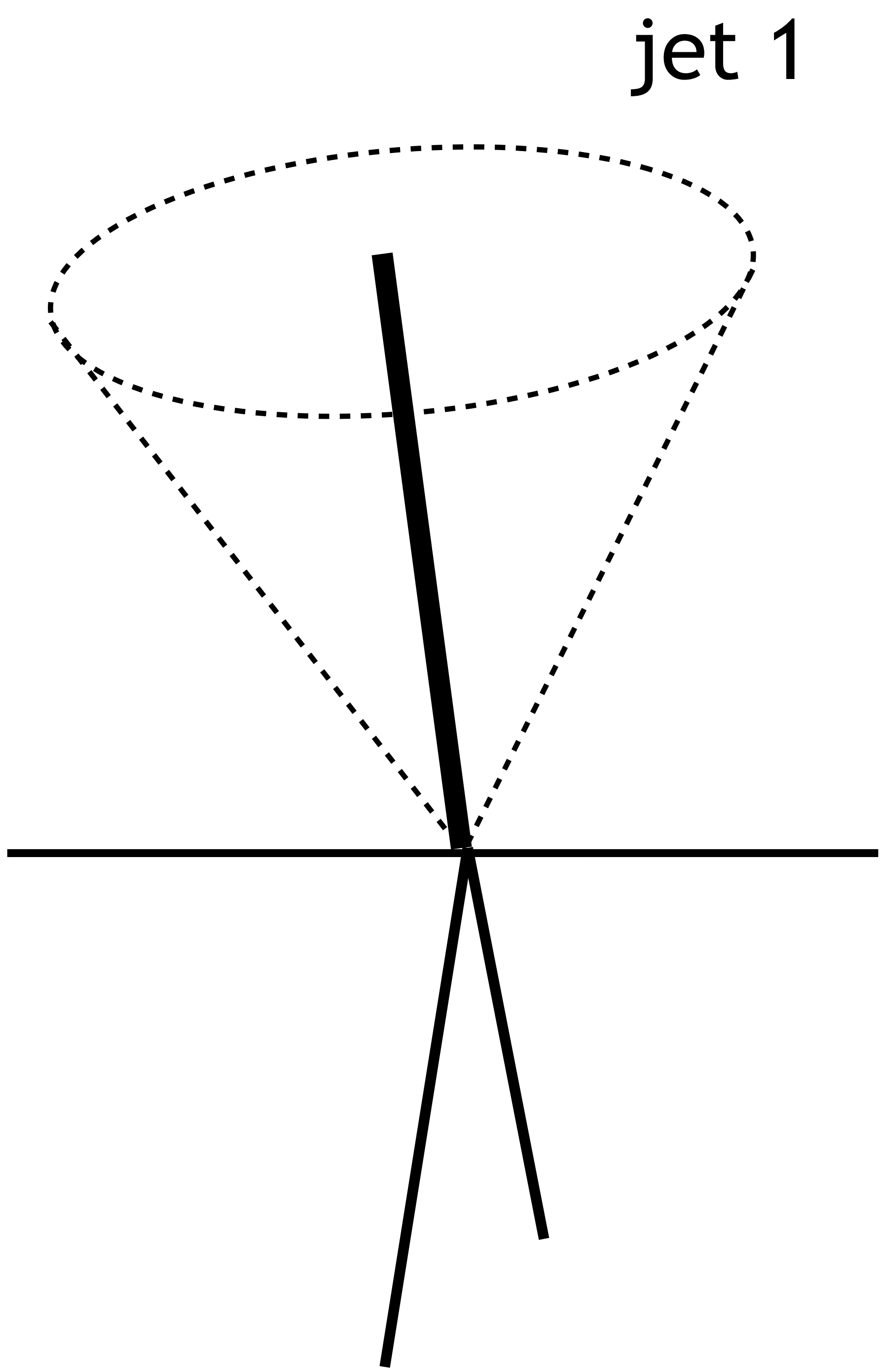
- "**bottom-up**" approach
- Based on the following distance measures:
  - $d_{ij} = \min\left(k_{Ti}^{2p}, k_{Tj}^{2p}\right) \frac{\Delta R_{ij}}{R}$
  - $d_{iB} = k_{Ti}^{2p}$
  - Find smallest of all d_{ij} and d_{iB}
  - If d_{ij}: recombine particles i and j
  - If d_{iB}: declare particle i a jet
  - Iterate until no particles are left
- p=1: kT-Algorithm
  - attempt to reverse QCD branching
- "Radius" Parameter R: any two jets will be separated at least by ΔR>R
- Sequential clustering algorithms are by construction **IRC-safe** for al finite p!

Sequential Clustering Algorithms



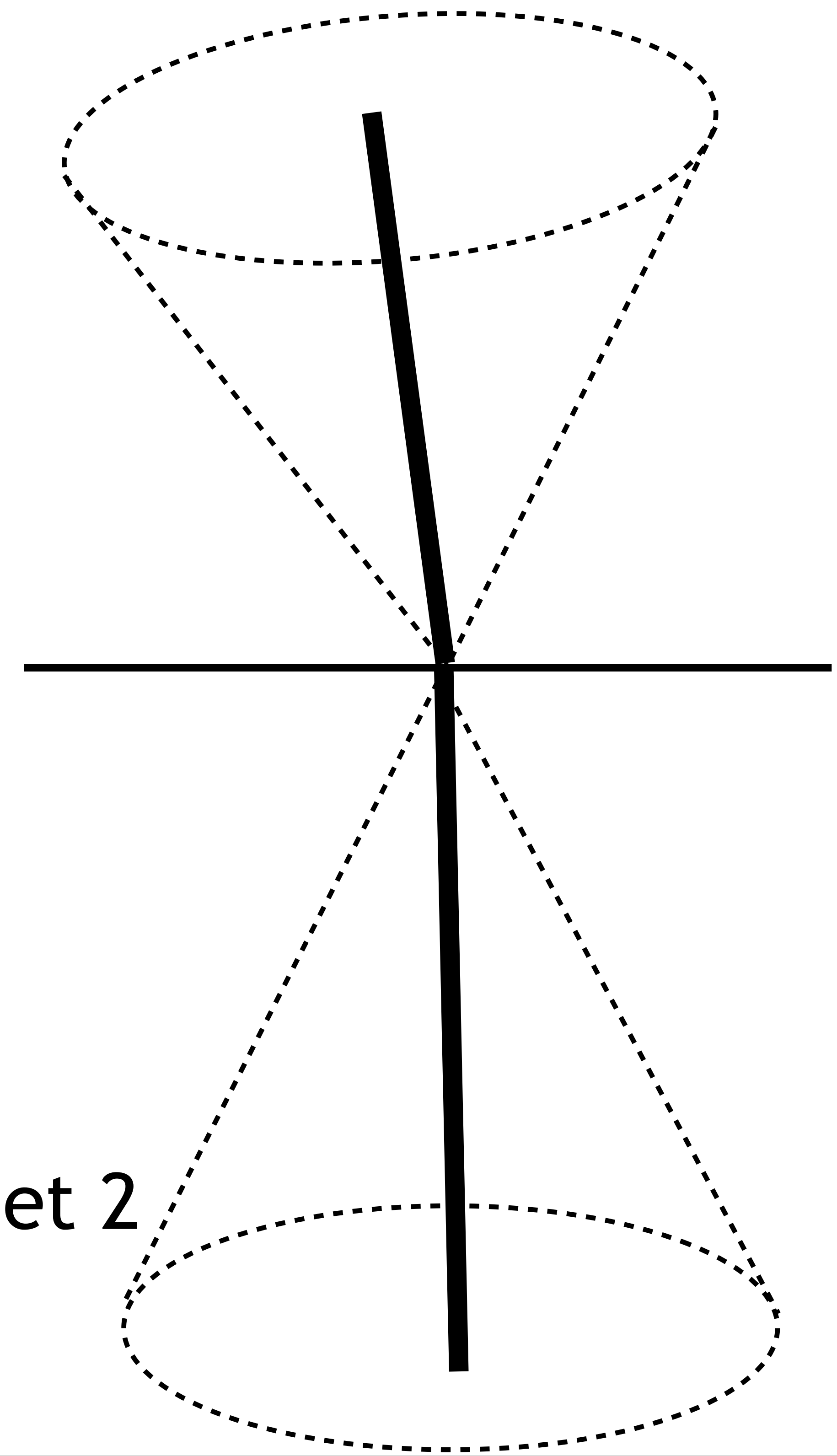
- "**bottom-up**" approach
- Based on the following distance measures:
  - $d_{ij} = \min\left(k_{Ti}^{2p}, k_{Tj}^{2p}\right) \frac{\Delta R_{ij}}{R}$
  - $d_{iB} = k_{Ti}^{2p}$
  - Find smallest of all d_{ij} and d_{iB}
  - If d_{ij}: recombine particles i and j
  - If d_{iB}: declare particle i a jet
  - Iterate until no particles are left
- p=1: kT-Algorithm
  - attempt to reverse QCD branching
- "Radius" Parameter R: any two jets will be separated at least by ΔR>R
- Sequential clustering algorithms are by construction **IRC-safe** for al finite p!

Sequential Clustering Algorithms



- **"bottom-up"** approach
- Based on the following distance measures:
  - $d_{ij} = \min\left(k_{Ti}^{2p}, k_{Tj}^{2p}\right) \frac{\Delta R_{ij}}{R}$
  - $d_{iB} = k_{Ti}^{2p}$
  - Find smallest of all d_{ij} and d_{iB}
  - If d_{ij}: recombine particles i and j
  - If d_{iB}: declare particle i a jet
  - Iterate until no particles are left
- p=1: kT-Algorithm
  - attempt to reverse QCD branching
- "Radius" Parameter R: any two jets will be separated at least by ΔR>R
- Sequential clustering algorithms are by construction **IRC-safe** for al finite p!

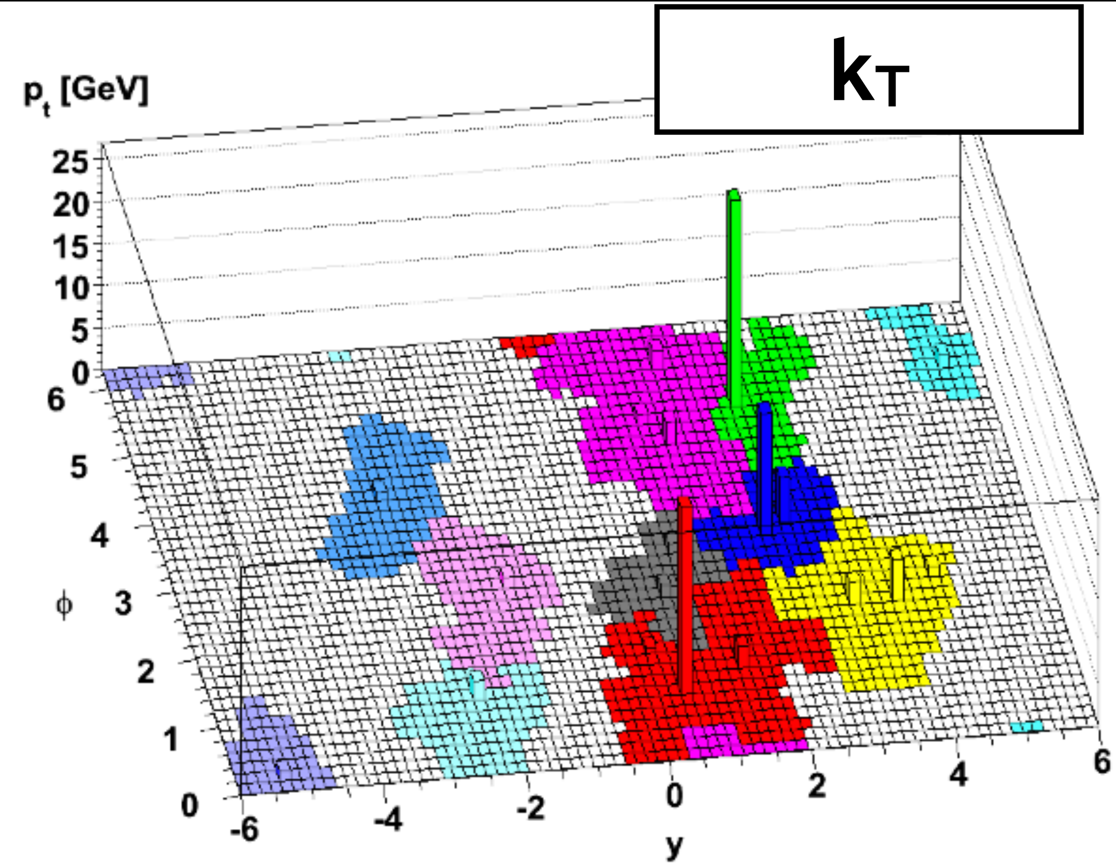
Sequential Clustering Algorithms



- "**bottom-up**" approach
- Based on the following distance measures:
  - $d_{ij} = \min\left(k_{Ti}^{2p}, k_{Tj}^{2p}\right) \frac{\Delta R_{ij}}{R}$
  - $d_{iB} = k_{Ti}^{2p}$
  - Find smallest of all d_{ij} and d_{iB}
  - If d_{ij}: recombine particles i and j
  - If d_{iB}: declare particle i a jet
  - Iterate until no particles are left
- p=1: kT-Algorithm
  - attempt to reverse QCD branching
- "Radius" Parameter R: any two jets will be separated at least by ΔR>R
- Sequential clustering algorithms are by construction **IRC-safe** for al finite p!

Making Everybody Happy: Anti-k_T

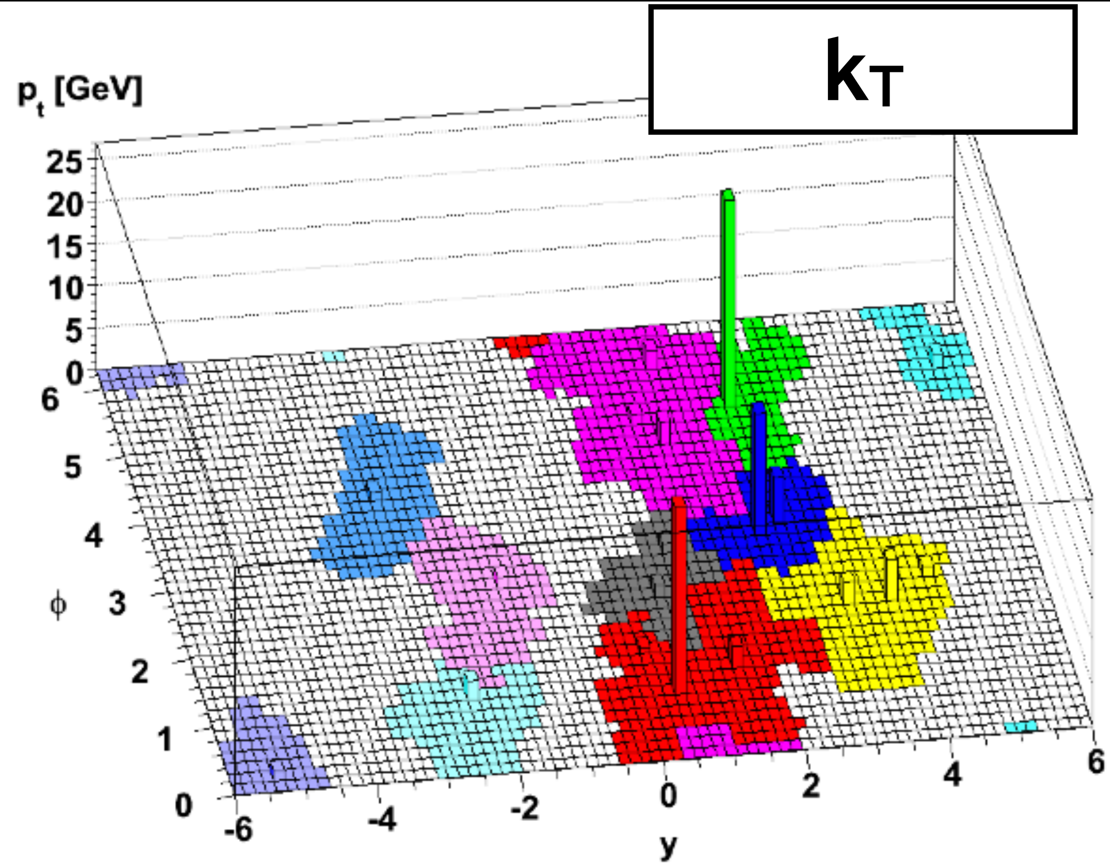
- *Theorists* are keen on IR&C-Safety (as should experimentalists!)
- *Experimentalists* do not like irregular shape of e.g. kT jets, prefer computable geometrical structure (e.g. for acceptance corrections)
- Both are likely wary of unnecessary complications (e.g. split&merge)

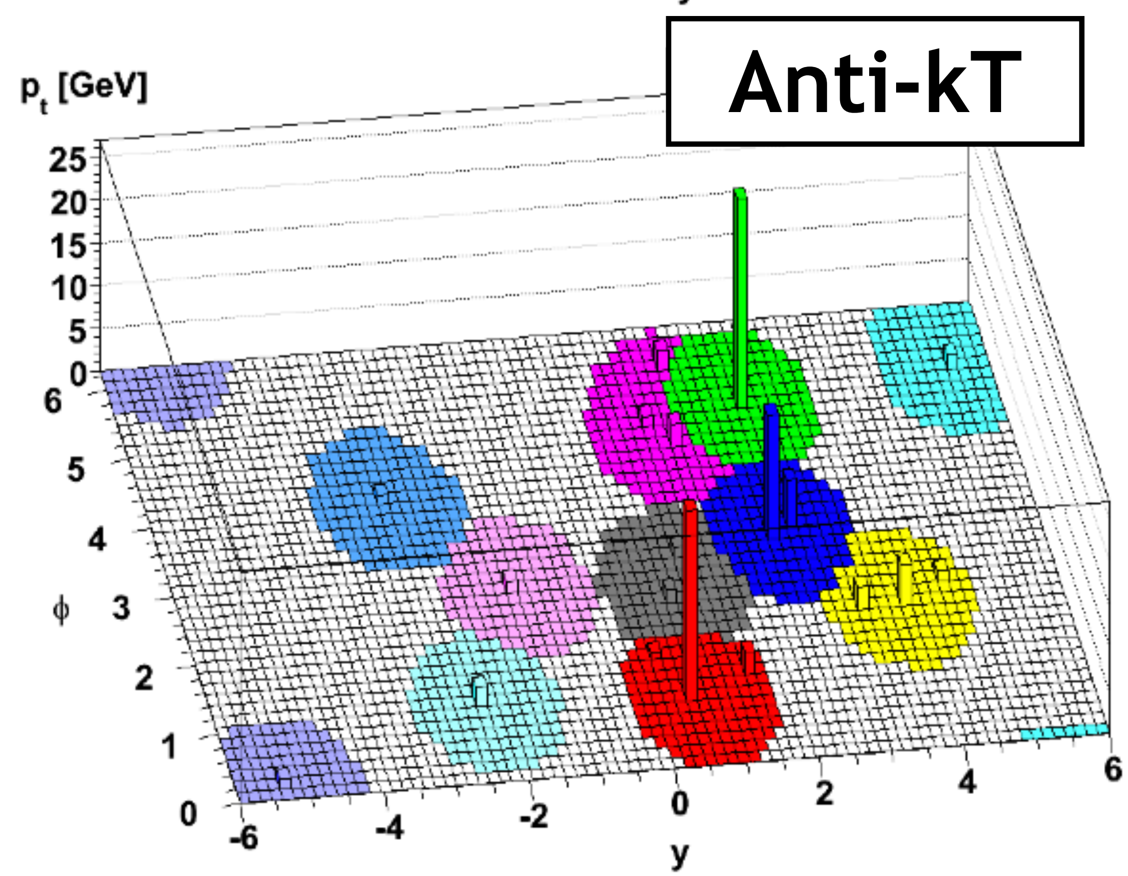


Making Everybody Happy: Anti-k_T

- *Theorists* are keen on IR&C-Safety (as should experimentalists!)
- *Experimentalists* do not like irregular shape of e.g. kT jets, prefer computable geometrical structure (e.g. for acceptance corrections)
- Both are likely wary of unnecessary complications (e.g. split&merge)
- **Solution: Anti-kT [1] Algorithm!**
  - sequential clustering (p=-1), IR&C-safe
  - simple, one parameter R
  - produces cone-shaped jets of radius R
  - clusters high-pT deposits first
  - fast as in fastjet

  ➡currently first choice at both ATLAS&CMS



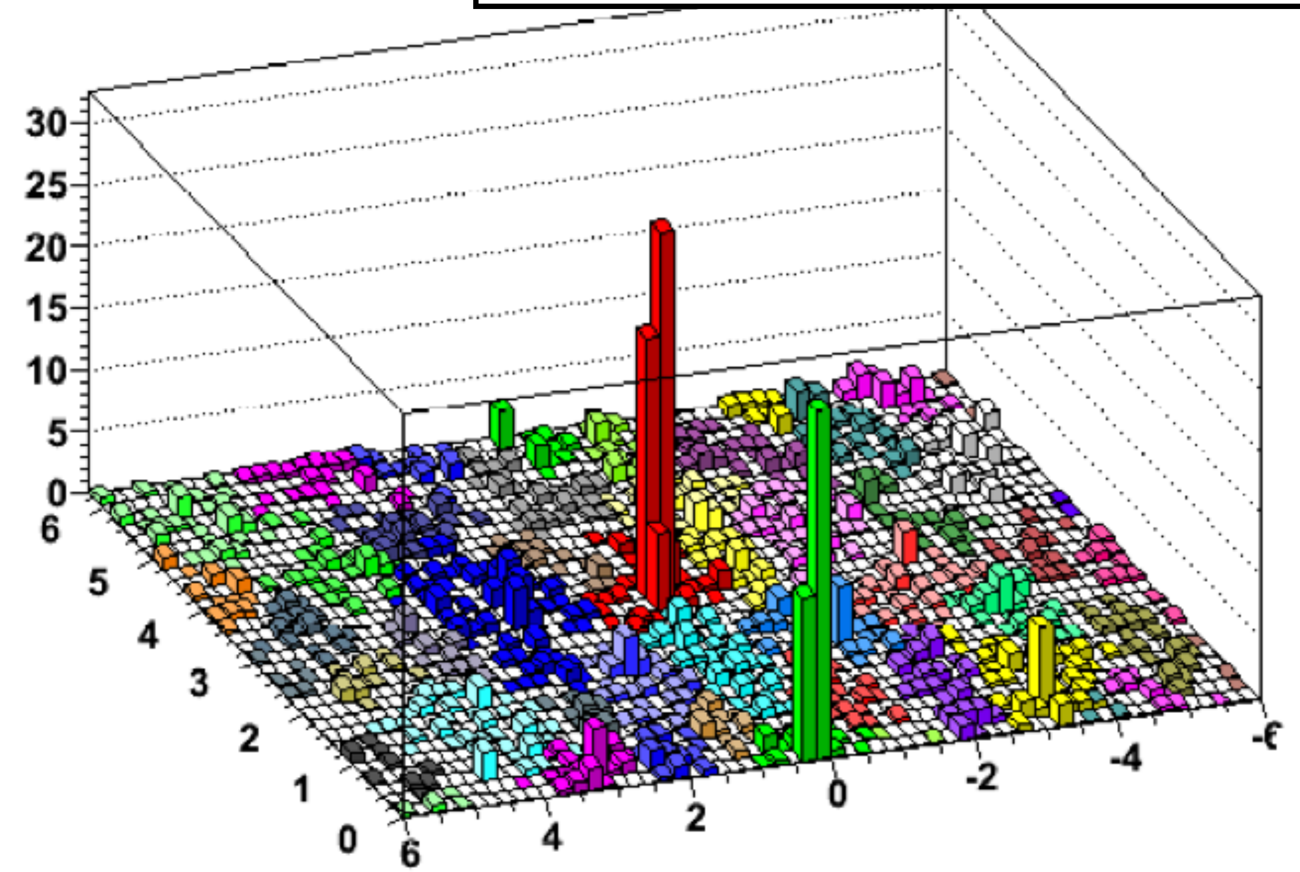


[1] http://arxiv.org/abs/0802.1189

Jet Areas

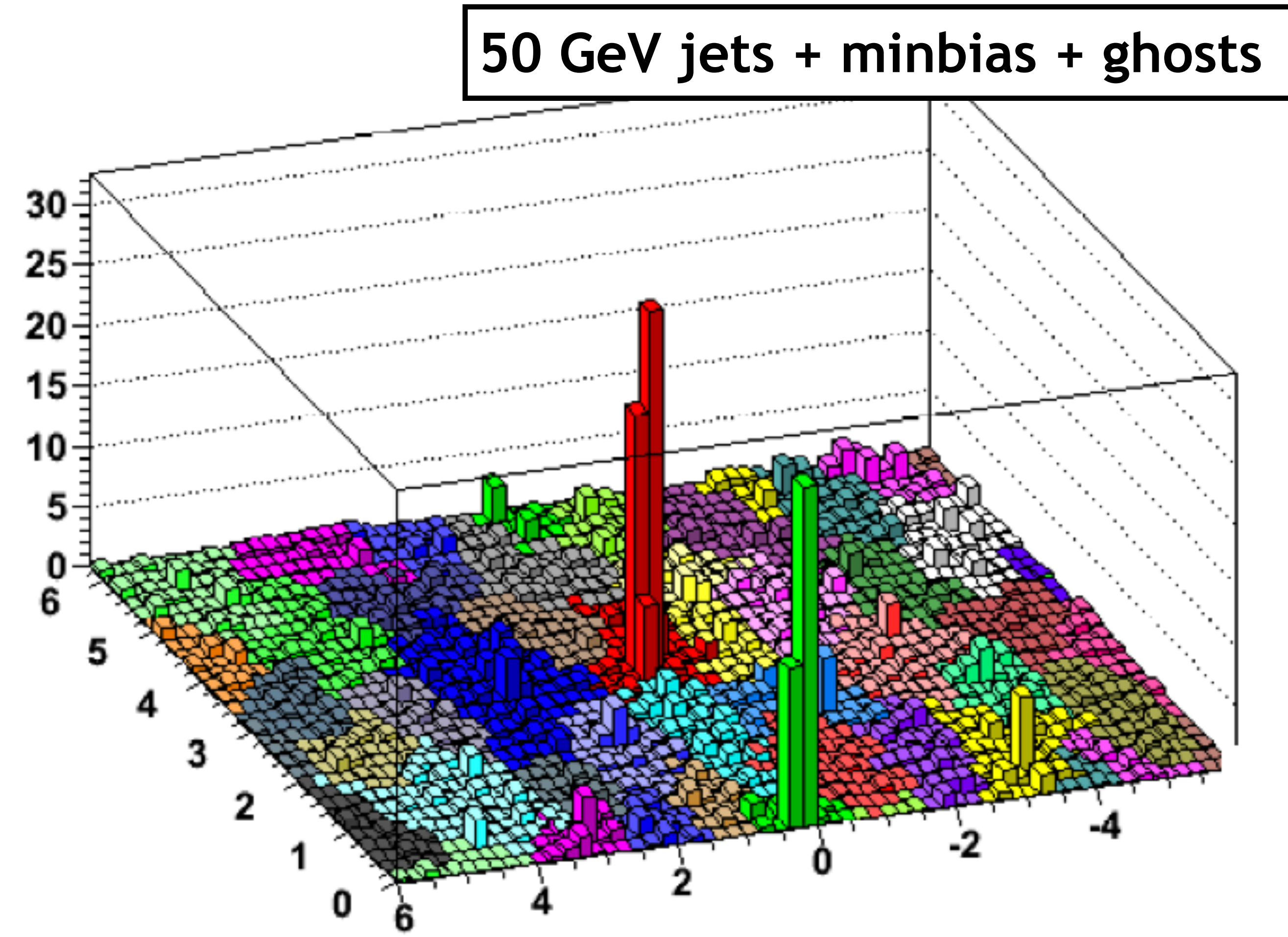
- Jet Area definition [1] suitable for all IR&C-safe jet definitions
  - i.e. those where hard jets are not modified by the addition of extra soft particles
  - geometrical jet shape irrelevant!

50 GeV jets + minbias



Jet Areas

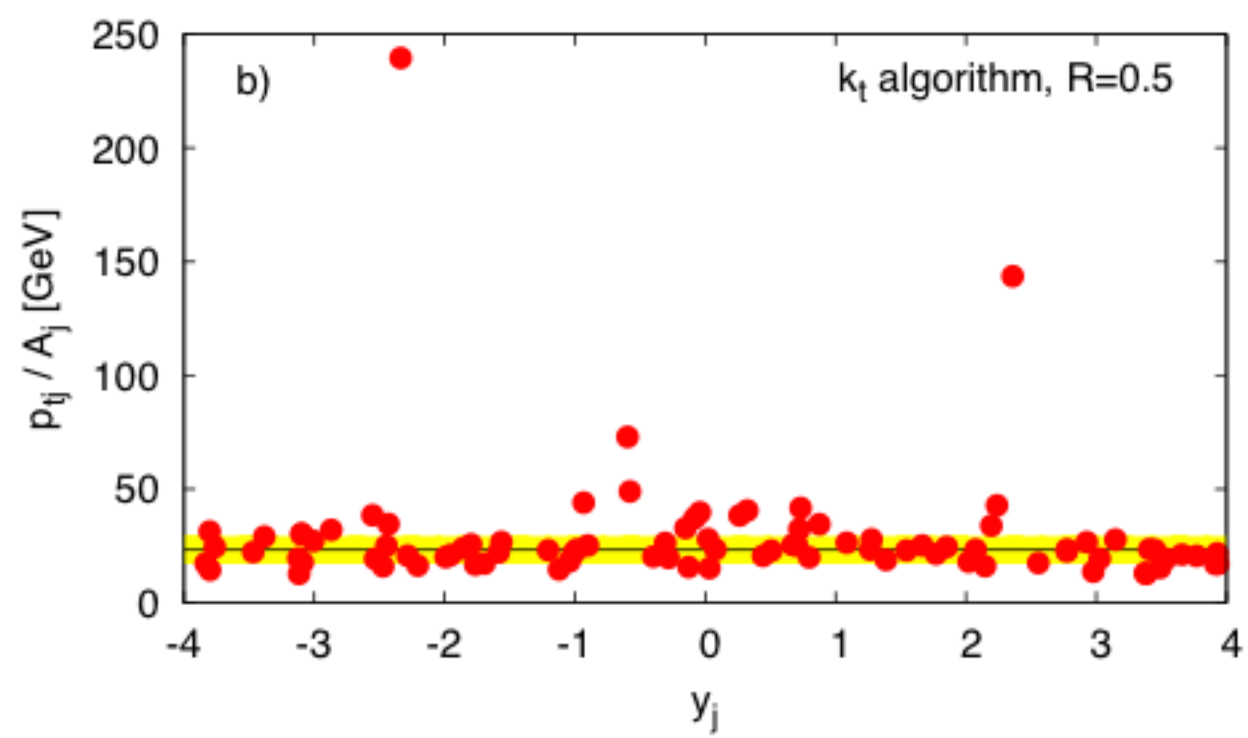
- Jet Area definition [1] suitable for all <u>IR&C-safe</u> jet definitions
  - i.e. those where hard jets are not modified by the addition of extra soft particles
  - geometrical jet shape irrelevant!
- <u>Prescription</u>: *Add infinitely soft particles (“ghosts”) and identify the region in y-φ where those ghosts are clustered with a given jet.*
  - The extension of that region gives a measure of the dimension-less (active) jet area
  - As ~10.000 ghost particles are typically needed to cover the relevant y-φ-space, exquisit computational performance is in reality required (fastjet: ~0.6s for N~10000)

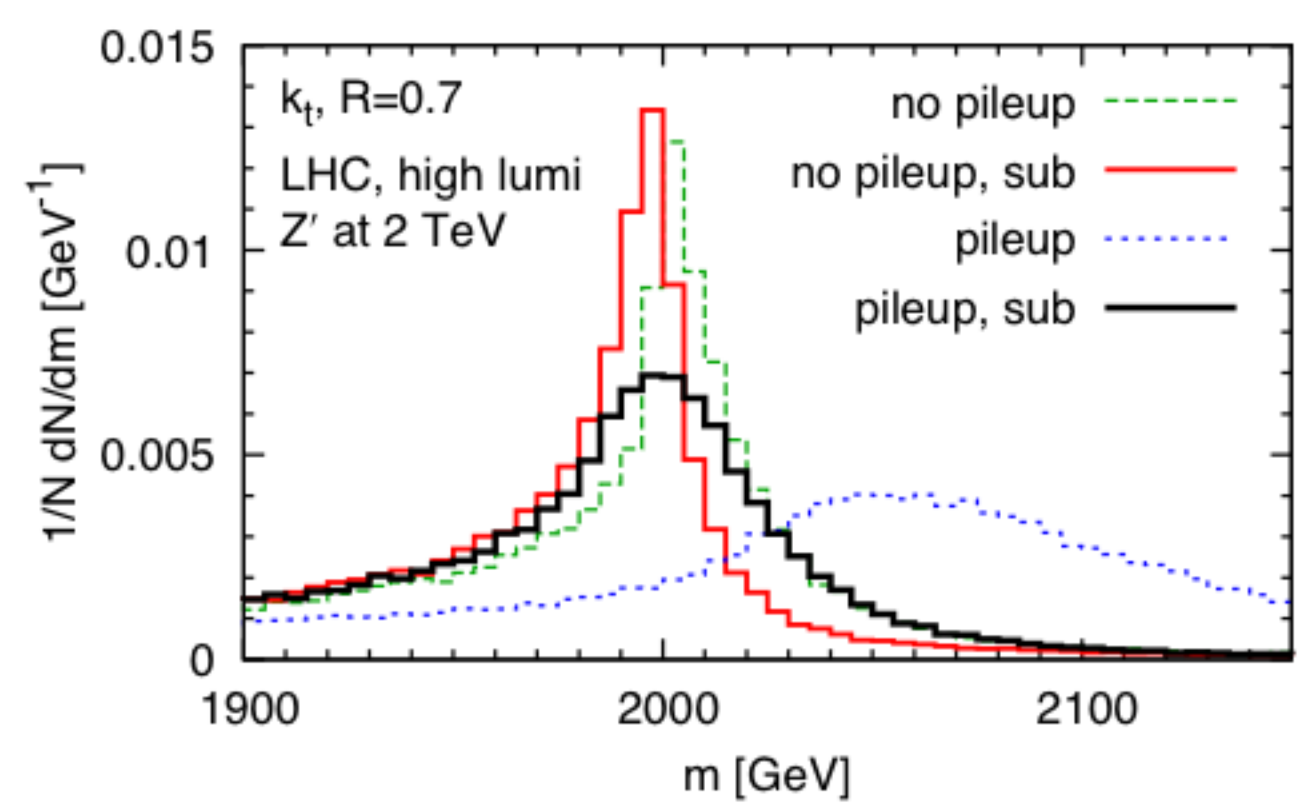


[1] <u>http://arxiv.org/abs/0802.1188v2</u>

Pile-Up Subtraction

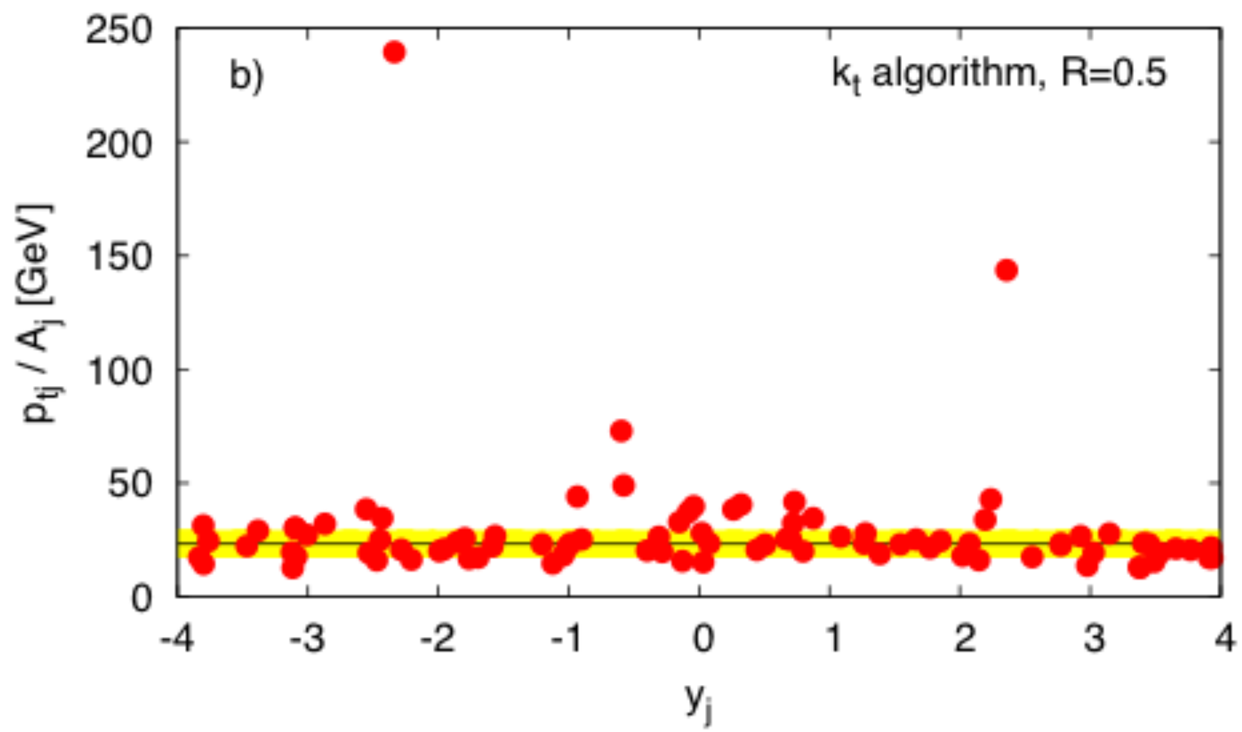
- In events containting Pile-Up, suitable algorithms (e.g. kT) yield a large number of soft jets (“PU jets”)
- In the limit where the contamination from PU is uniform and dense, each such PU jet will have the property that pT/A is equal to the **event PU density per unit area, ρ**
- In reality ρ can be measured event-by-event by e.g. fitting pT/A as a function of y [1]
- Given ρ and each jet’s area A, the PU contamination of each jet can be calculated as ρA and subtracted, event-by-event

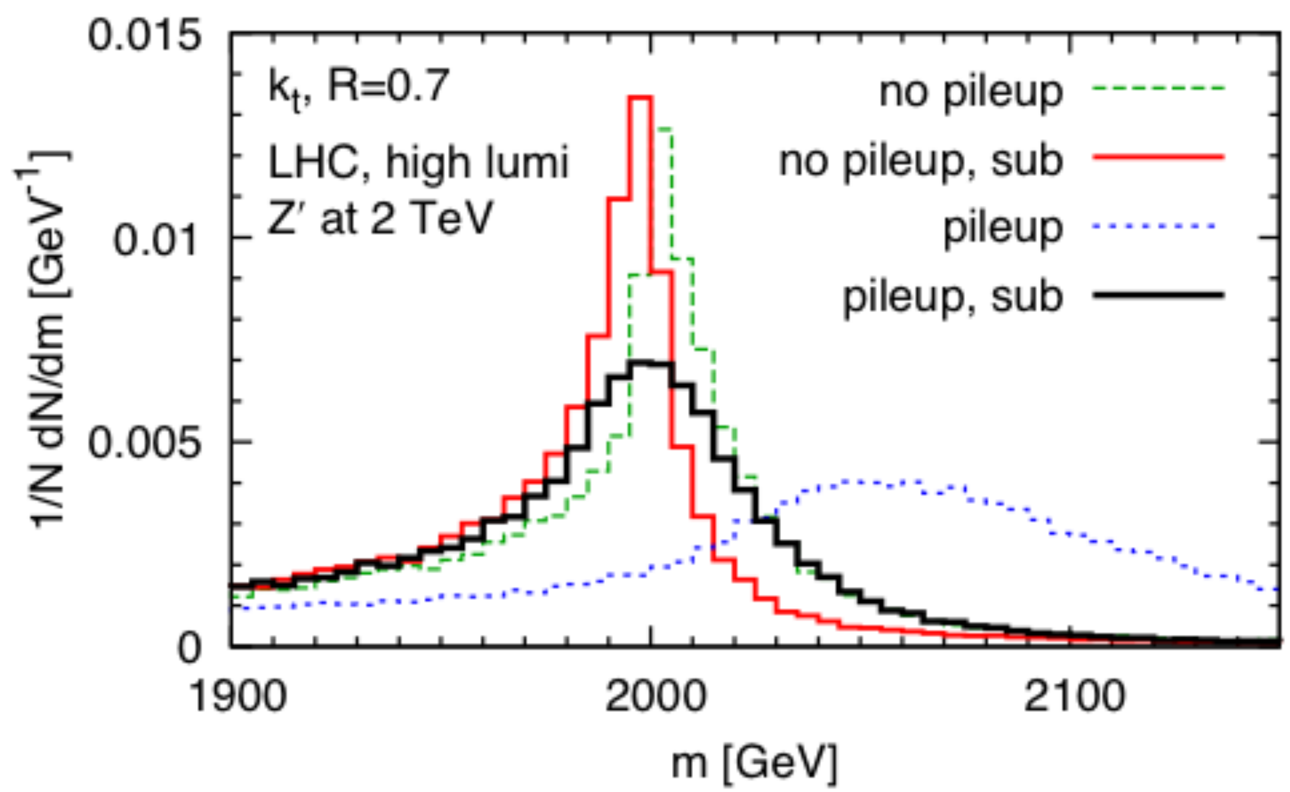




Pile-Up Subtraction

- In events containting Pile-Up, suitable algorithms (e.g. kT) yield a large number of soft jets (“PU jets”)
- In the limit where the contamination from PU is uniform and dense, each such PU jet will have the property that pT/A is equal to the **event PU density per unit area, ρ**
- In reality ρ can be measured event-by-event by e.g. fitting pT/A as a function of y [1]
- Given ρ and each jet’s area A, the PU contamination of each jet can be calculated as ρA and subtracted, event-by-event





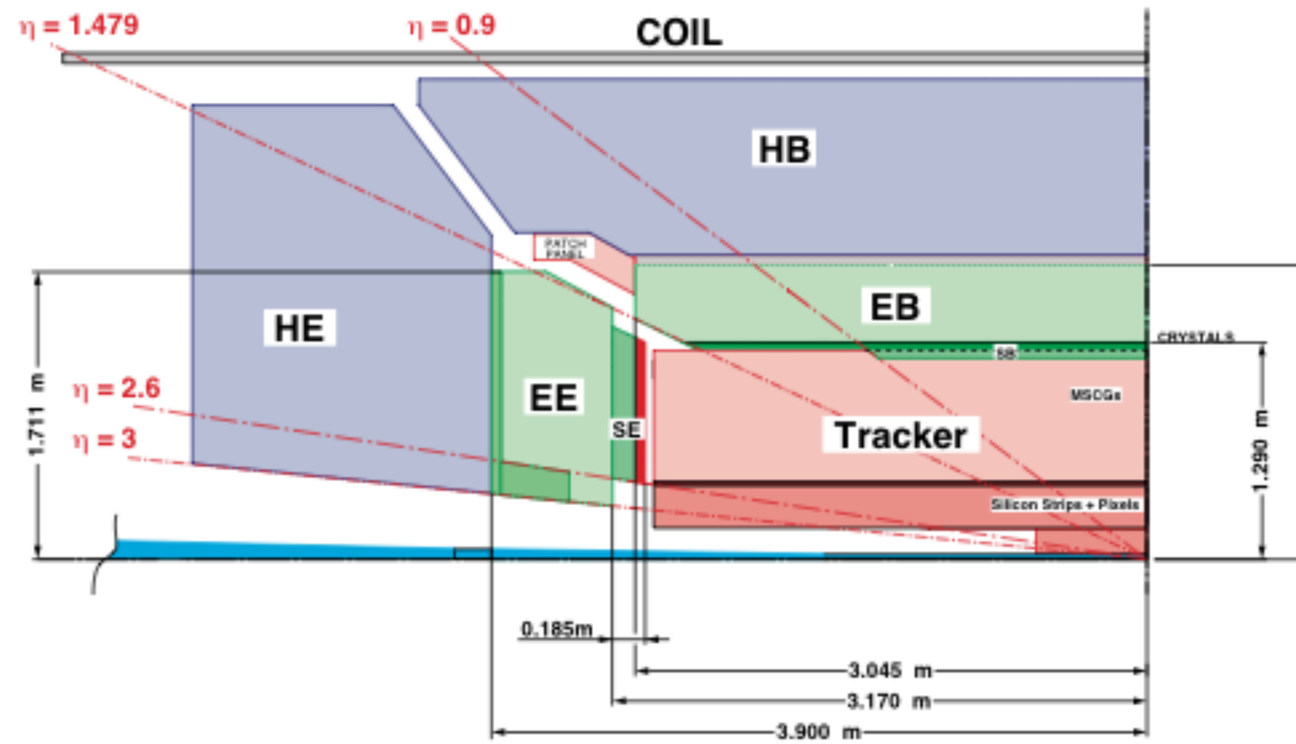
[1] http://arxiv.org/abs/0707.1378v2

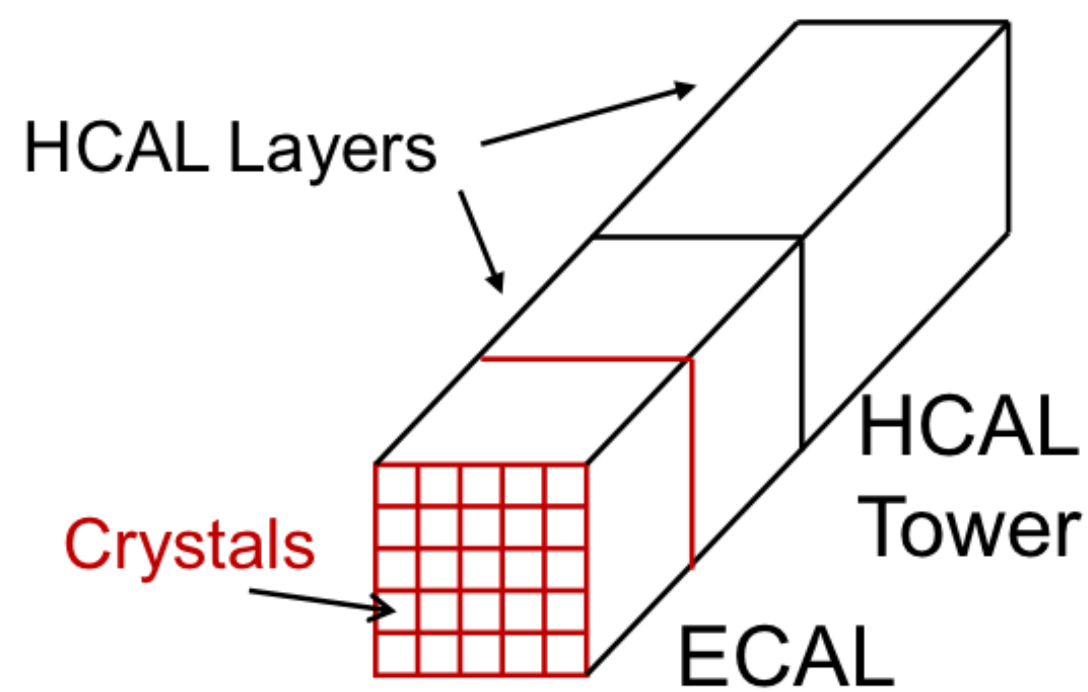
“Particle” Reconstruction

- Calorimeter Towers
- Topological Clusters
- Particle Flow

Calorimeter Towers at CMS

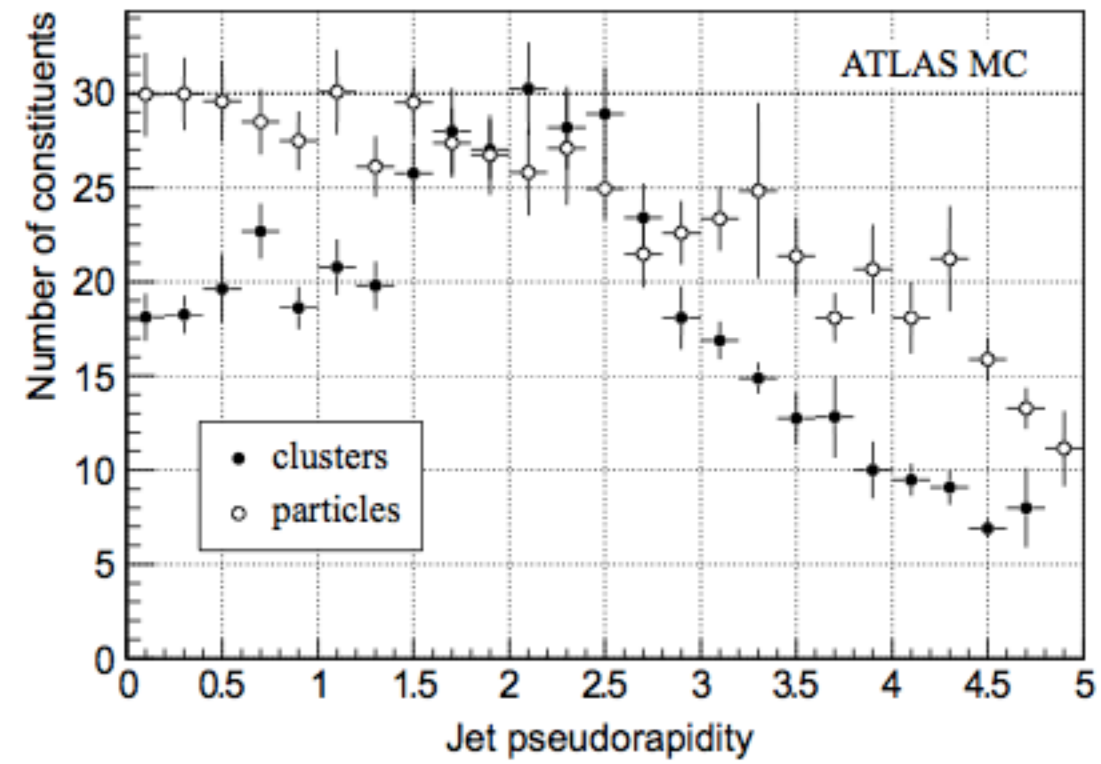
- Calorimeter towers consist of one HCAL tower and 5x5 ECAL crystals
- Calorimeter tower four-vectors constructed by
  - considering deposits above threshold
  - summing HCAL and ECAL deposits
  - adjusting the direction for
    - ★ primary vertex
    - ★ optimized common shower depth
    - ★ profile of the ECAL depoistions
  - forcing the mass to be 0
- Drawbacks
  - CMS Calorimeter System non-linear and non-compensating
  - High magnetic field bends low pT particles significantly (>1GeV to reach calo-surface)

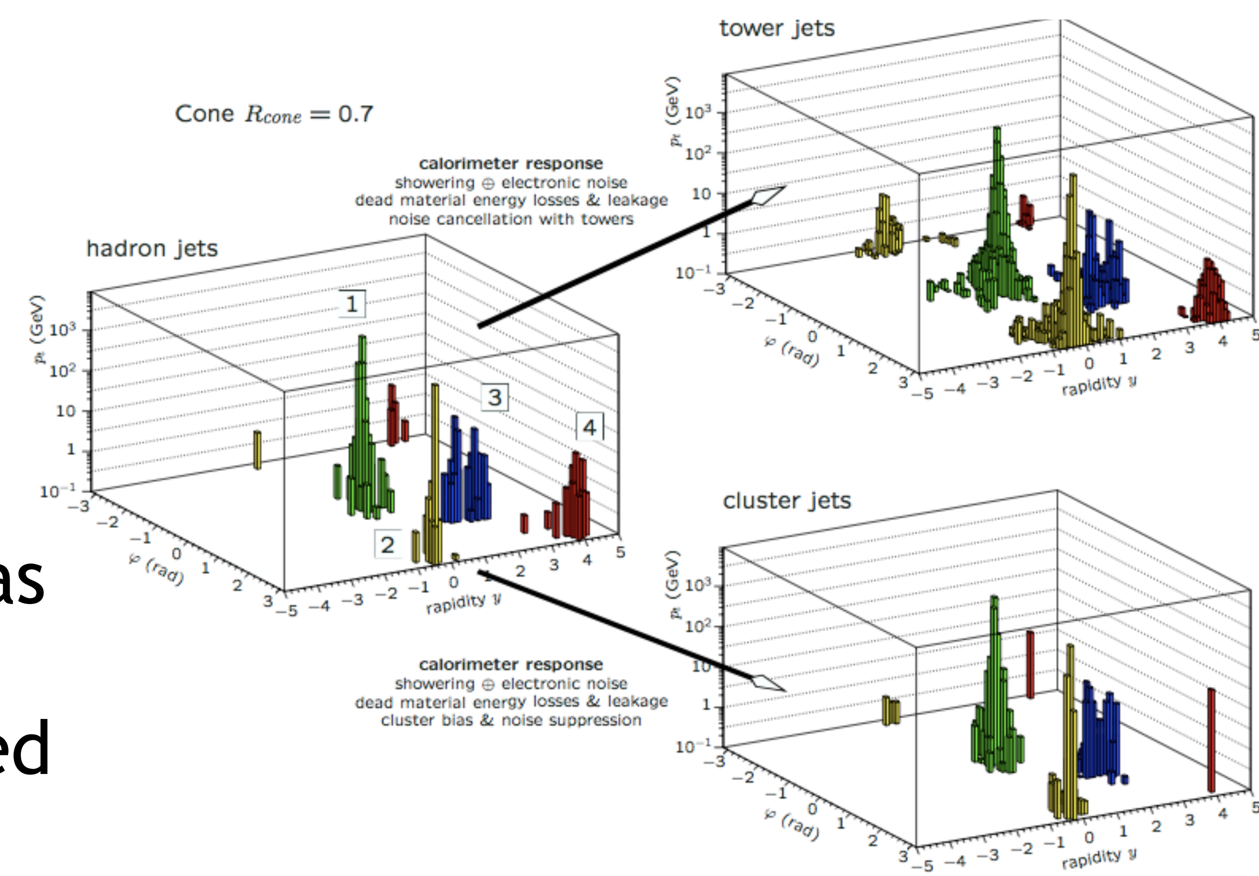




Topological Clusters at ATLAS

- reconstruct three-dimensional “energy blobs” representing the showers developing for **each particle** entering the calorimeter
- attempt to **resolve shower overlaps** works best for 1.5<|eta|<3.
- **Noise suppression** scheme part of the clustering algorithm via cell signal significance cuts
- Cluster characteristics used to **classify** as electromagnetic or hadronic, corresponding cluster **calibration** derived from single-particle simulation





Particle Flow at CMS

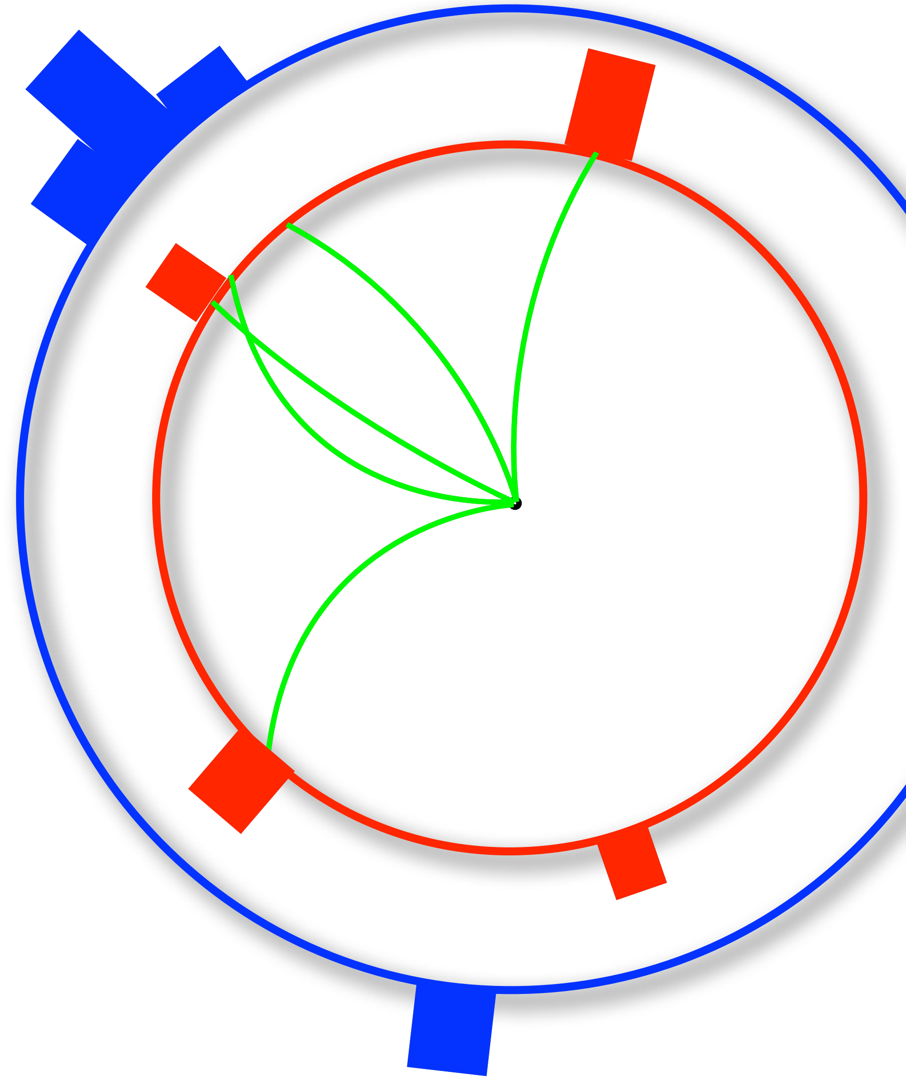
- Exploit CMS strongest features to reconstruct (particles then) jets!
  - High-resolution tracker & muon system (~65%)
  - High-granularity (space & time) and high-resolution ECAL (~25%)
  - Minimize reliance on HCAL (~10%)

Goal: Reconstruct, classify & calibrate all individual stable particles in the event, **optimally combining the information from *all* CMS subsystems**

- **Distinguish the following five particle types:**
  Photons, Neutral Hadrons, Charged Hadrons, Electrons & Muons
- Improved jet (&MET!) reconstruction from PF particle candidates
- Note fundamental difference to approaches where jets are first reconstructed from calorimeter deposits, then corrected for matched tracks based on expected calo deposits of those tracks! (E.g. Jet-Plus-Track @CMS)

The Particle Flow Algorithm

Reconstruct all stable particles, making use of all CMS sub-detectors.

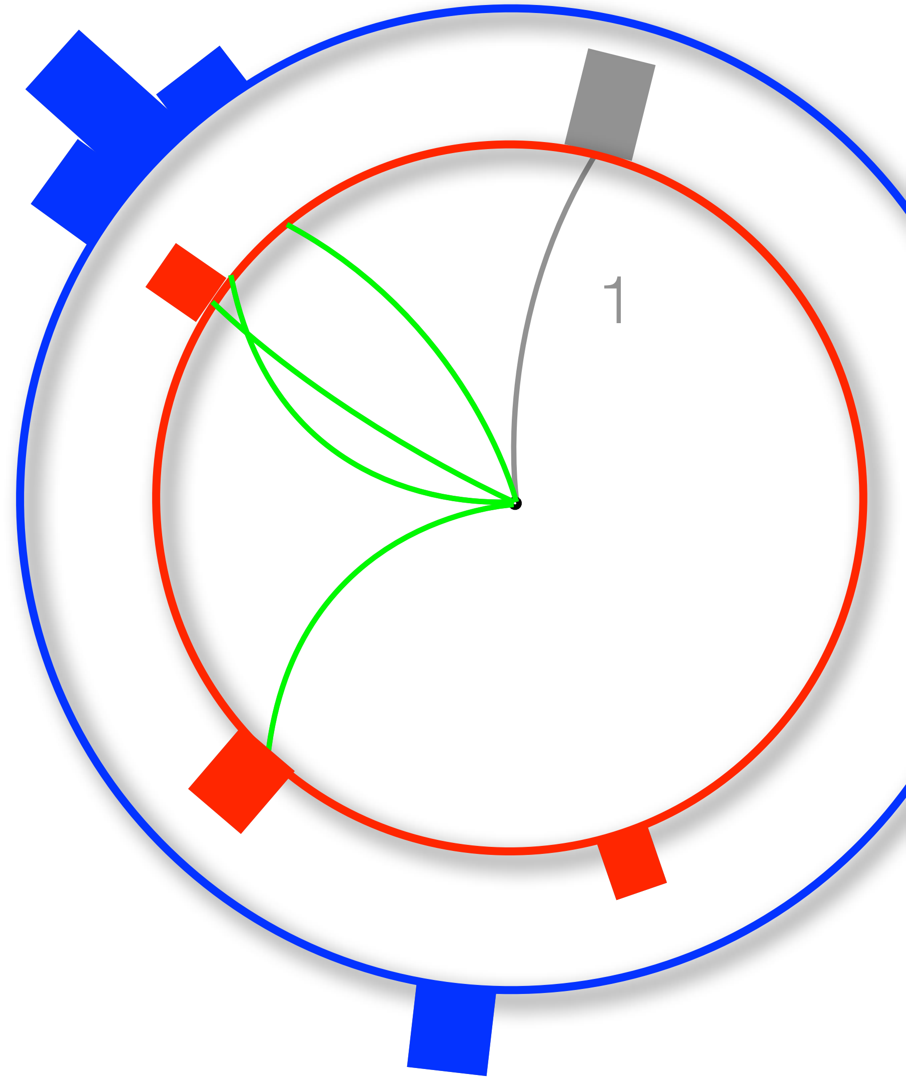


Five Particle Types

The Particle Flow Algorithm

Reconstruct all stable particles, making use of all CMS sub-detectors.

1. Get tracks linked to a single ECAL cluster:
   - →identify and create **electrons**



Five Particle Types

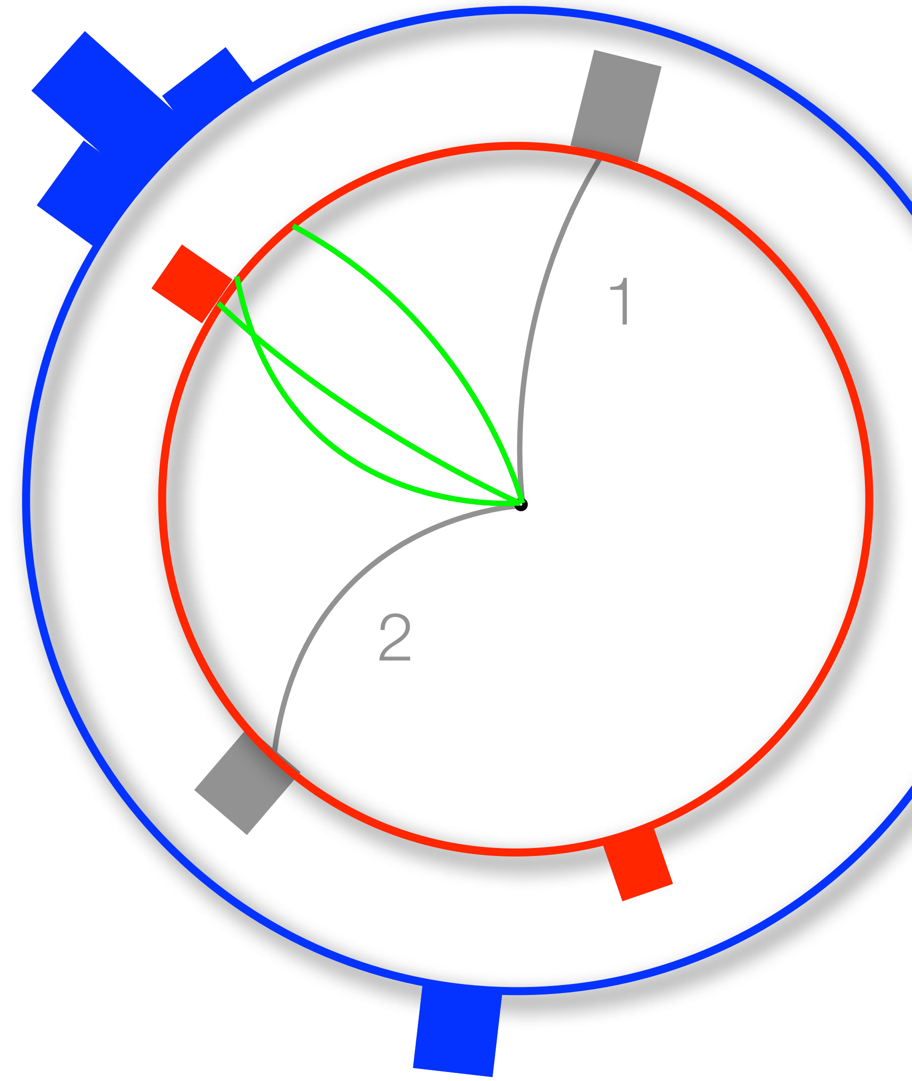
The Particle Flow Algorithm

Reconstruct all stable particles, making use of all CMS sub-detectors.

1. Get tracks linked to a single ECAL cluster:

   ➞identify and create **electrons**

2. If track/clusters compatible with muon chamber hits
   ➞create a **muon**
   else
   ➞create a **charged hadron**



Five Particle Types

The Particle Flow Algorithm

Reconstruct all stable particles, making use of all CMS sub-detectors.

1. Get tracks linked to a single ECAL cluster:

   ➞identify and create **electrons**

2. If track/clusters compatible with muon chamber hits

   ➞create a **muon**

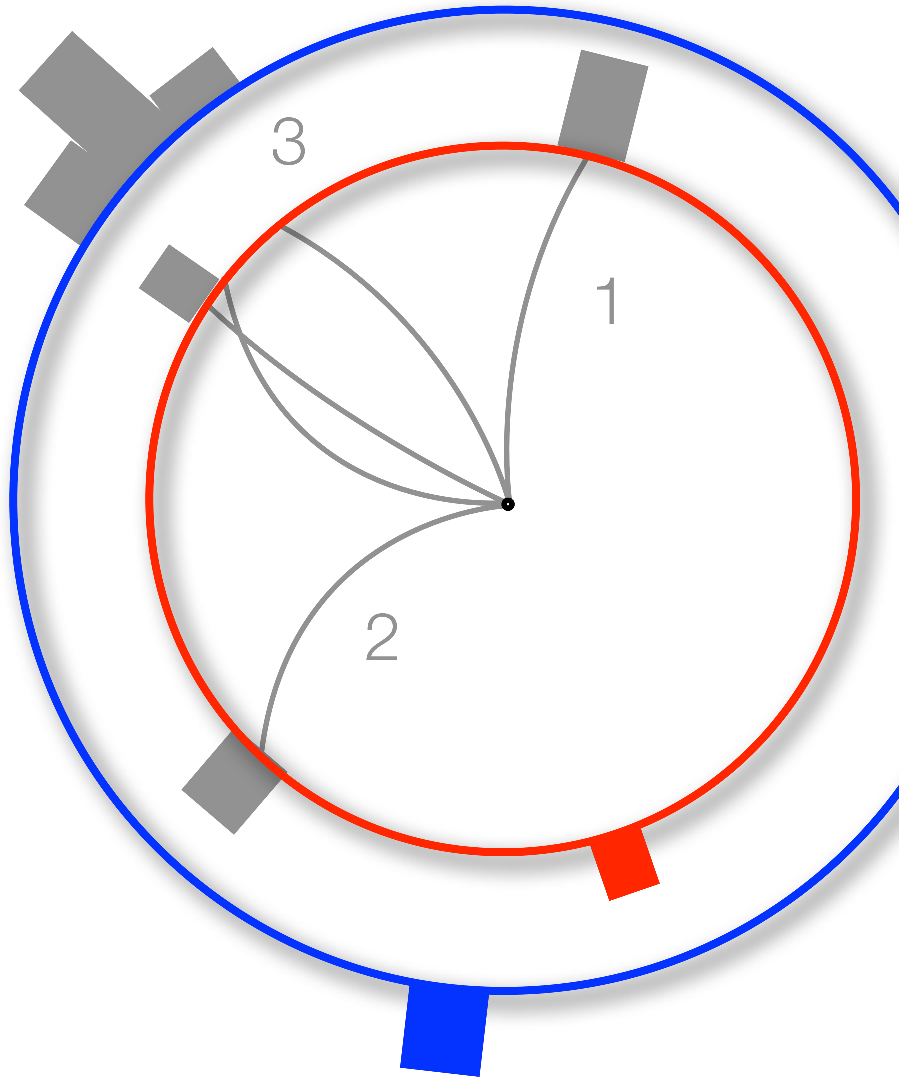
   else

   ➞create a **charged hadron**

3. For each HCAL cluster get all linked tracks and all ECAL clusters linked to tracks.

   Compute $E_{\mathrm{calo}} = E_{\mathrm{HCAL}} + E_{\mathrm{ECAL}}$

   - **if** (E_{calo} compatible with $\sum p_{\mathrm{tracks}}$) create a charged hadron for each track
   - **if** ($E_{\mathrm{calo}} > \sum p_{\mathrm{tracks}}$) create **photons** and/or **neutral hadrons** to account for missing E_{calo}



Five Particle Types

The Particle Flow Algorithm

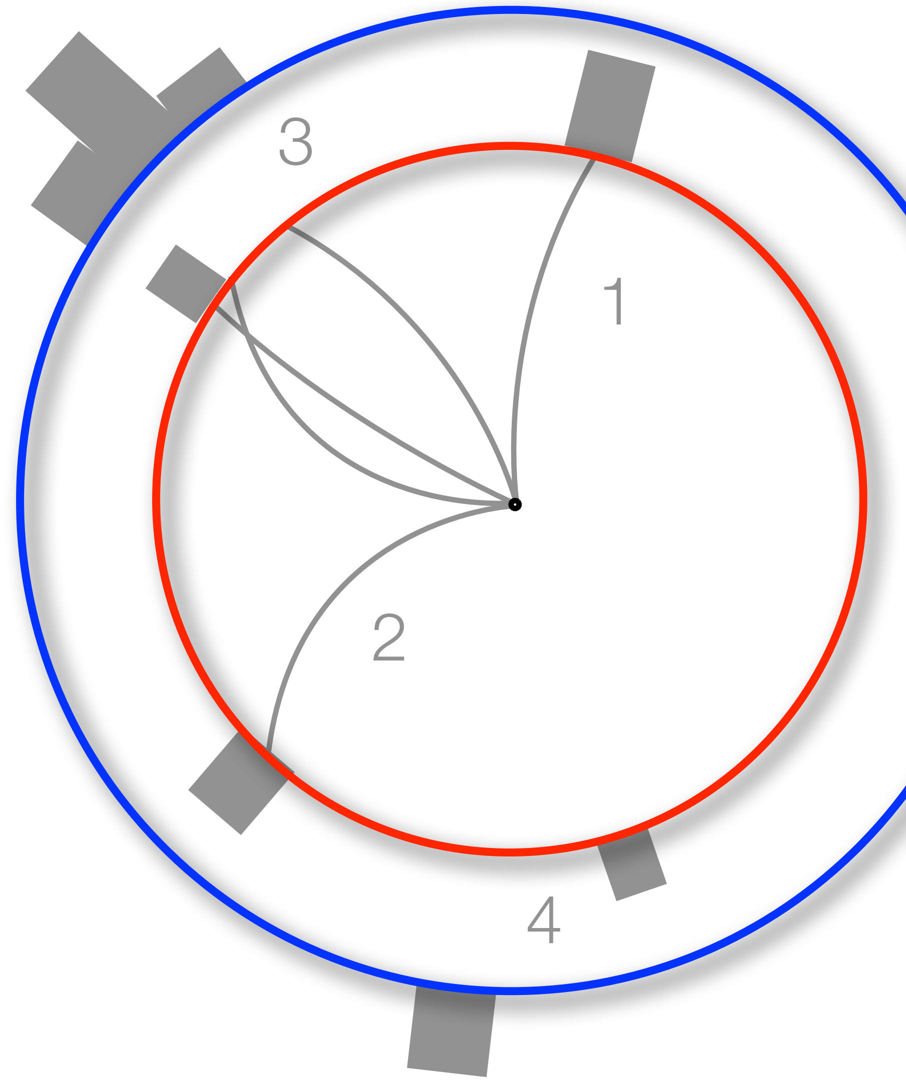
Reconstruct all stable particles, making use of all CMS sub-detectors.

1. Get tracks linked to a single ECAL cluster:
   →identify and create **electrons**

2. If track/clusters compatible with muon chamber hits
   →create a **muon**
   else
   →create a **charged hadron**

3. For each HCAL cluster get all linked tracks and all ECAL clusters linked to tracks.
   Compute $E_{\mathrm{calo}} = E_{\mathrm{HCAL}} + E_{\mathrm{ECAL}}$
   - **if** (E_{calo} compatible with $\sum p_{\mathrm{tracks}}$) create a charged hadron for each track
   - **if** ($E_{\mathrm{calo}} > \sum p_{\mathrm{tracks}}$) create **photons** and/or **neutral hadrons** to account for missing E_{calo}

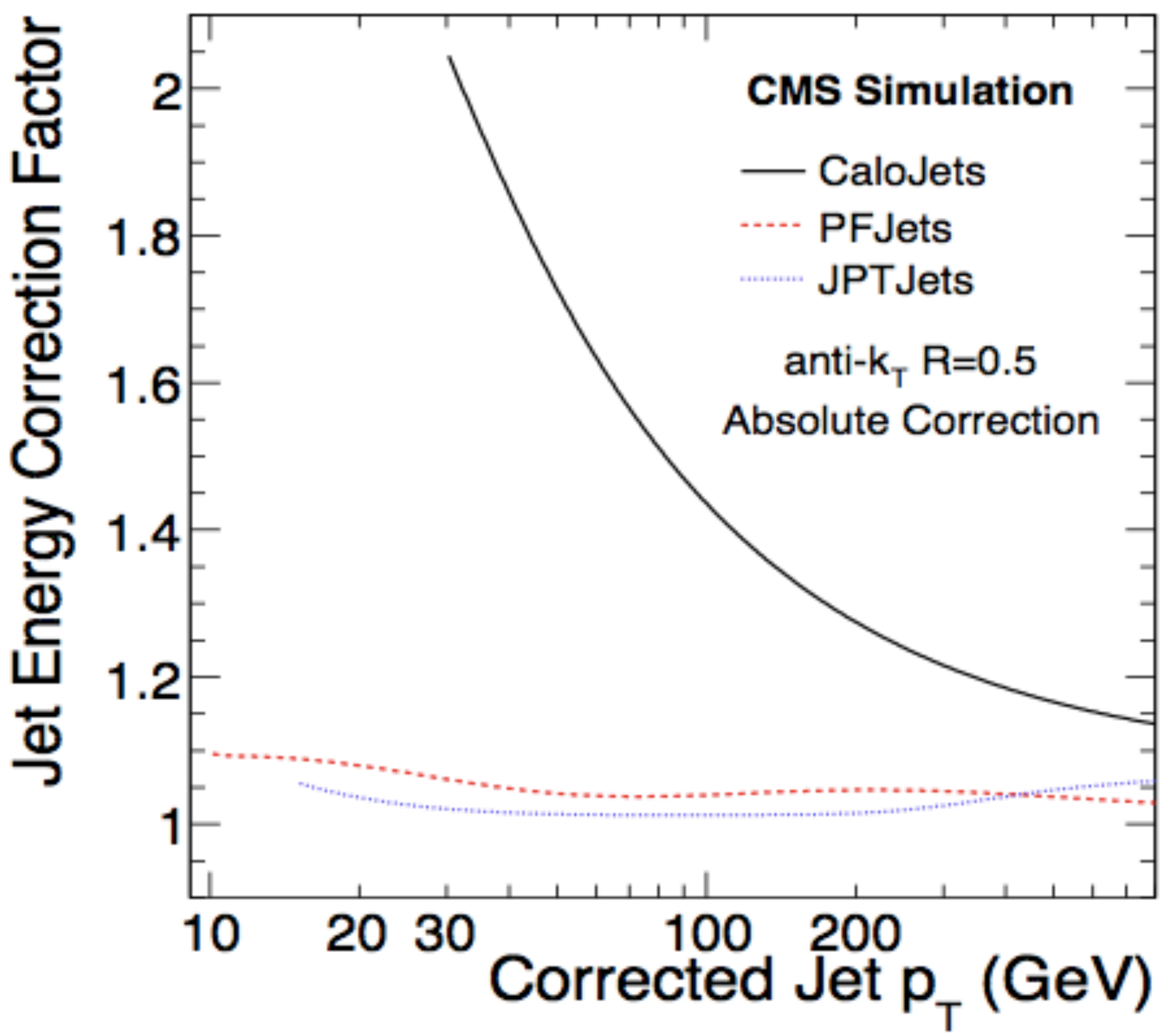
4. For all remaining ECAL (HCAL) clusters not linked to tracks, create a **photon** (**neutral hadron**)



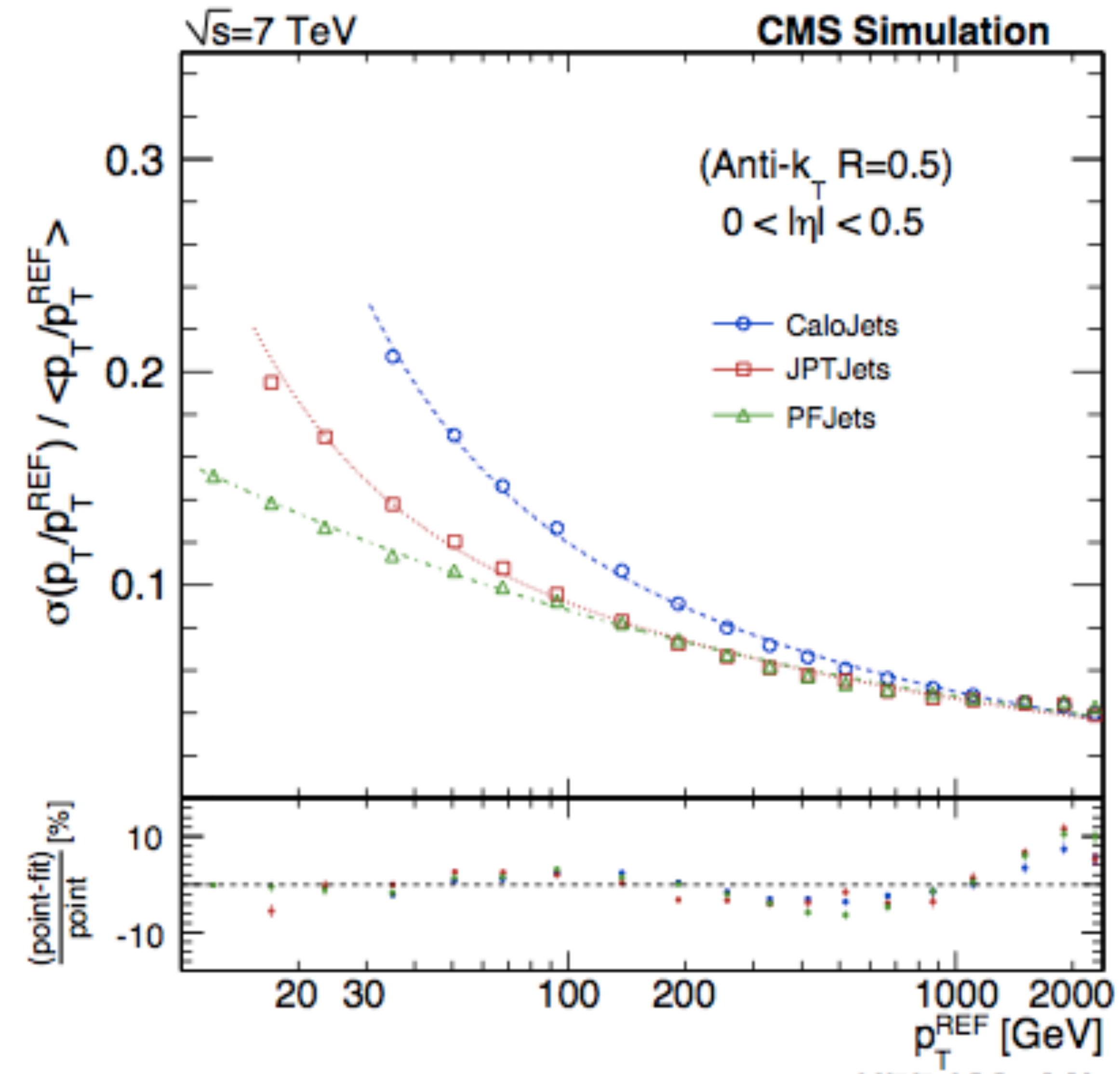
Five Particle Types

PFJet Performance

large uniform response

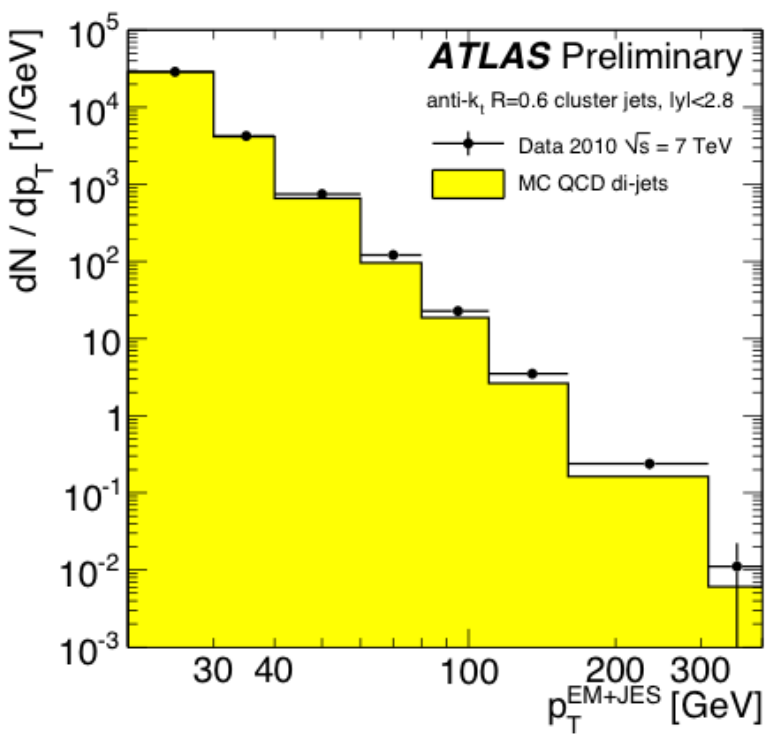


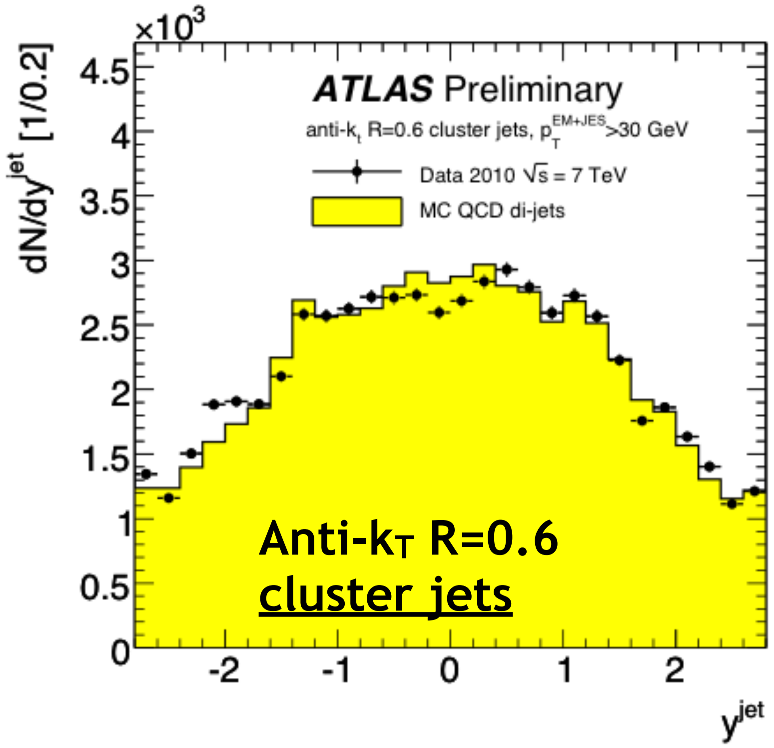
improved resolution

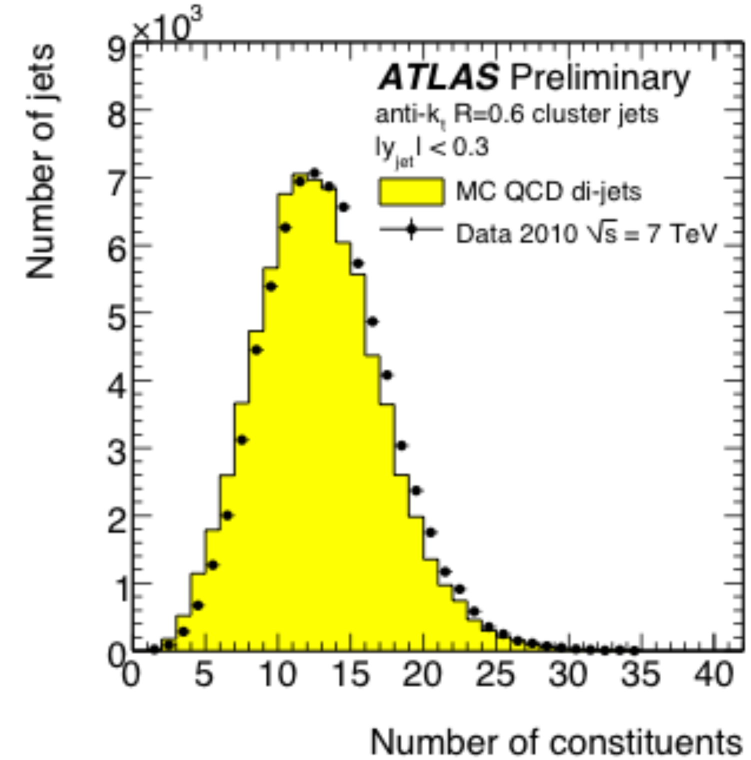


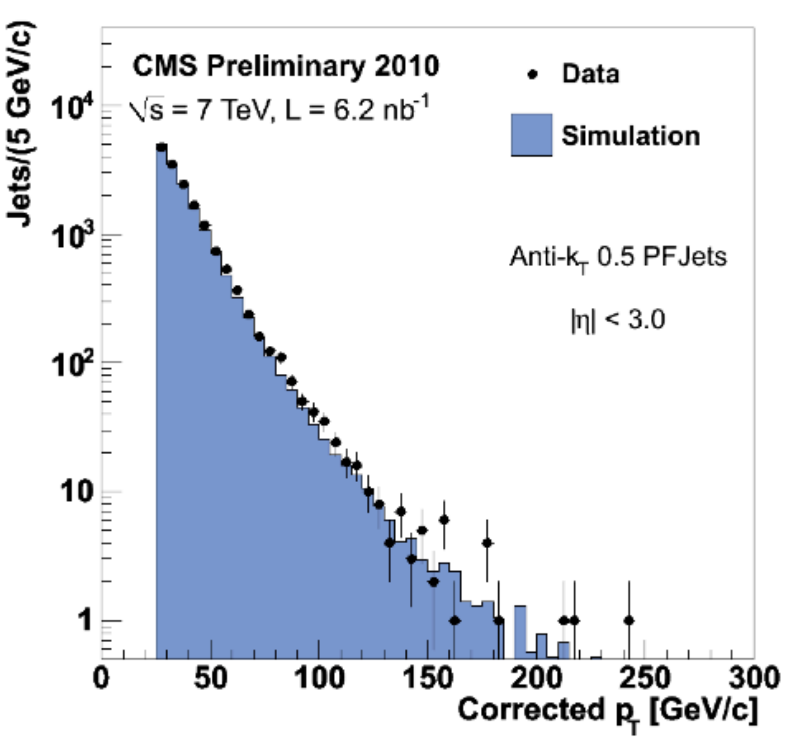
Jets in first 7 TeV collisions at ATLAS & CMS

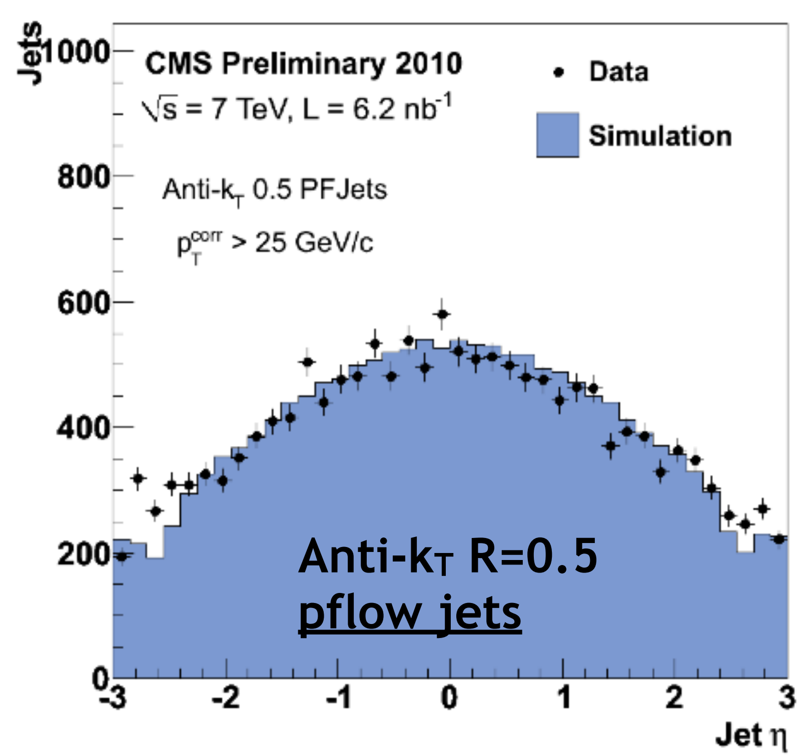
Jet Performance in 7 TeV data

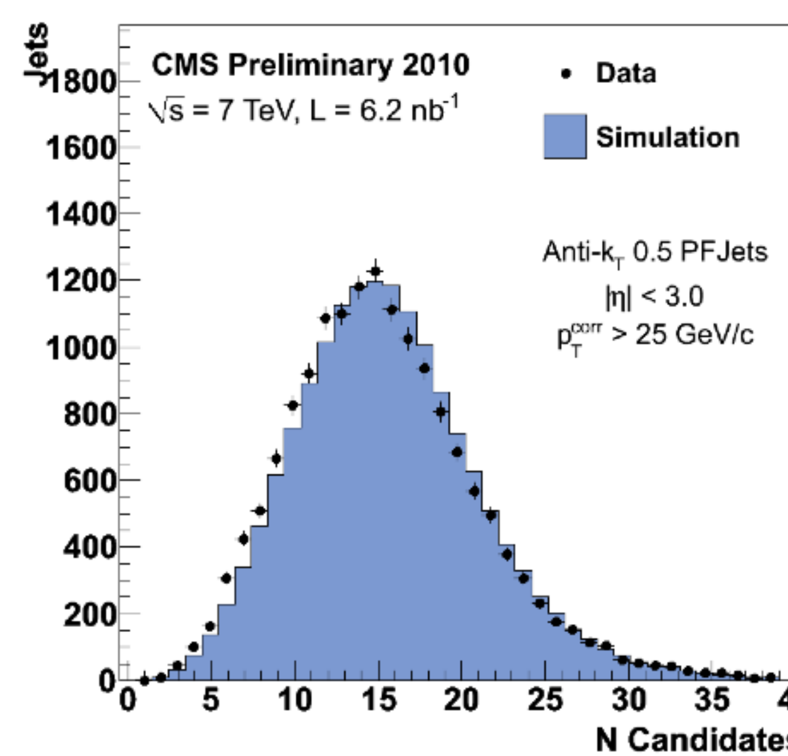






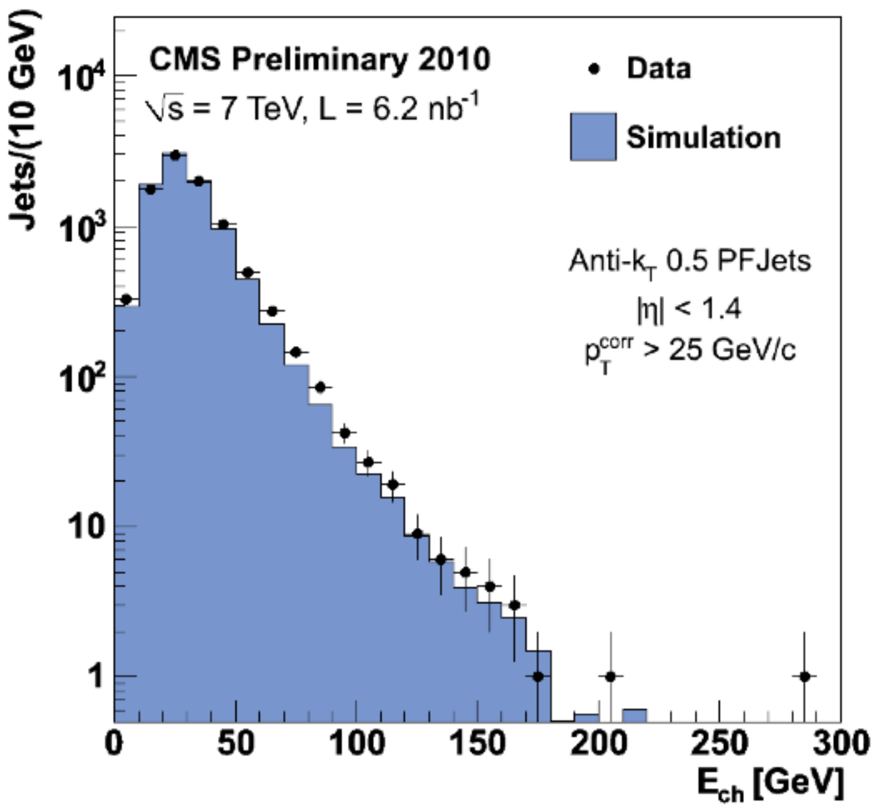




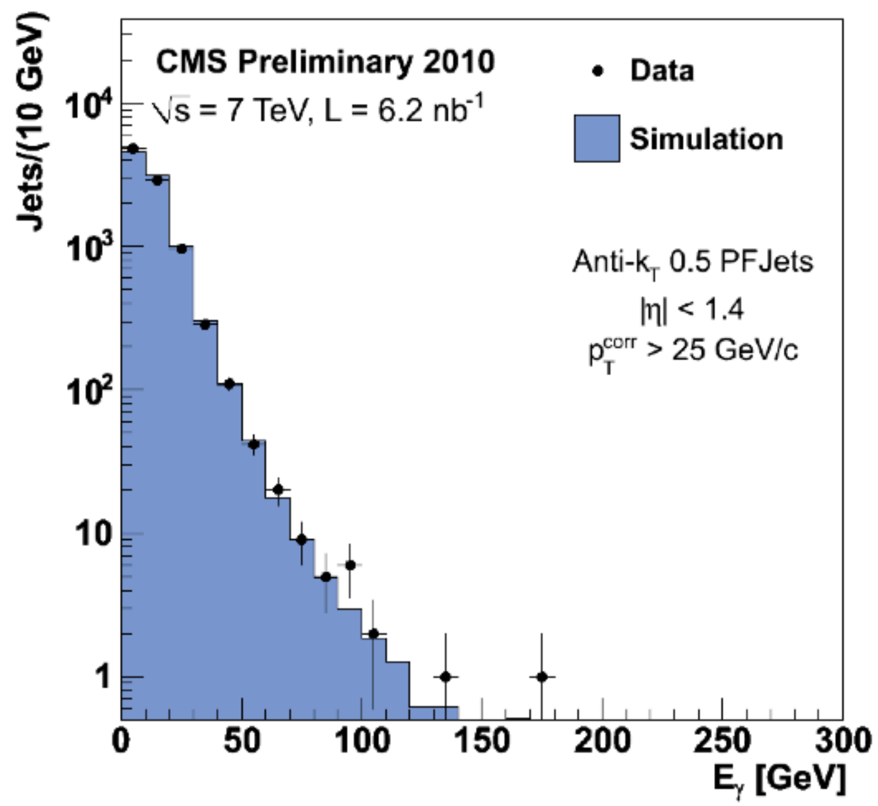


PFJet Composition in 7 TeV data

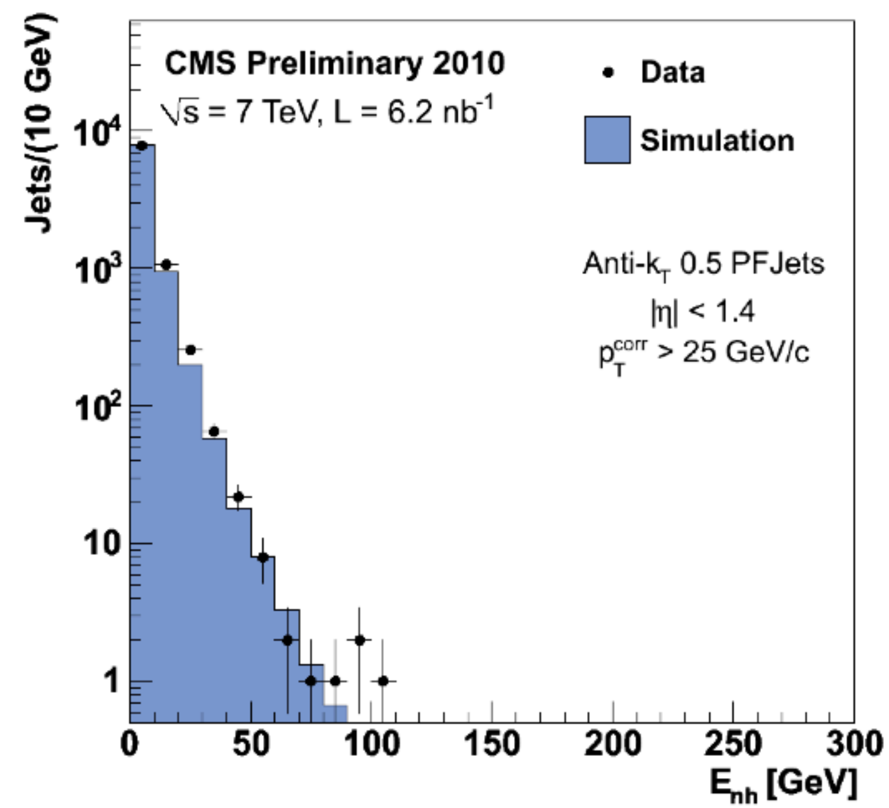
charged hadron
energy fraction



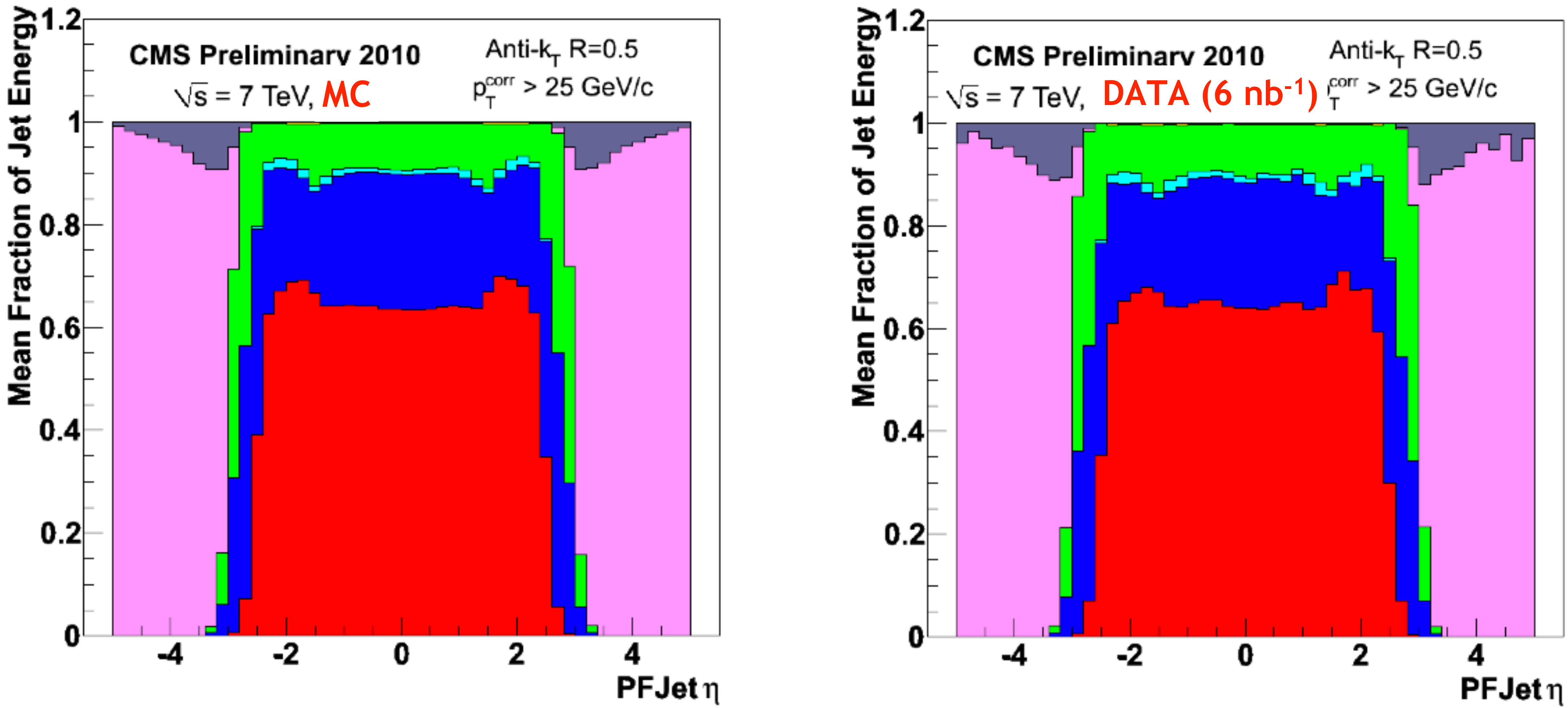
photon
energy fraction



neutral hadron
energy fraction



PFJet Composition (mean)



charged hadrons
photons
electrons
neutral hadrons
HF hadrons
HF EM particles

Summary

- Jets are the observable objects to relate experimental *observations* to theory *predictions* formulated in terms of quarks and gluons

- we observe collimated bunches of stable colorless hadrons, originating from partons after *fragmentation* and *hadronization*: clustering partons & hadrons into jets is supposed to yield two *projections* we can relate to each other

- Among all *cone* and *sequential-clustering* algorithms, the *Anti-kT* algorithm has emerged as the favorite compromise among experimentalists & theorists

  ➡ But is there a physics case for collaborations to maintain a variety of jet algorithms and paramter choices for different final states?

- ATLAS (*topological clusters*) and CMS (*Particle Flow*) have both devised advanced strategies to reconstruct & calibrate input particles to jet clustering

- Given the complexity of jet reconstruction, their *good performance* in first 7 TeV collision data events at both experiments is *amazing*!