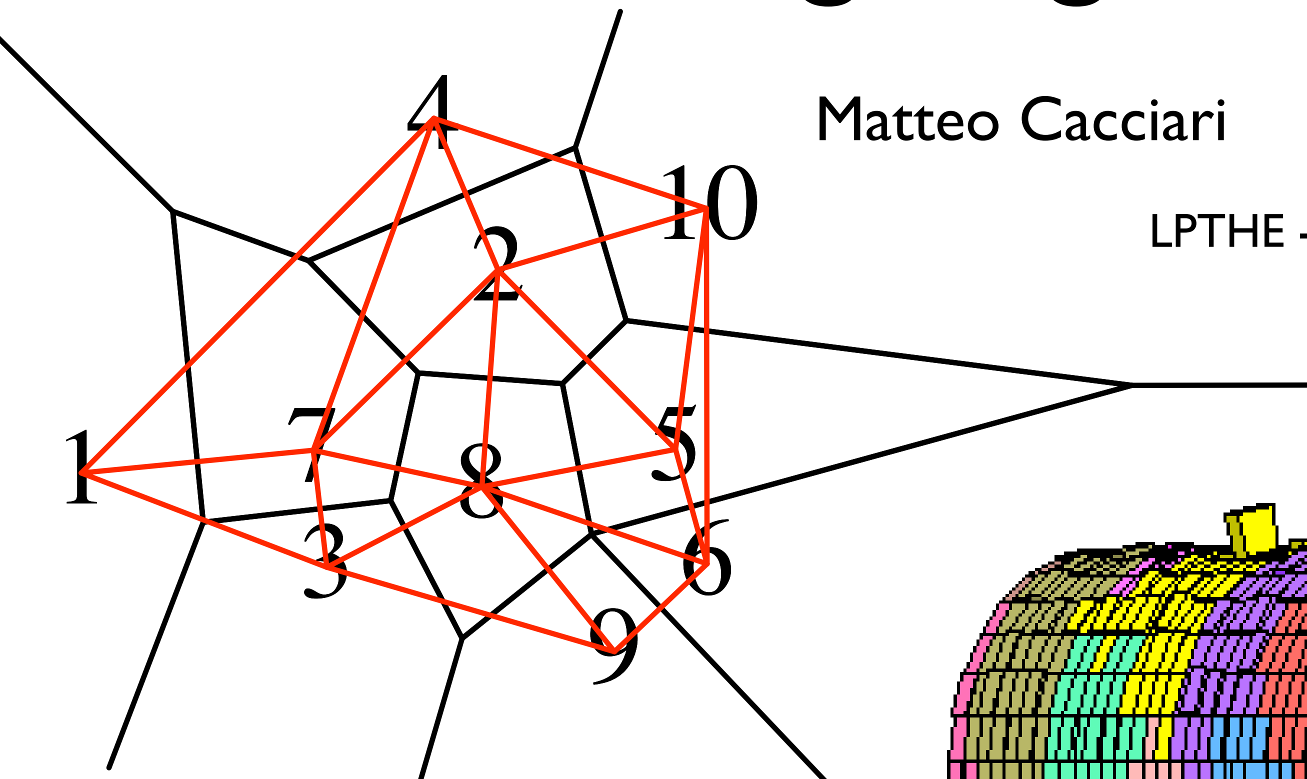
IFAE 2007
Napoli - 11 Aprile 2007

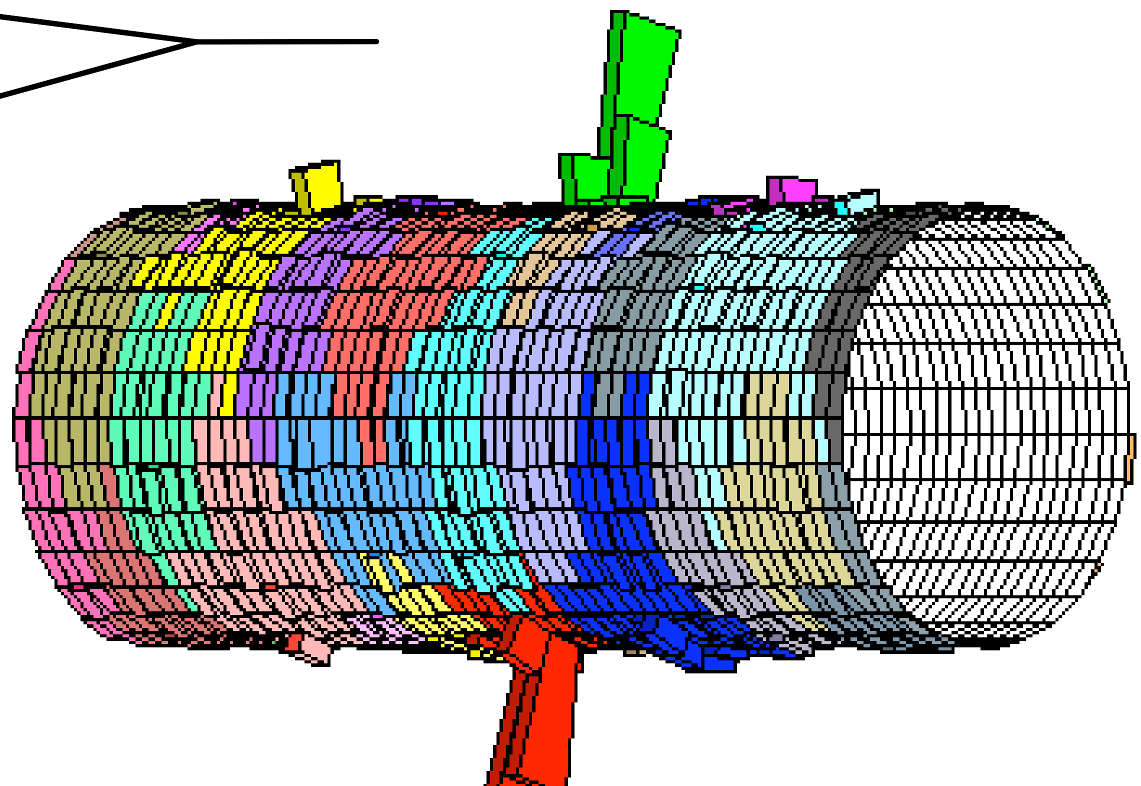
Sviluppi recenti negli algoritmi di jet

Matteo Cacciari

in collaboration with
Gavin Salam and Gregory Soyez

LPTHE - Paris 6,7 and CNRS





Outline

- Recent developments in k_t/Cambridge and in cone jet algorithms
- Definition of '**area**' of a jet
- Use of areas in underlying event/pile-up subtraction

Jet algorithm

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

particles,
4-momenta,
calorimeter towers, ...

jets

Running a jet algorithm gives a well defined physical observable

Requirements: **infrared and collinear safety**
Adding a soft or collinear particle should not change the set of hard jets

Jet Algorithms as of 2005

Two main jet-finder classes:
cone algorithms and **sequential clustering algorithms**

- **Cone-type** algorithms are mainly used at the Tevatron. Extensions of original Sterman-Weinberg idea, i.e. **identify energy flow into cones**.

  Examples: **PxCone**, **JetClu**, **MidPoint**, **SearchCone**...
  - Difficult to tell which is which
  - Many unphysical parameters in definition
  - Not really infrared safe

- **Sequential clustering** algorithms are based on **pair-wise successive recombinations**. Widely used at LEP and HERA.

  Examples: $\mathbf{k_t}$, **Cambridge/Aachen**
  + Simple definition, infrared and collinear safe
  - Slow numerical implementation, typically N^3

Jet Algorithms from 2005 to 2007

MC, G. Salam, G. Soyez

Use of
geometrical methods + **computer science** techniques:

The implementation of sequential algorithms is greatly improved

A practical infrared safe cone algorithm can be defined

MC and G. Salam, hep-ph/0512210 (FastJet)

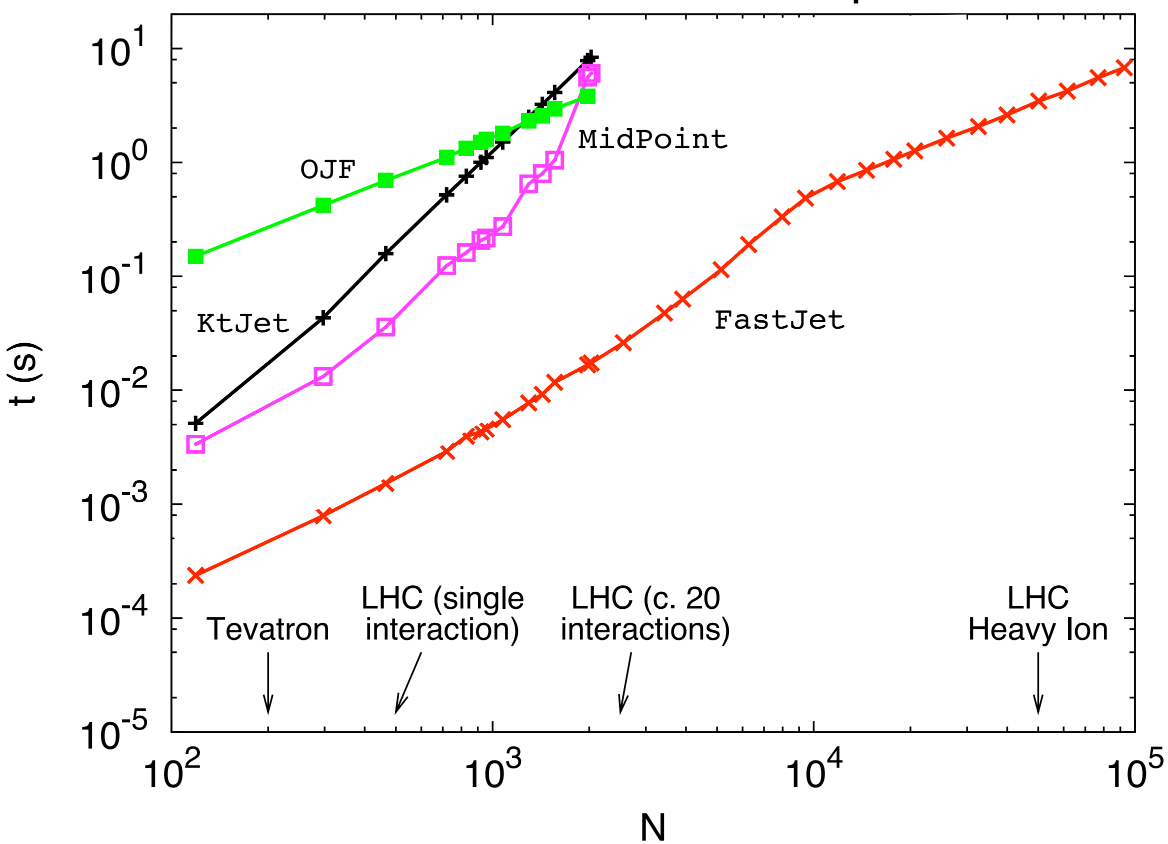
www.lpthe.jussieu.fr/~salam/fastjet

The algorithmic complexity of k_t/Cambridge implementation is lowered from N^3 to **NlnN**

Time taken to cluster N particles:

10 s

1 ms



Thousands of particles can now be clustered in < 1 second

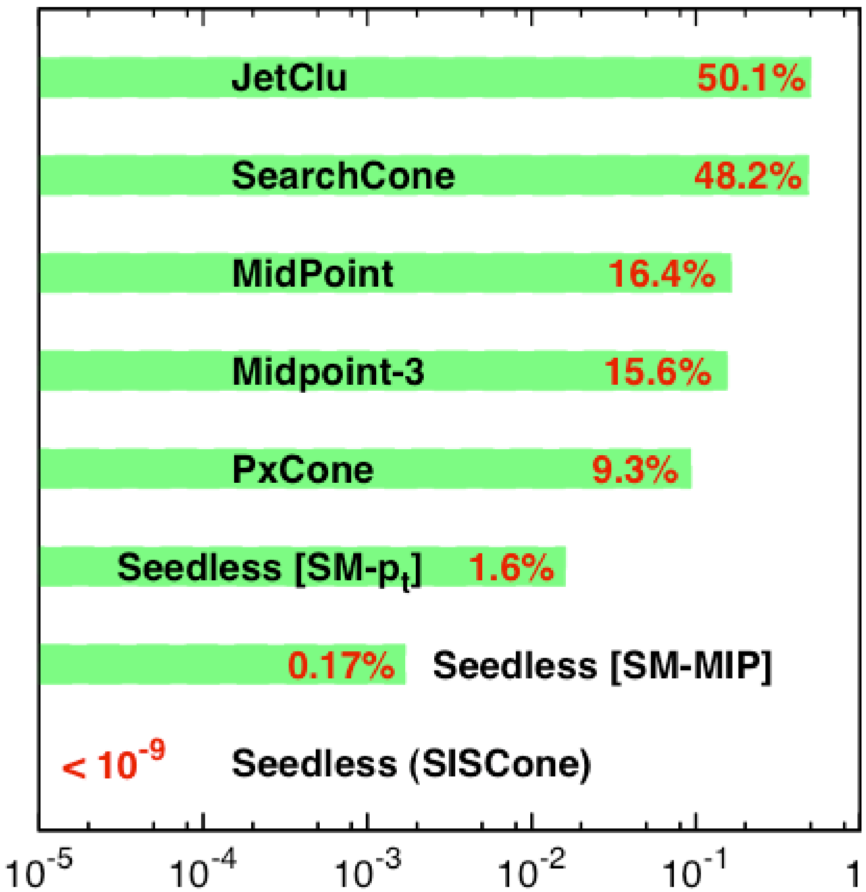
Results **identical** to older k_t/Cambridge implementations. Just a lot faster

G. Salam and G. Soyez, arXiv:0704.0292

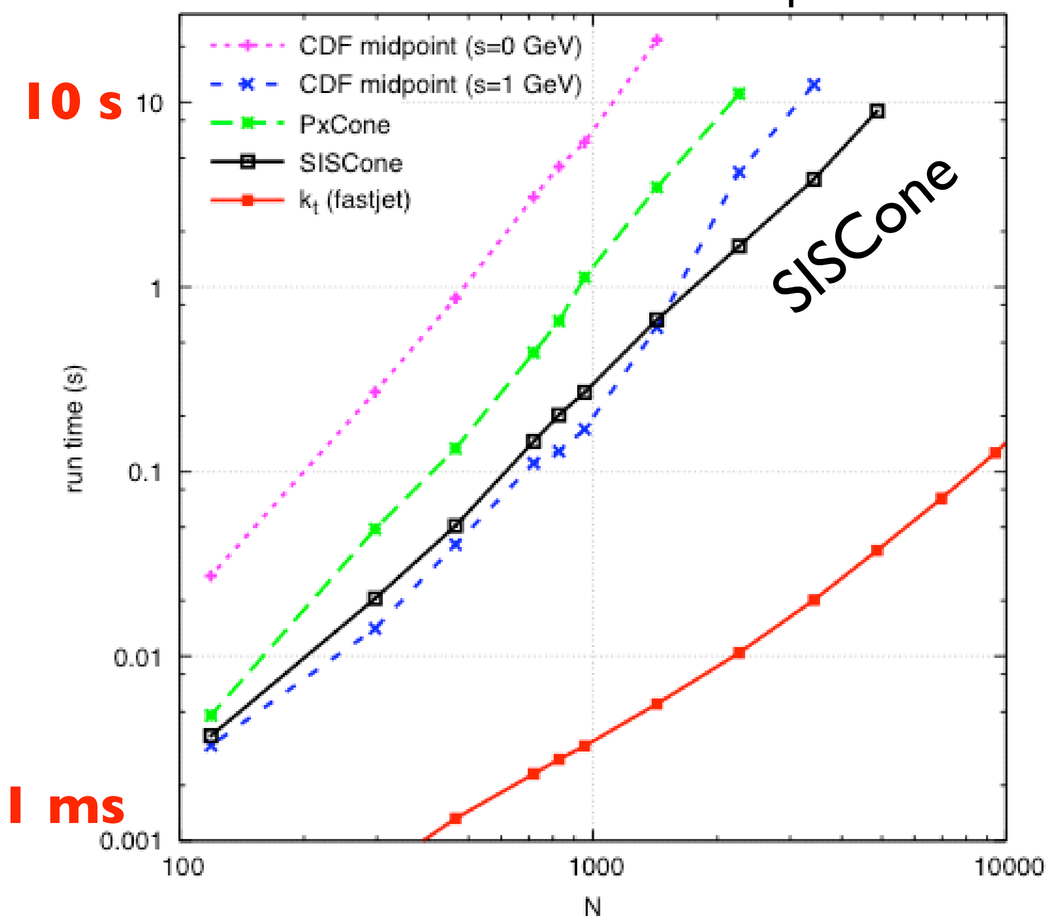
projects.hepforge.org/siscone

An infrared safe cone jet algorithm, **SISCone**, is defined by introducing a manageable seedless search for all stable cones

Fraction of events failing the IR safety test



Time taken to cluster N particles:



This is a **new** cone algorithm (results can differ)

The new (unpublished) stuff

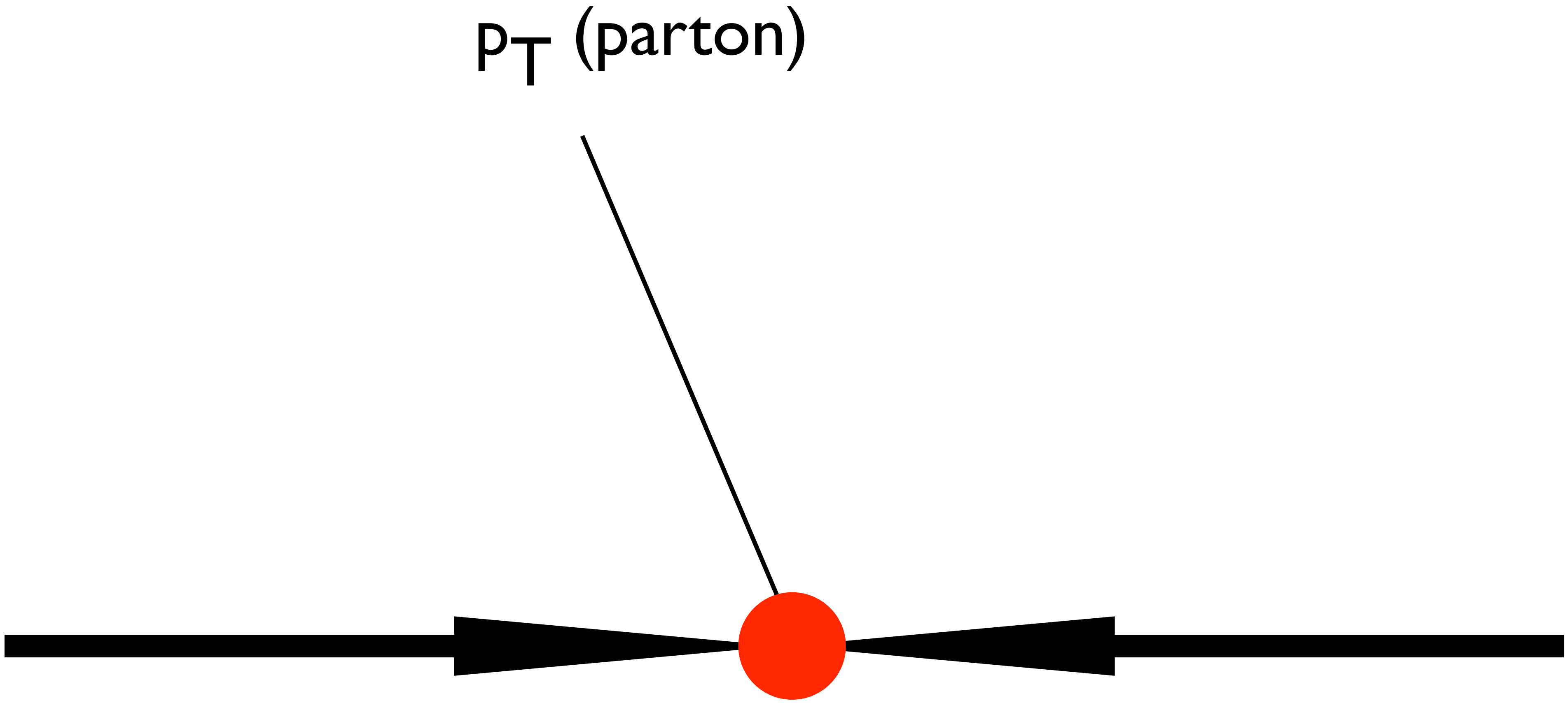
So far, incremental improvements

(better cone, faster k_t/Cambridge)

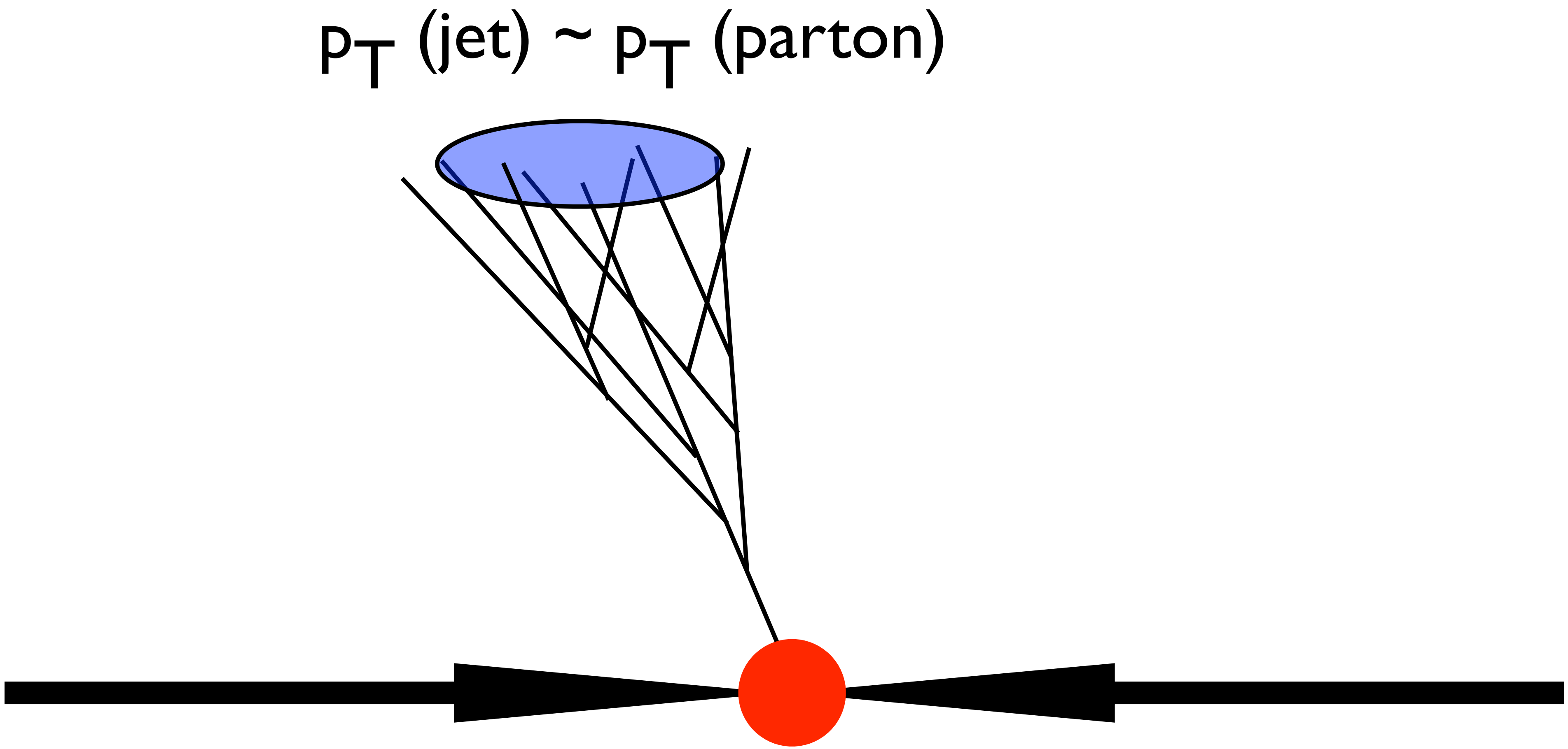
Next, a new concept:

the **area** of a jet

A simple event

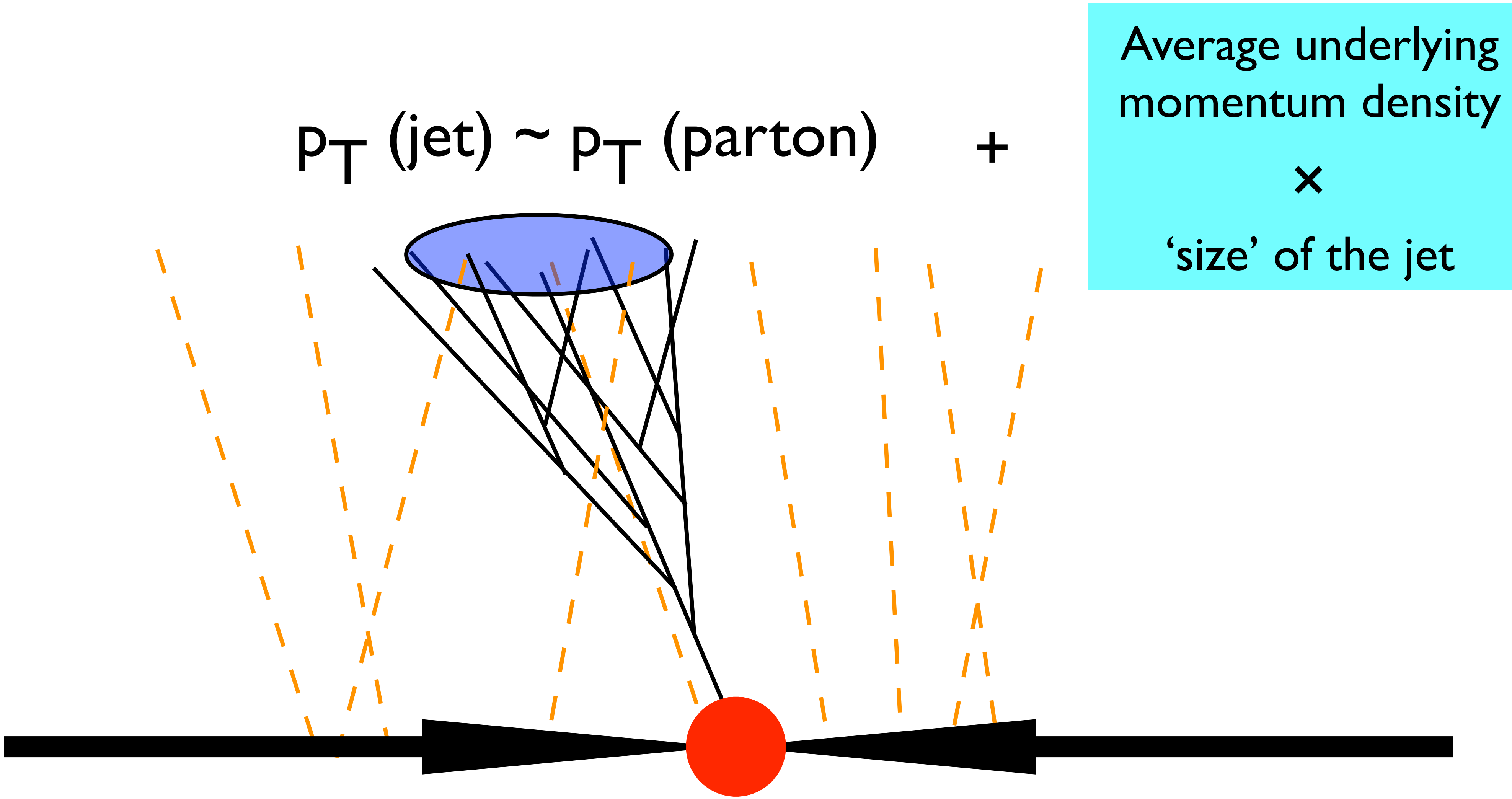


A simple event



The parton radiates, but we can usually collect most of its momentum into a jet

A messier event

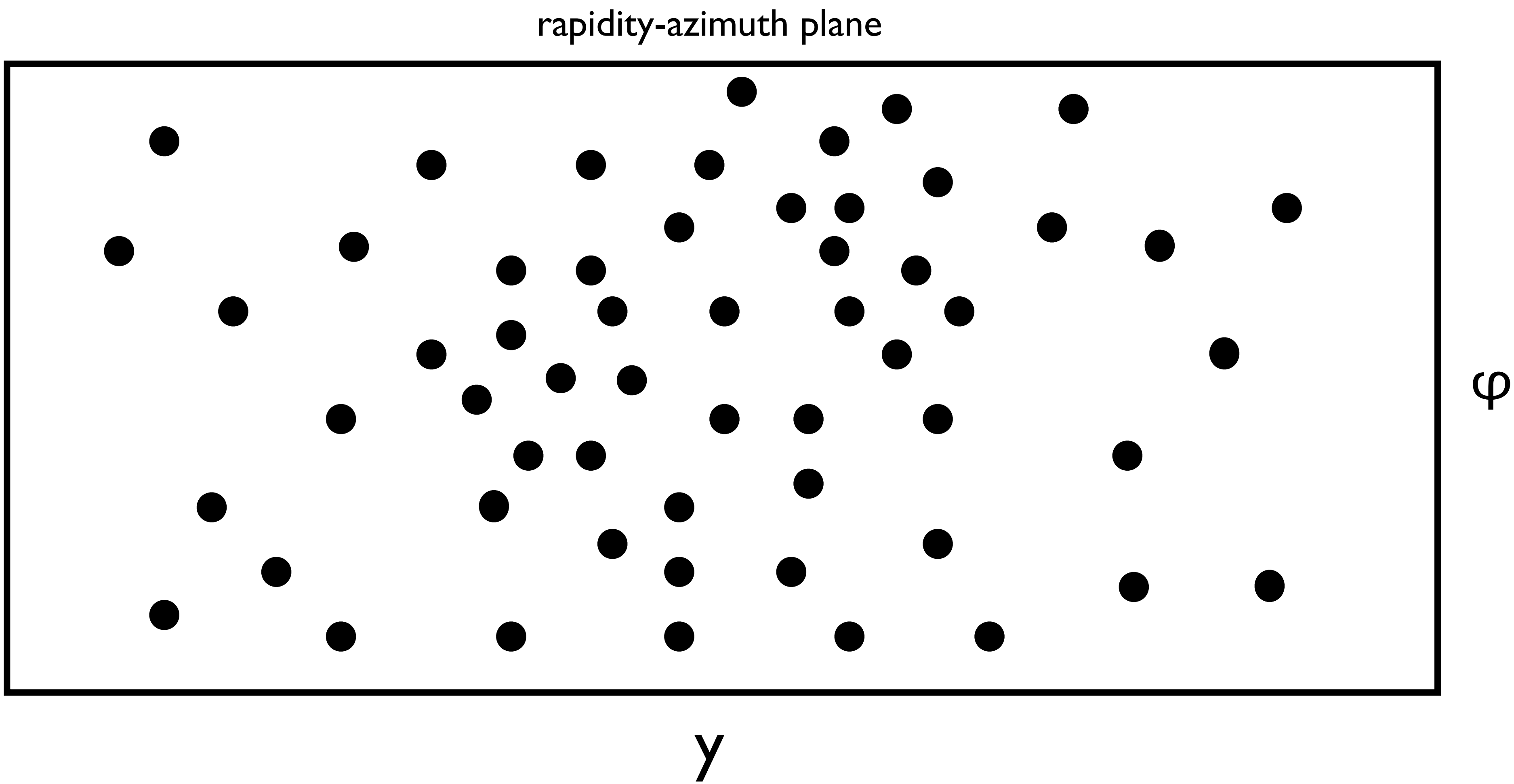


Can we get to know the momentum density of the radiation?

Can we subtract it from the jet to find the parton momentum?

What is the 'size' of a jet?

Consider an event made up of a number of particles

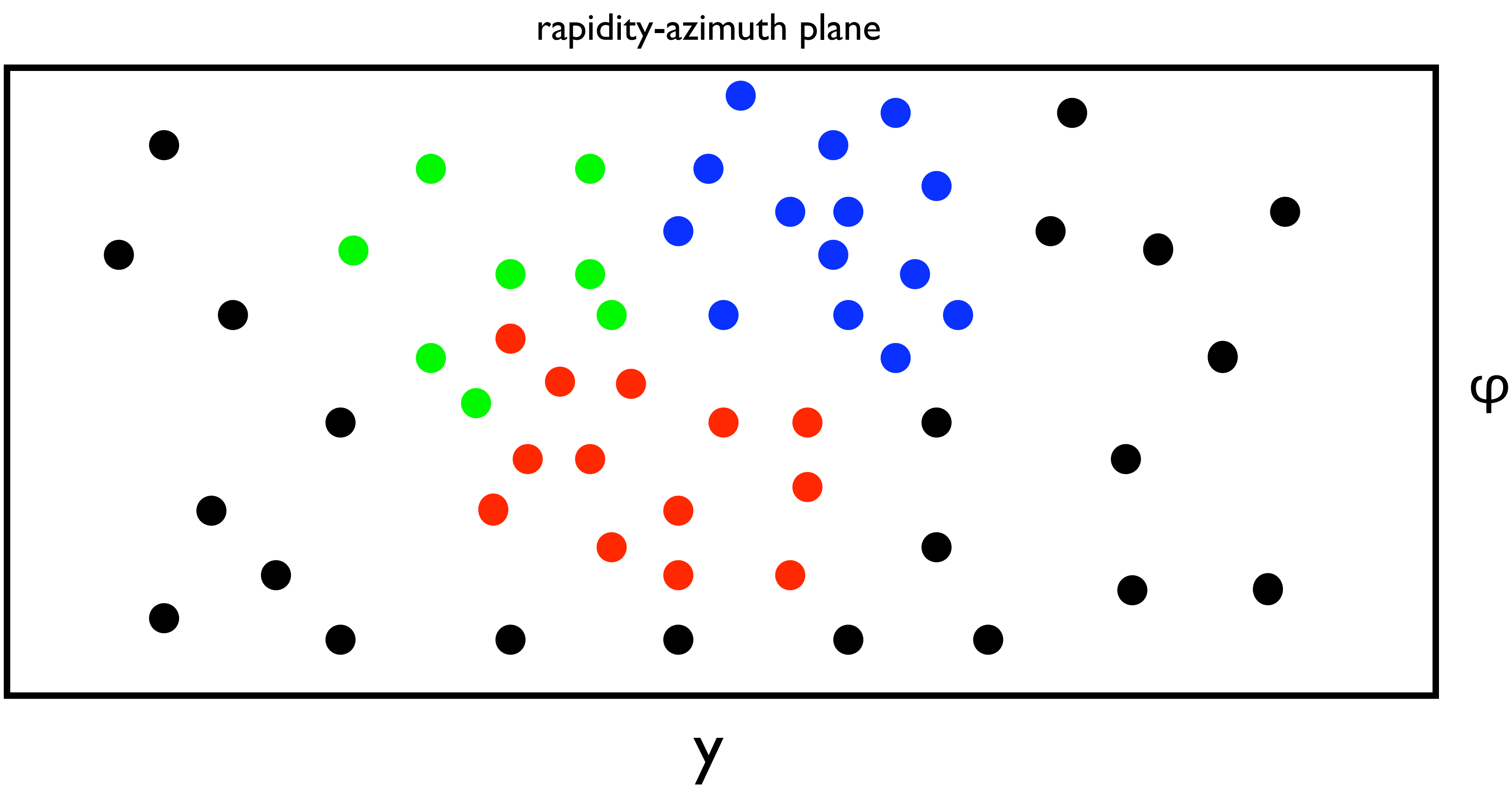


$\{p_i\}$ → $\{j_k\}$

particles — jet-finder algorithm — jets

What is the ‘size’ of a jet?

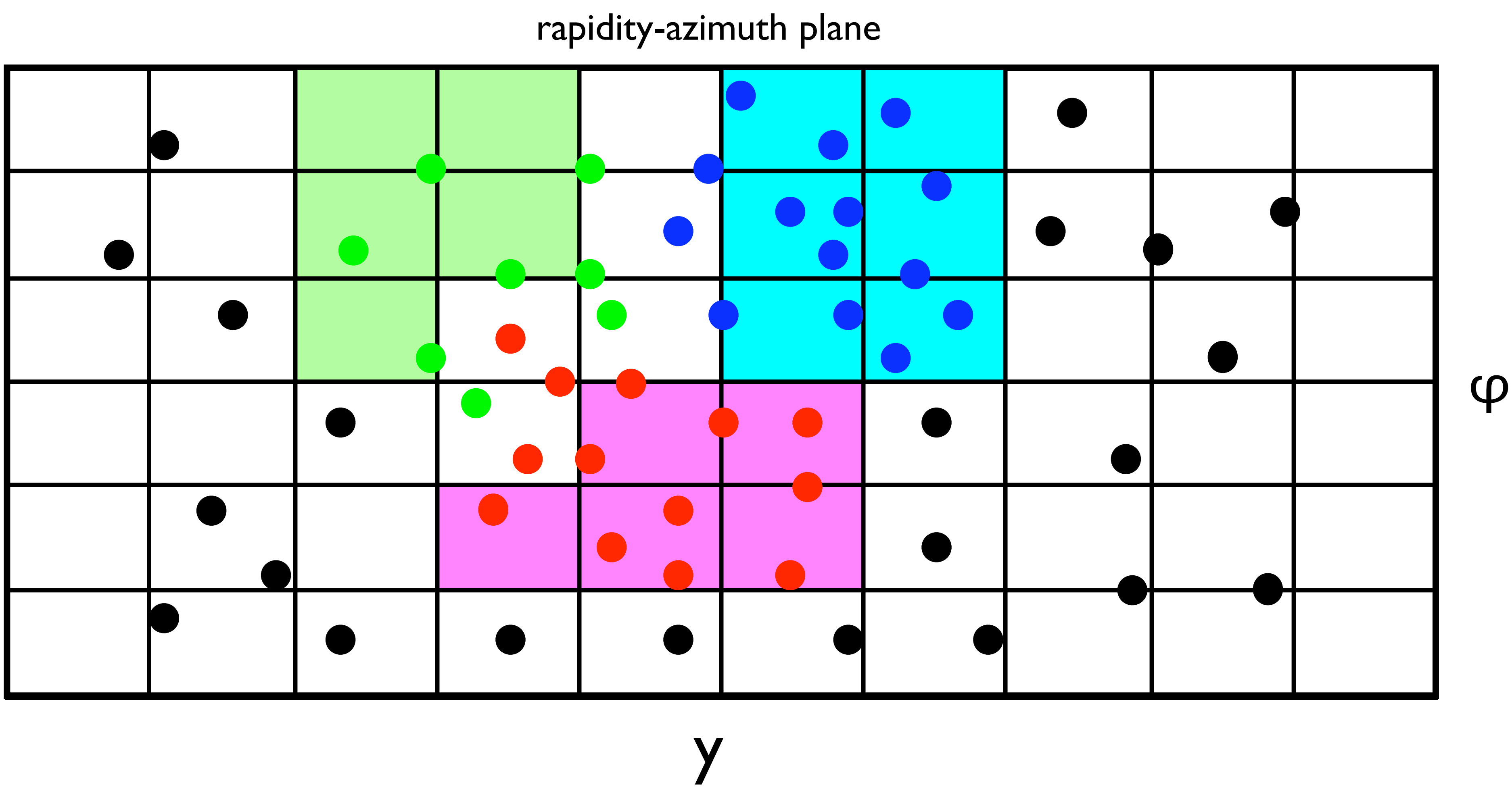
The clustering procedure assigns each particle to a jet:



But... where exactly does a jet end, and another begins?

Jet Area

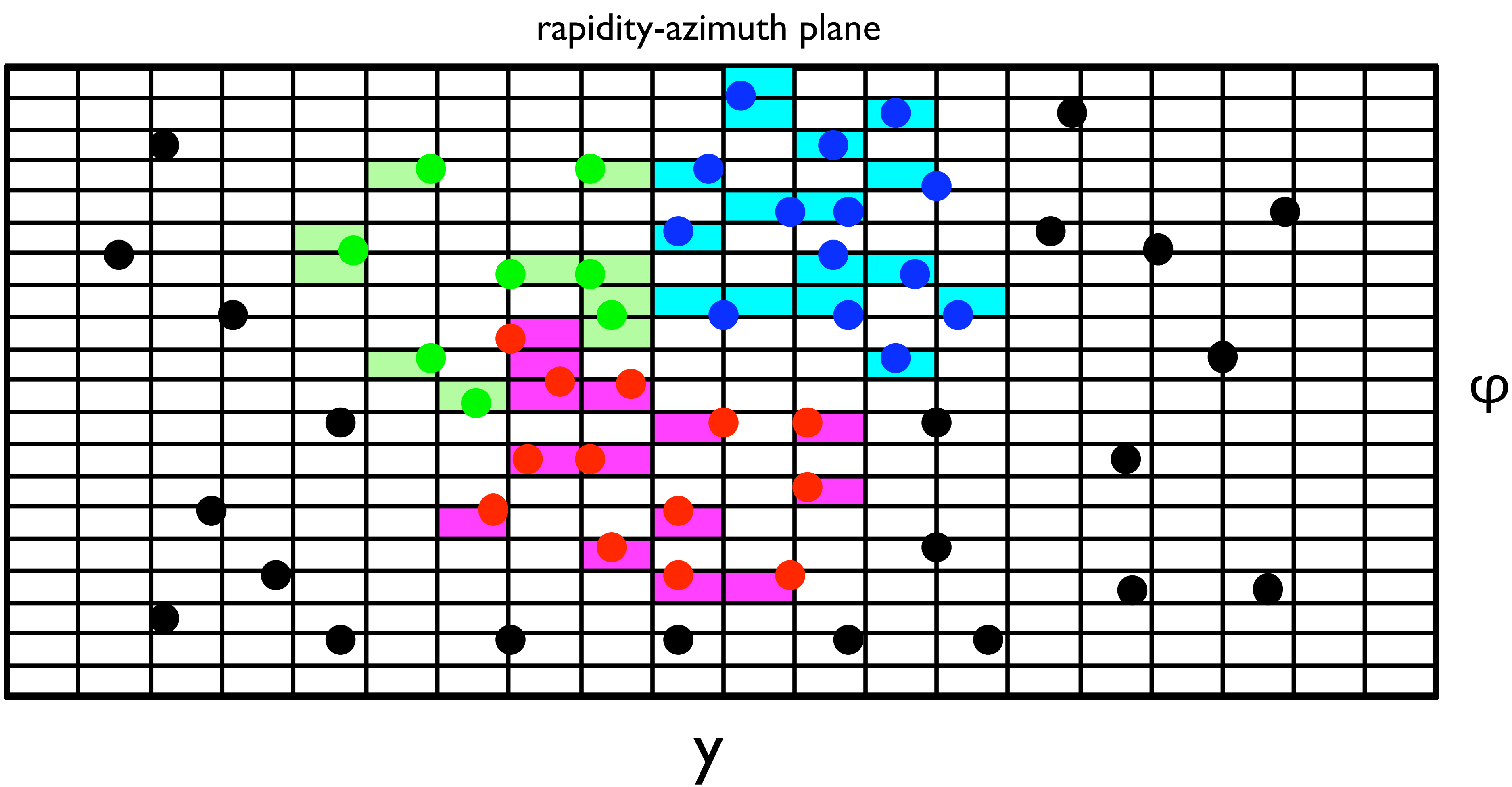
One idea: tile the plane, count the cells of a jet, sum the areas



But what do I do when different jets share a cell?

Jet Area

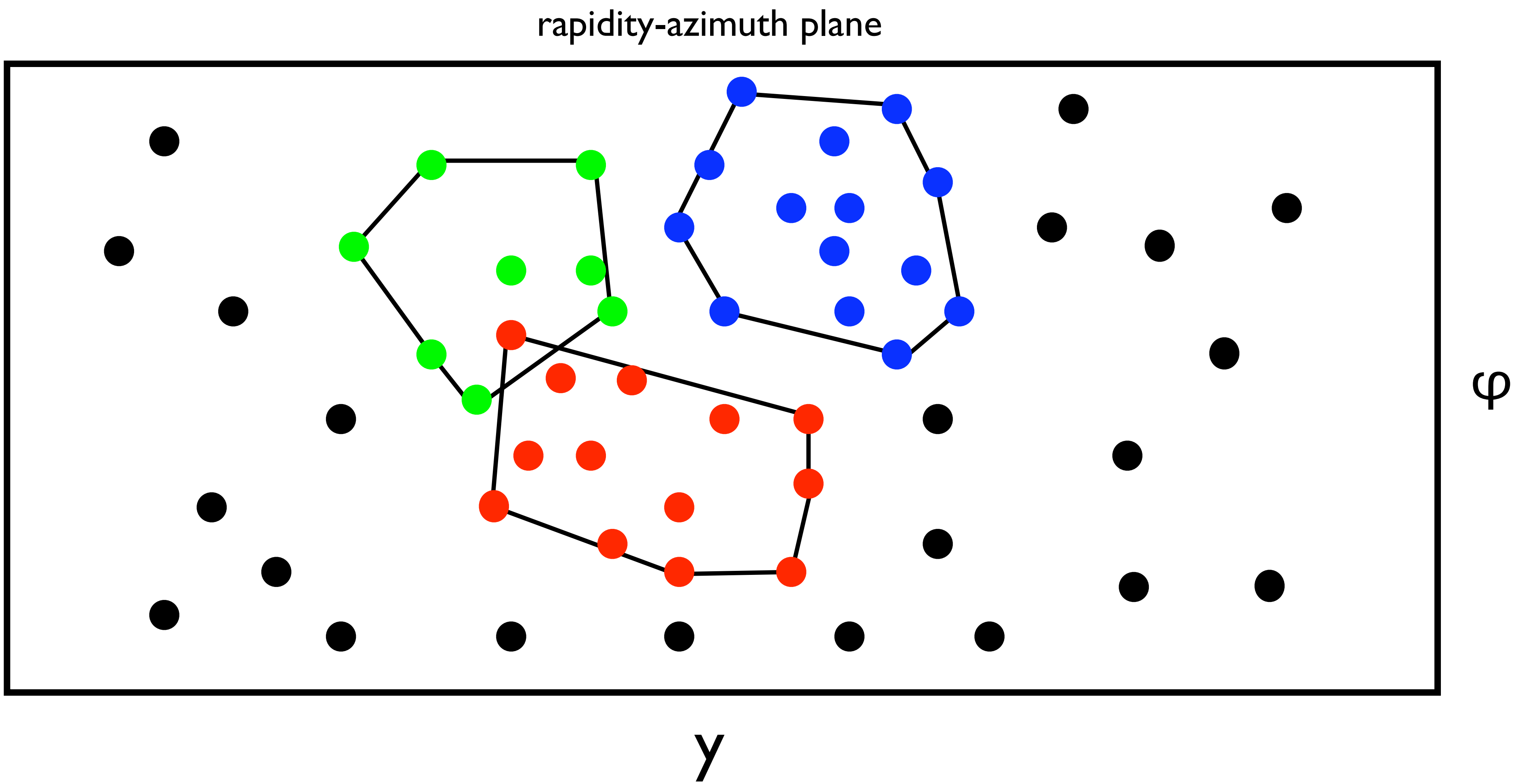
Obviously, make the cells smaller to improve accuracy



Unfortunately, particles being pointlike, the area tends to zero!

Jet Area

Next try, use the **convex hull**



But what do I do if they overlap?

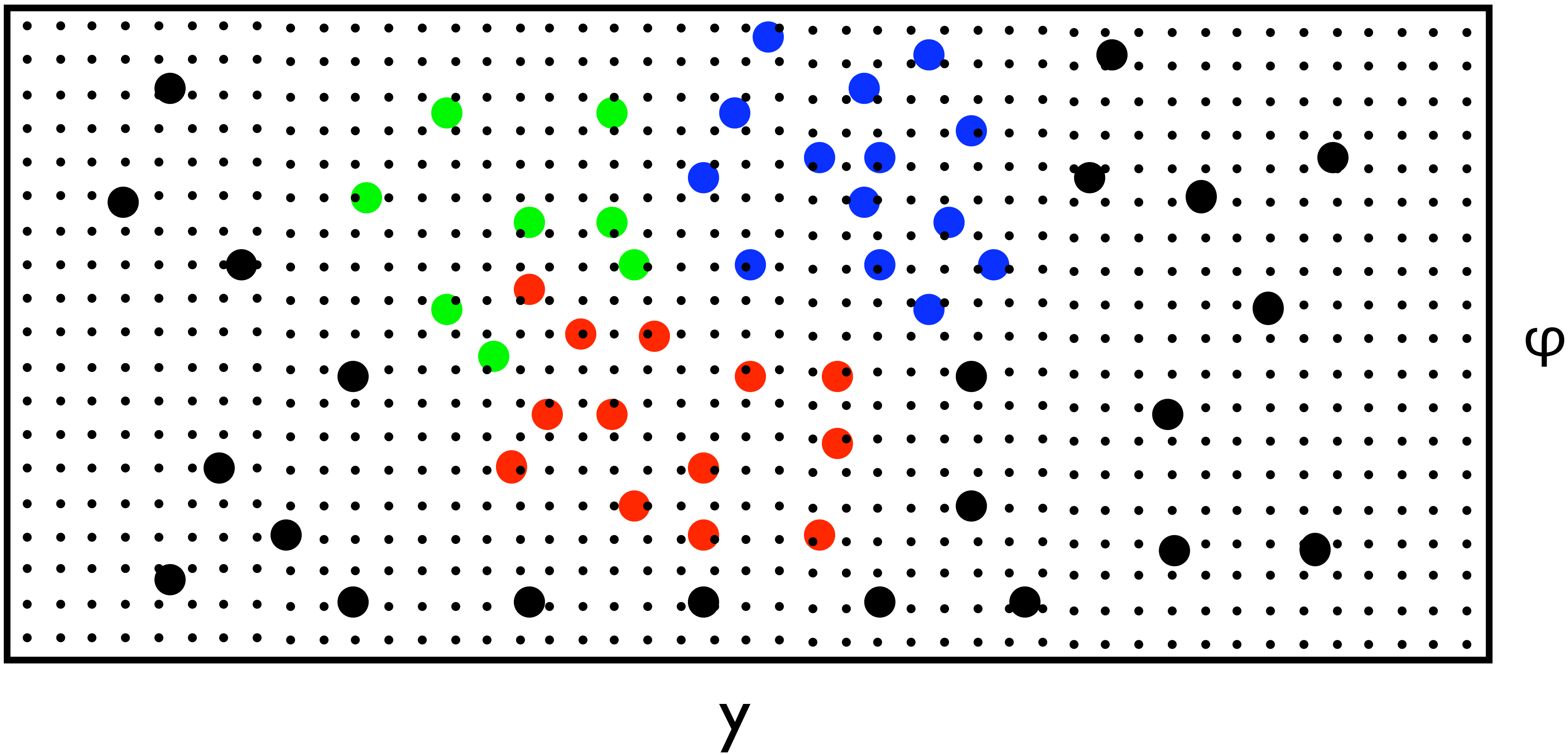
Moreover, what about the 'no man's land' ?

The Active Jet Area

We propose the following definition:

The 'active area' of a jet is (proportional to) the number of uniformly distributed infinitely soft particles that get clustered in it

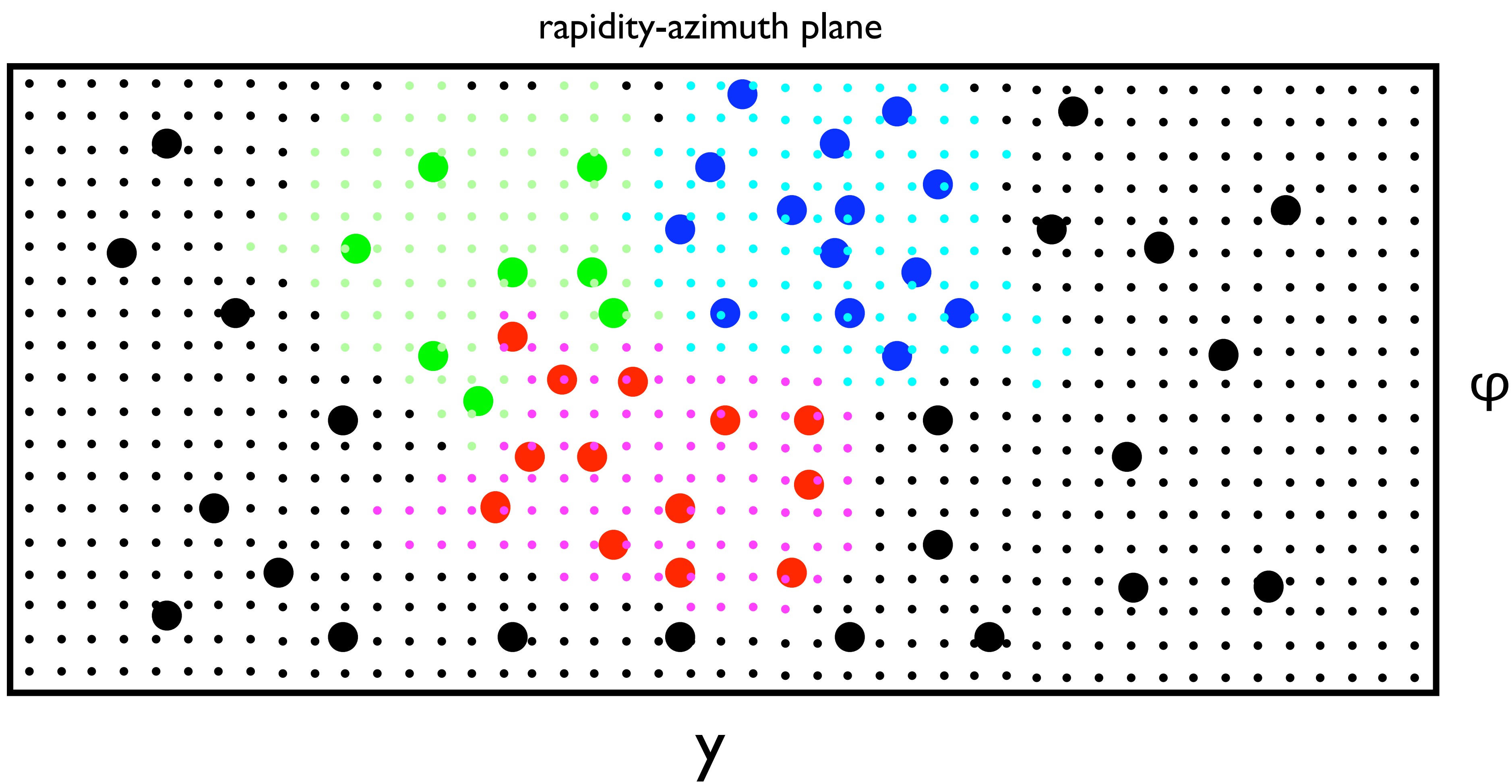
rapidity-azimuth plane



The Active Jet Area

After the clustering, a given set of ghosts belong to each jet

Their number (times the average area of a single ghost) defines the **catchment area** of the jet



The Active Jet Area

The definition of **active area** mimics the behaviour of the jet-clustering algorithms in the presence of a large number of randomly distributed soft particles

Tools needed to implement it:

1. An **infrared safe jet-finder** (the ghosts should not change the jets)
2. A reasonably **fast implementation** (we are adding thousands of ghosts) [$O(10^4)$]

Both these characteristics are found in kt and Cambridge/Aachen jet-finders (as implemented in FastJet) and in SISCone

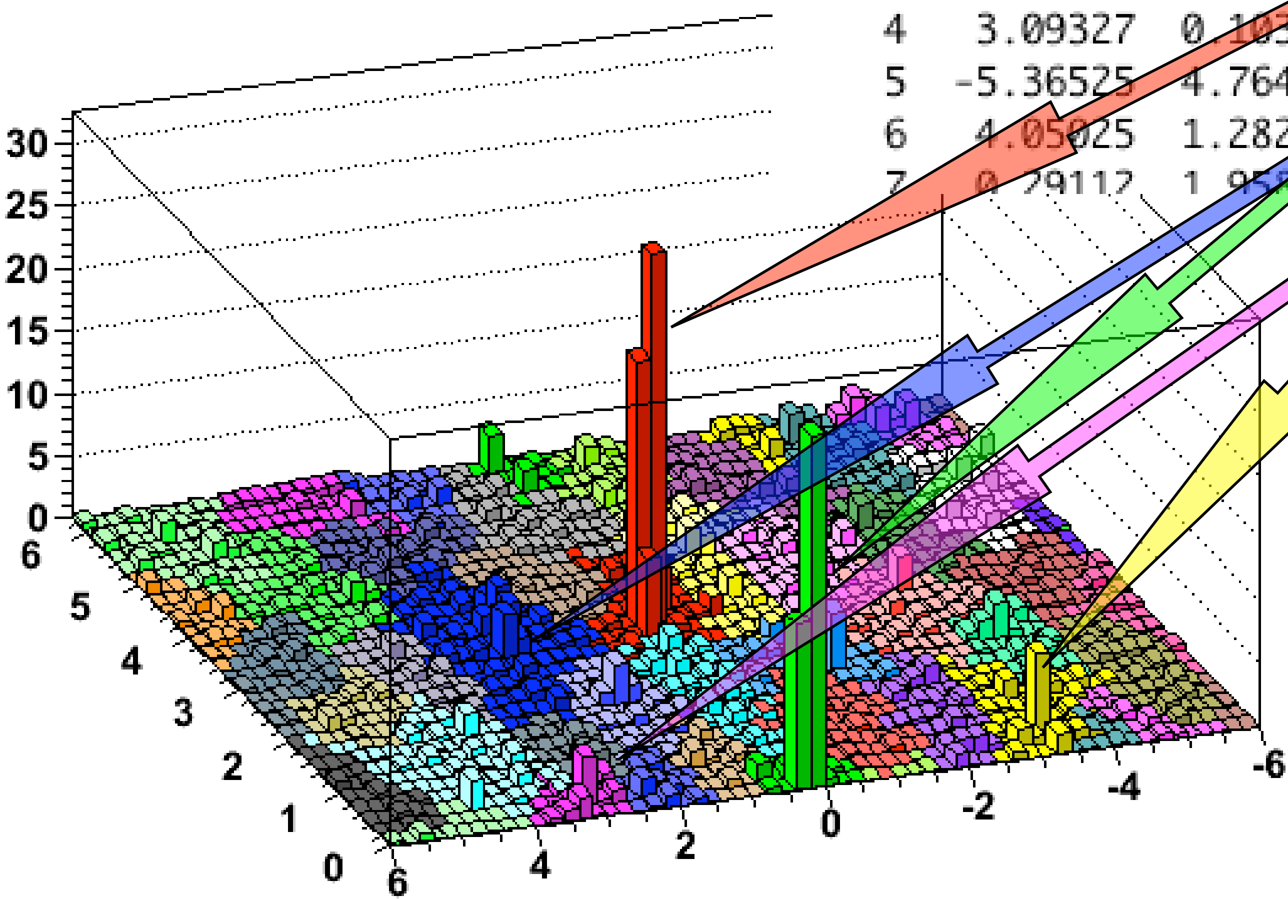
[~ 0.1 s] [~ 100 s]

www.lpthe.jussieu.fr/~salam/fastjet
projects.hepforge.org/siscone

The Active Jet Area

A concrete example:
a 50 GeV di-jet event at the LHC with pile-up
(10 min-bias events added)

```
iev 0 (irepeat 24): number of particles = 1428
strategy used =  NlnN
number of particles = 9051
Total area: 76.0265
Expected area: 76.0265
 ijet    eta        phi         Pt          area  +-   err
    0   0.15050  3.24498      69.970       2.625 +-  0.020
    1   0.18579  0.13150      59.133       1.896 +-  0.020
    2   2.33840  3.23960      31.976       4.749 +-  0.028
    3  -3.41796  0.52394      26.595       3.084 +-  0.021
    4   3.09327  0.10350      20.072       2.688 +-  0.023
    5  -5.36525  4.76491      19.596       2.780 +-  0.012
    6   4.05025  1.28279      15.561       3.592 +-  0.028
    7   0.79112  1.95835      14.566       2.114 +-  0.018
```



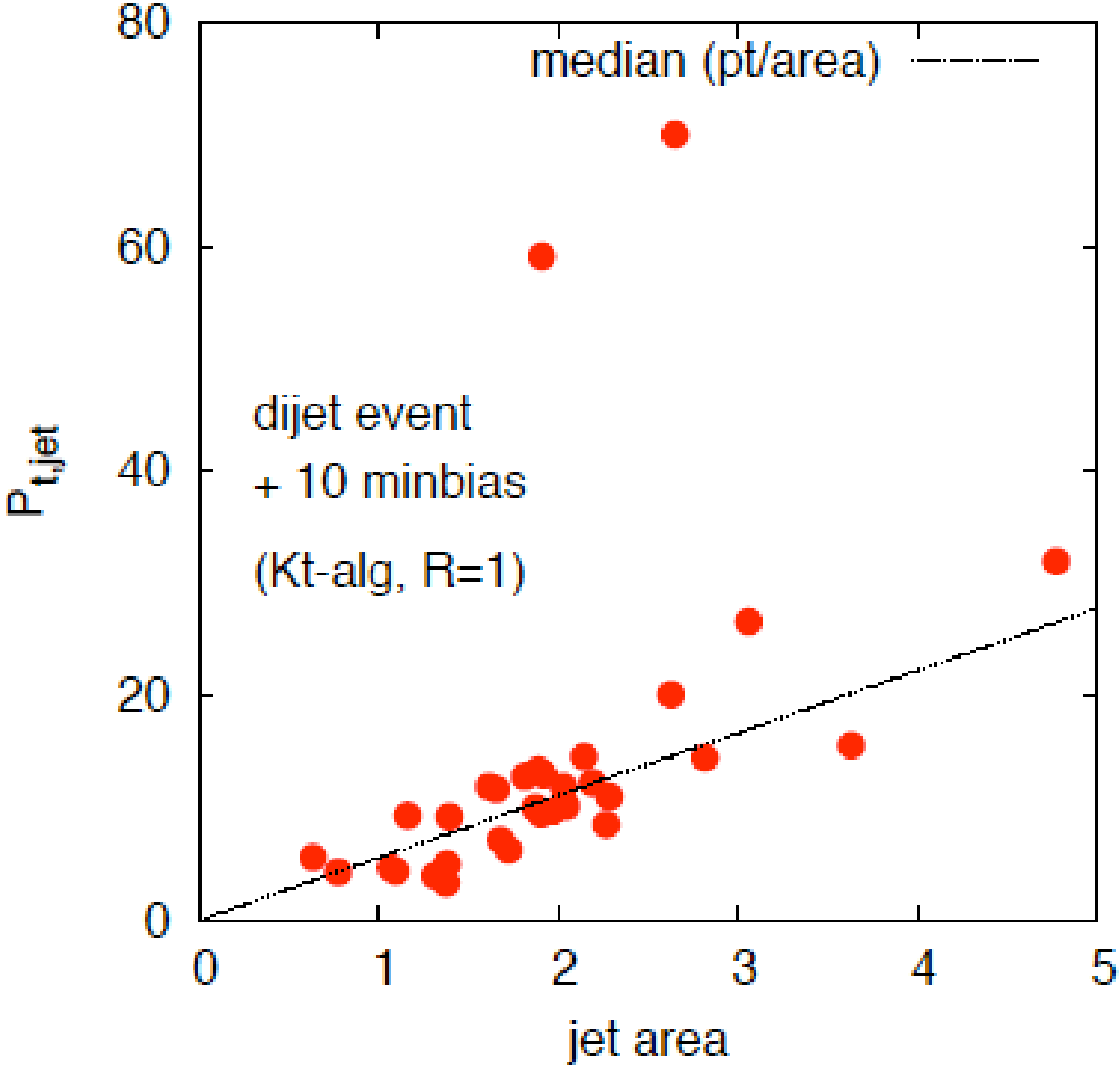
Plan (i.e. ok, I have an area. What do I do with it now?)

- A proper operative definition of **jet area** can be given
- When a hard event is superimposed on a **roughly uniformly distributed background**, we can determine the noise density **ρ** (and its fluctuation) on an event-by-event basis
- Once measured, the background density can be used to correct the transverse momentum of the hard jets:

$$p_T^{\text{hard jet, corrected}} = p_T^{\text{hard jet, raw}} - \rho \times \text{Area}_{\text{hard jet}}$$

But how to determine ρ?

Areas distribution



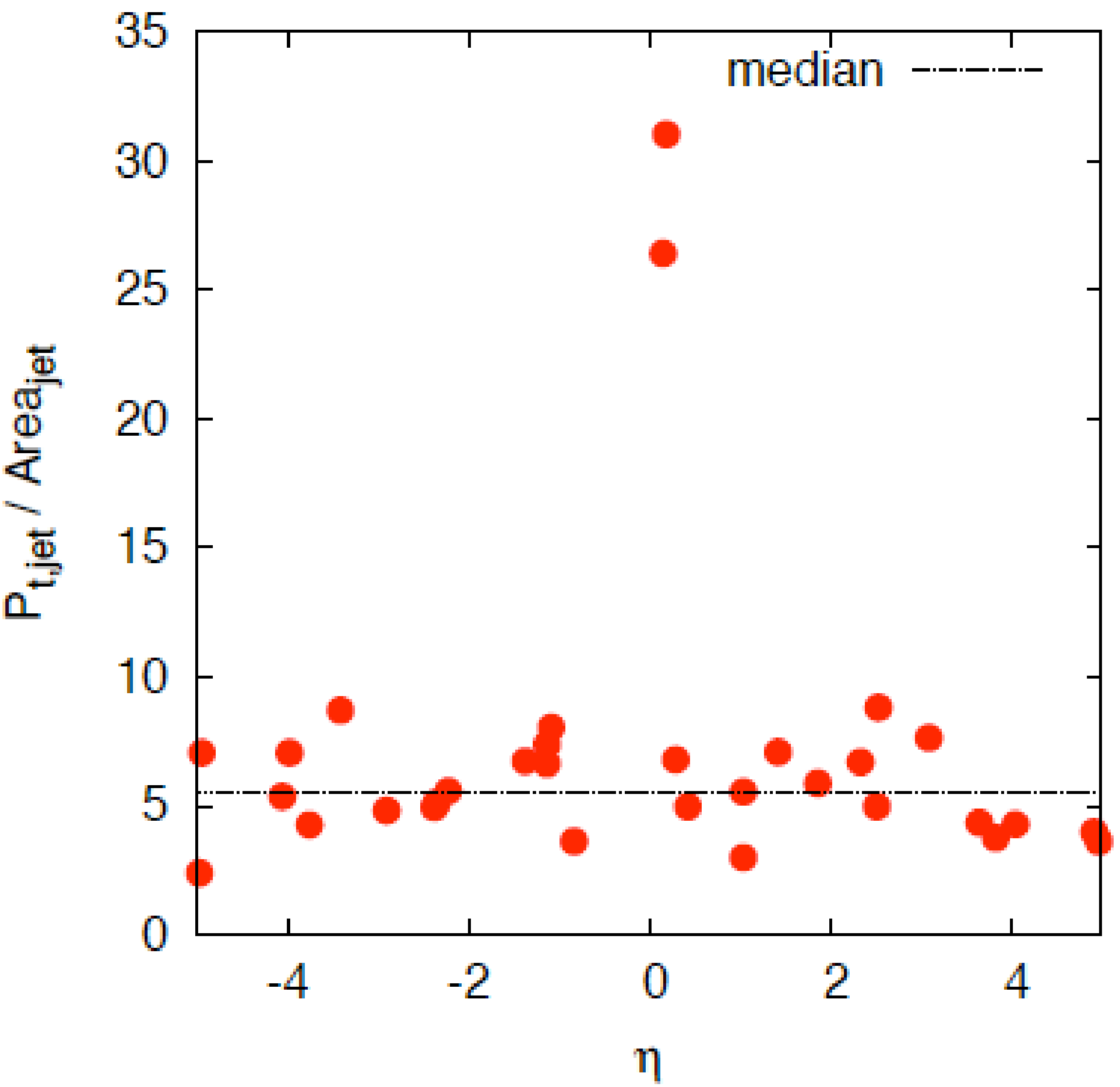
The jets adapt to the surrounding environment

They can have very different areas

Area vs. p_T

Key observation:

p_T/Area is fairly constant, except for the hard jets



The distribution of background jets establishes its own average momentum density

(given, e.g., by the **median** of p_T/area)

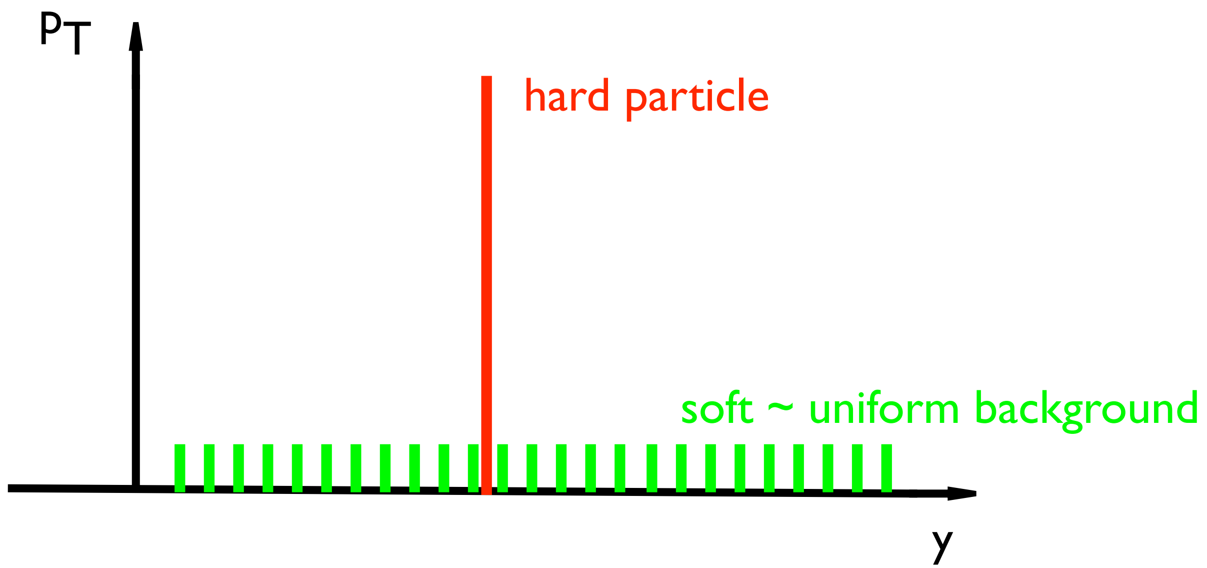
(NB. this is true on an event-by-event basis)

[Implemented in FastJet > v2.0]

Does it work? A toy model

Consider a uniform distribution of soft particles,
e.g. 10000 in the rapidity range [-4,4], with p_T = 1 GeV

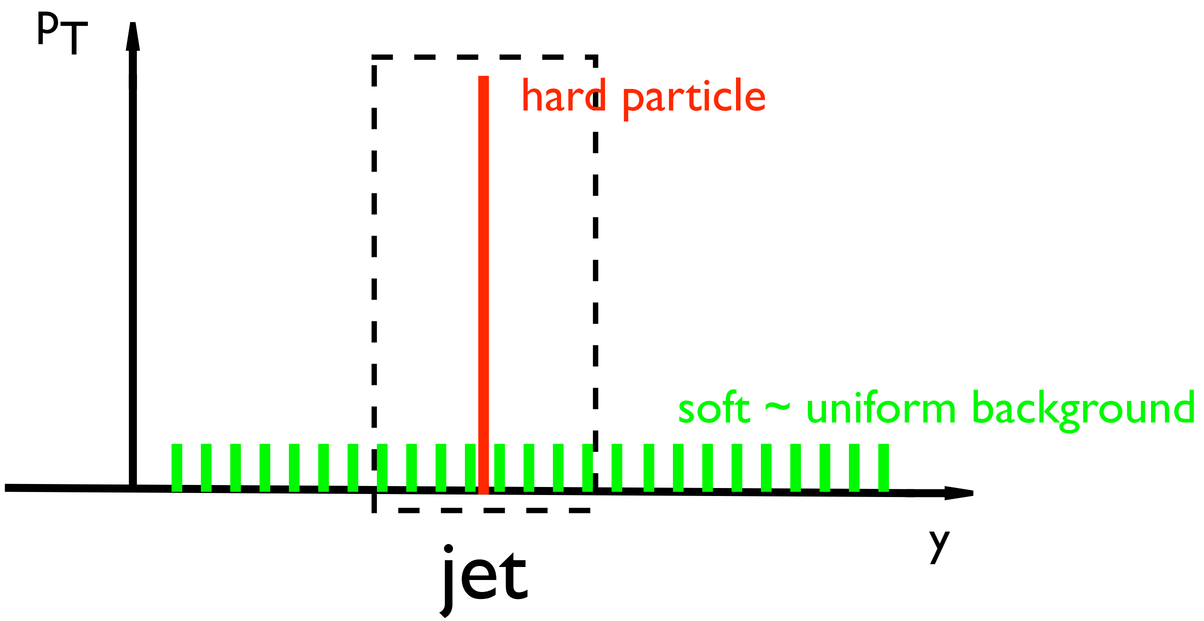
In addition, insert a single 100 GeV hard particle



Does it work? A toy model

Consider a uniform distribution of soft particles,
e.g. 10000 in the rapidity range [-4,4], with p_T = 1 GeV

In addition, insert a single 100 GeV hard particle

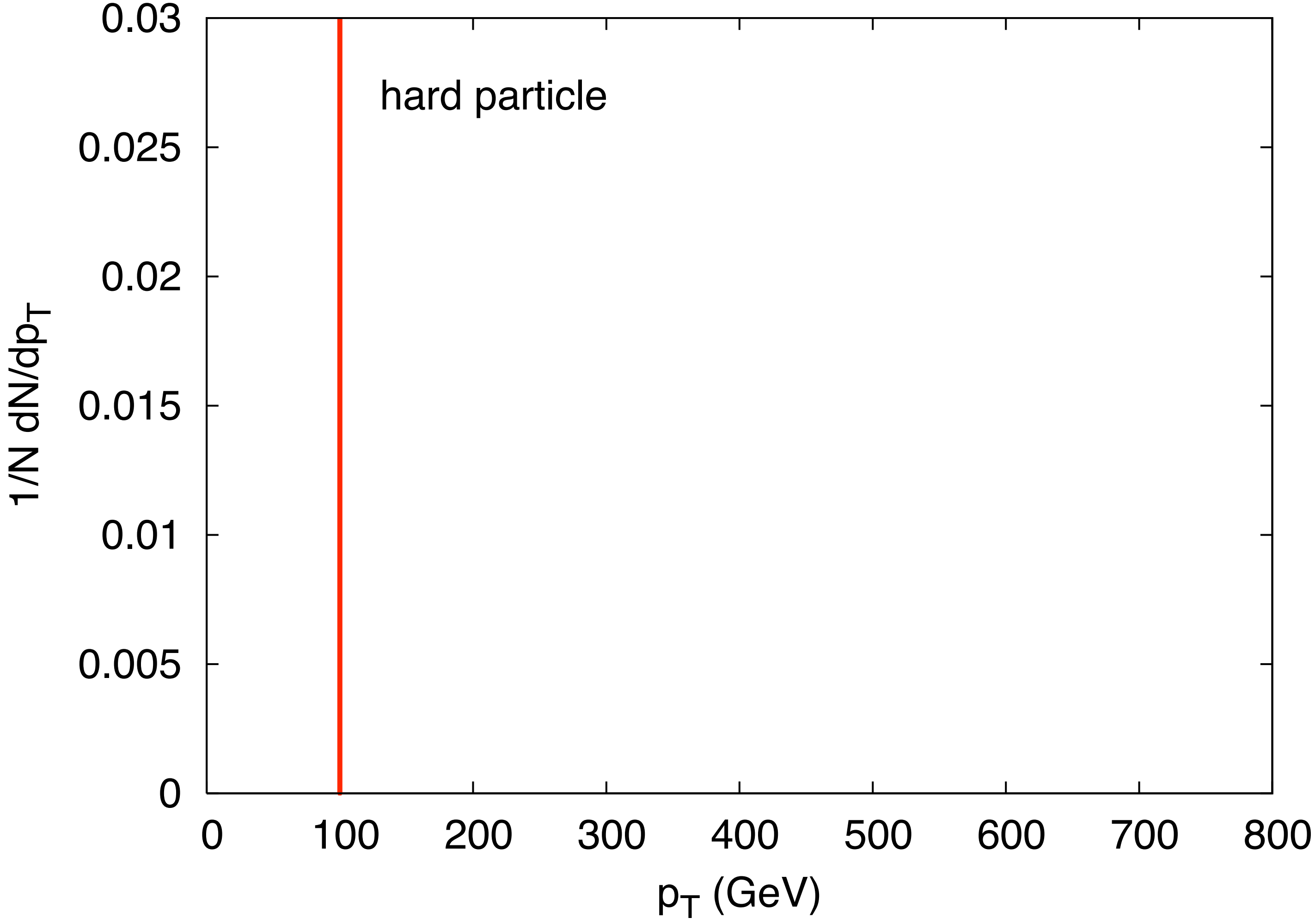


The jet will also contain soft particles, and have therefore a much larger p_T
Fluctuations in the background will degrade the resolution

Can we recover the momentum of the hard particle?

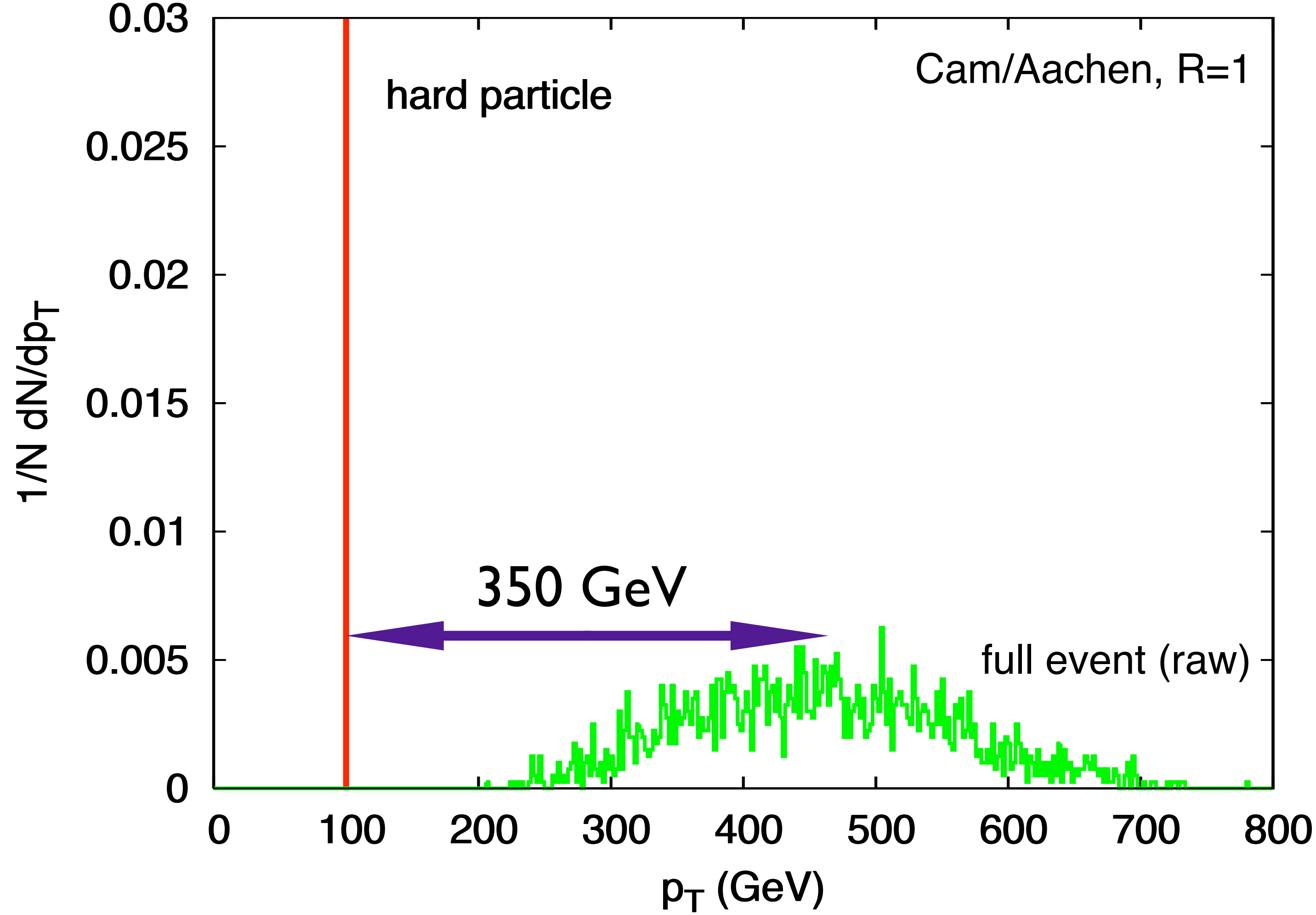
Does it work? A toy model

The hard particle



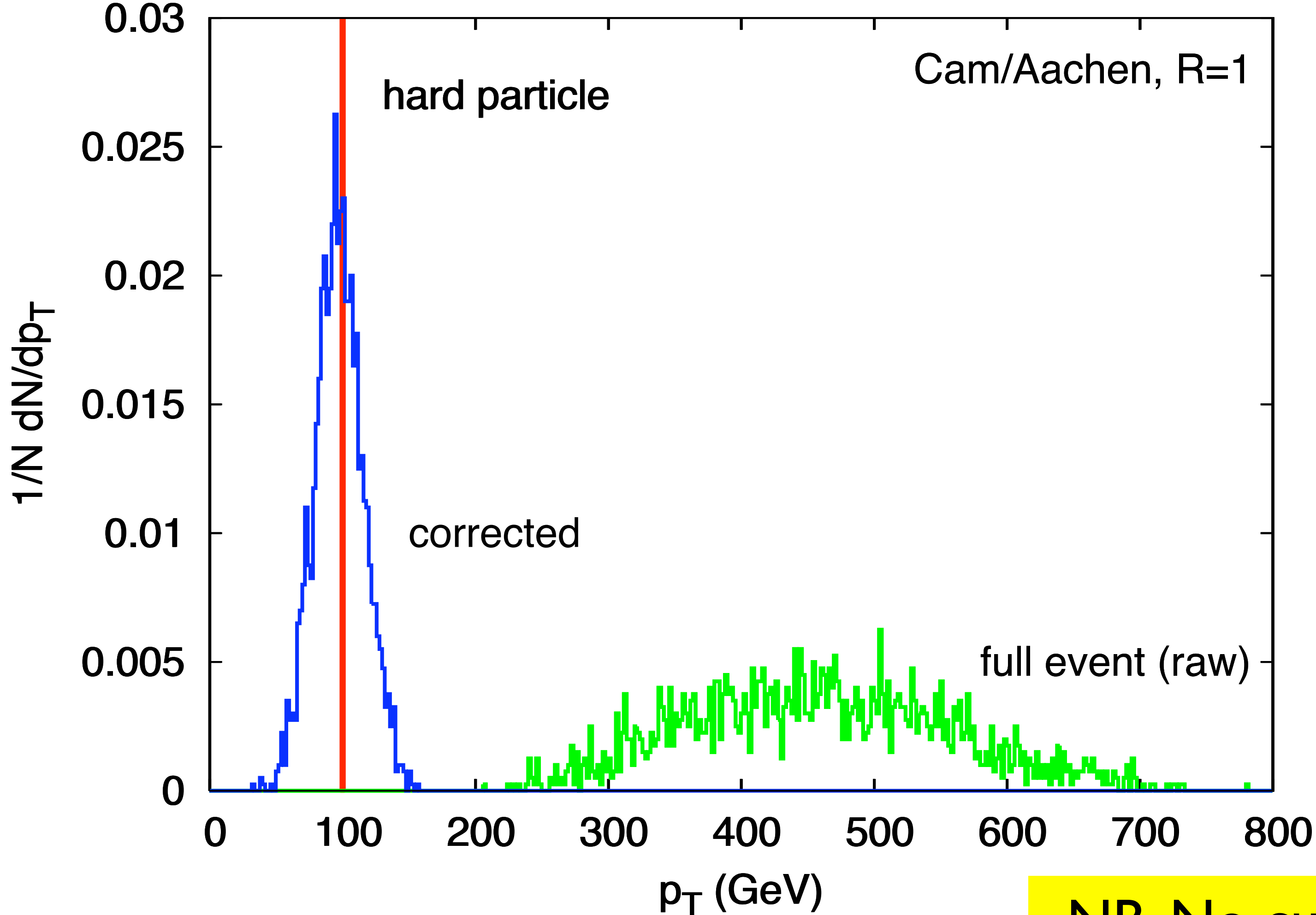
Does it work? A toy model

The hard particle clustered with the soft background



Does it work? A toy model

The hard particle clustered with the soft background, **after the subtraction**



The correct transverse momentum is recovered, with an important gain in resolution

NB. No cuts, no Monte Carlo correction: exclusively data driven

Roughly uniformly distributed background

In increasing order of number of particles/uniformity, we have, at the LHC,

- **Underlying event** in a single pp collision
  (about 200 particles)

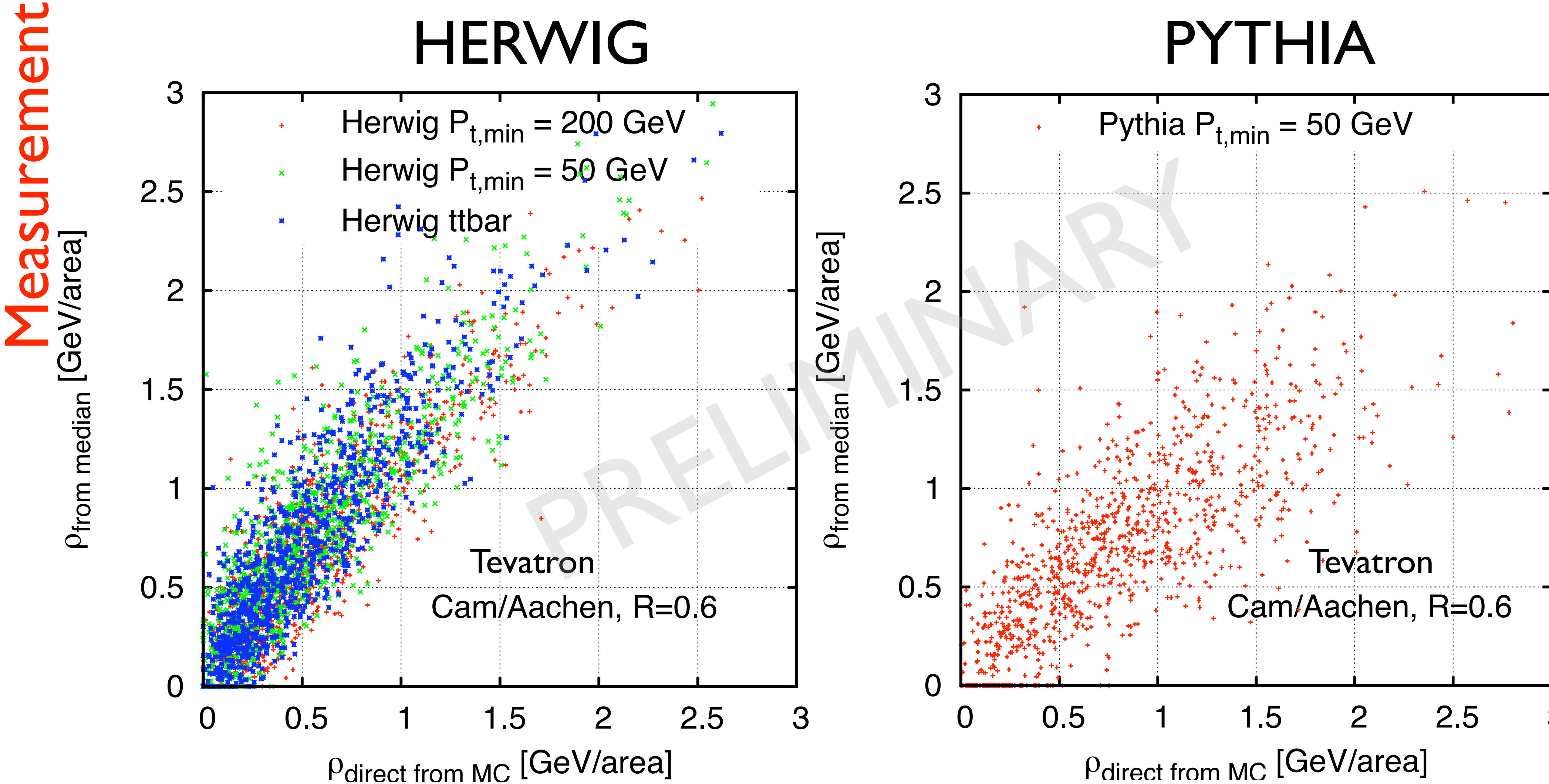
- **Pile-up** in high luminosity pp collisions
  (up to ~ 20 overlapping collisions, ⇒ ~ 4000 particles/event)

- **Background** in heavy ion collisions
  (~ 30000 particles / event)

Since the measurement of the background level relies on a uniform distribution of the 'background particles' themselves, and assumes the background to be uncorrelated with the hard jets, we must expect the underlying event case to be the most challenging one

Underlying Event estimation

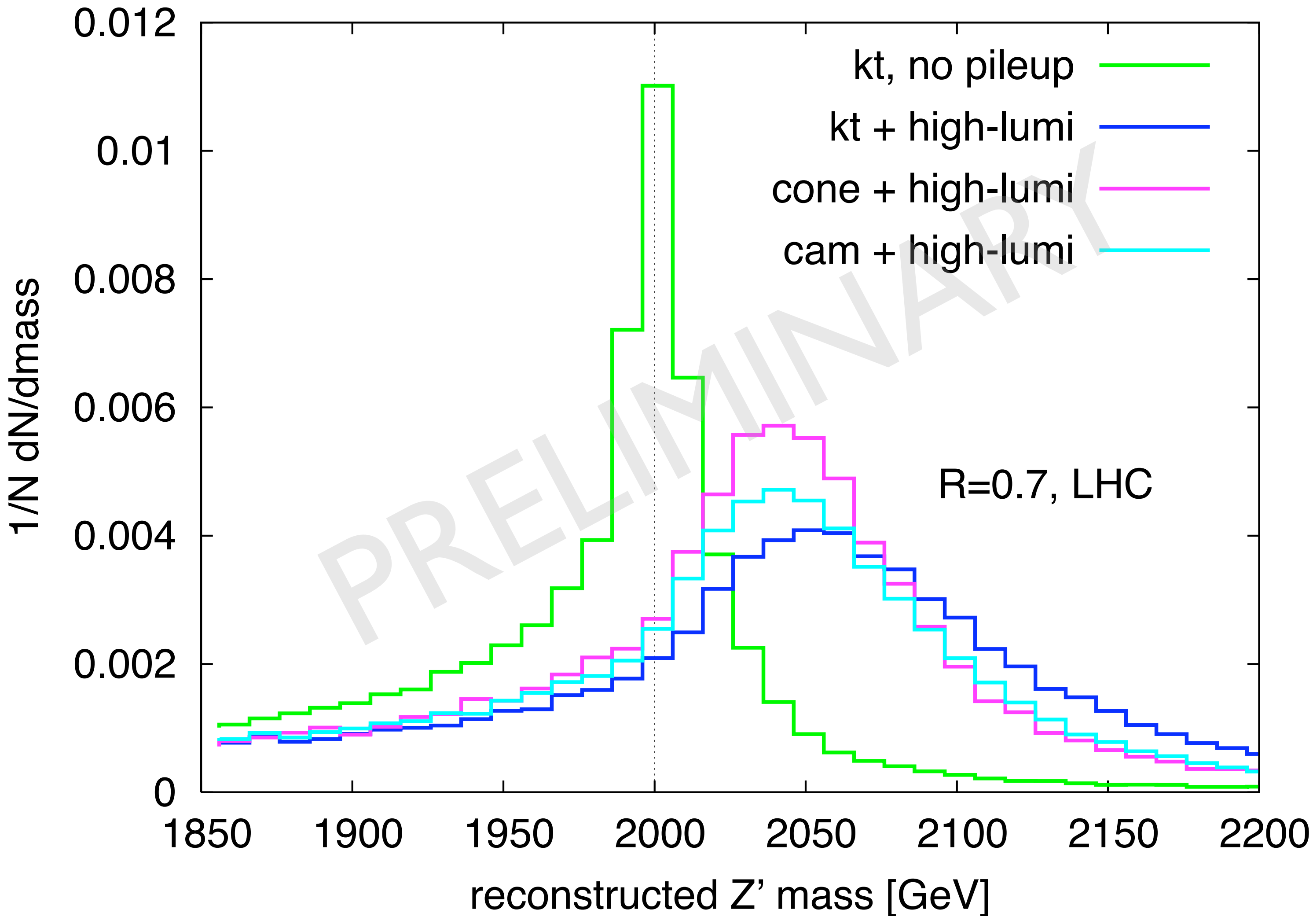
To test the procedure for the Underlying Event, compare the measurement of the background level made with areas with the known amount a Monte Carlo put in



Input from Monte Carlo

Pile-up at the LHC

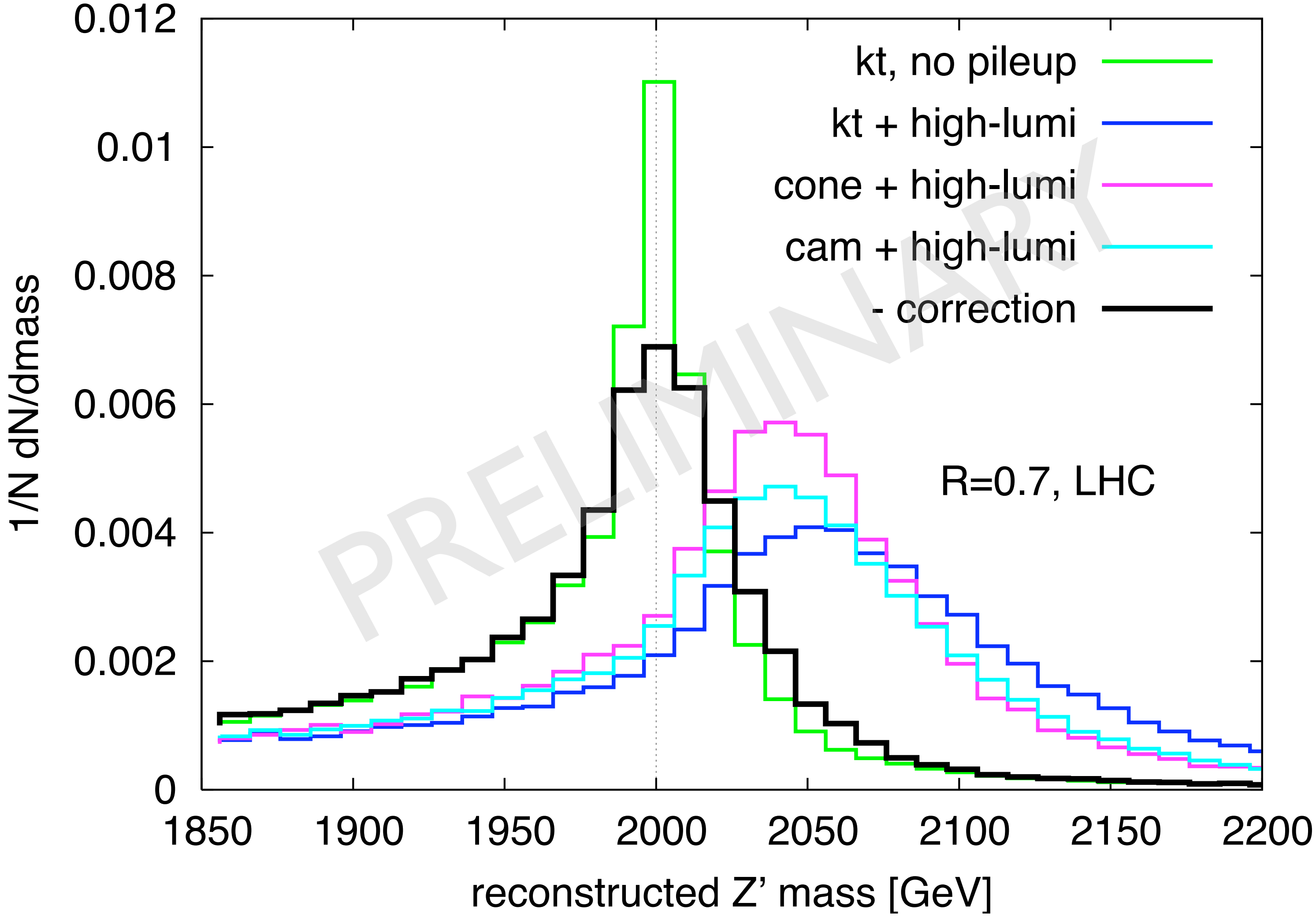
An hypothetical Z' invariant mass distribution



The peak is shifted and smeared when clustering together with the pile-up

Pile-up at the LHC

An hypothetical Z' invariant mass distribution

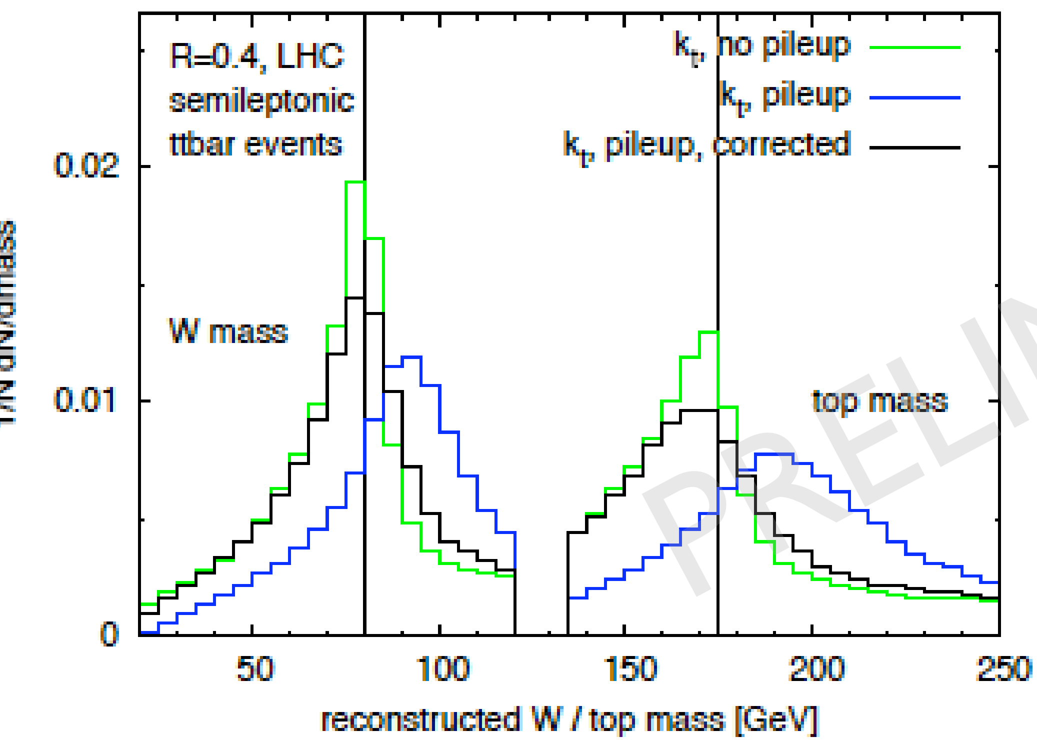


The correct mass is recovered, with good resolution, after subtraction

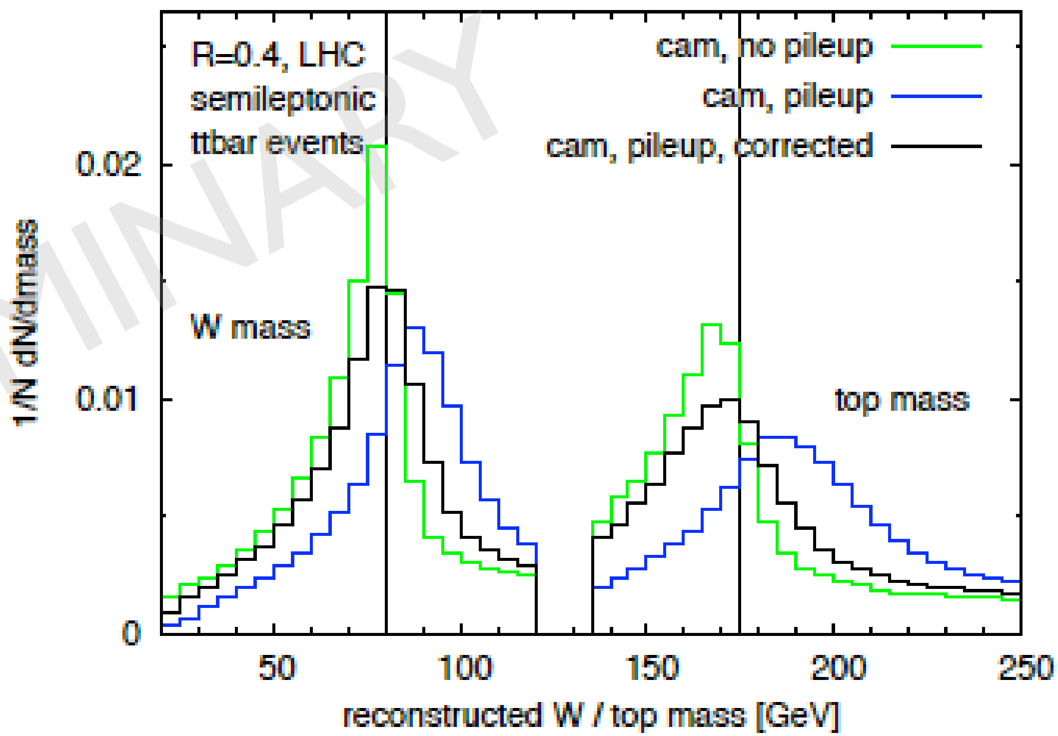
Pile-up at the LHC

Top production

k_t



Cambridge

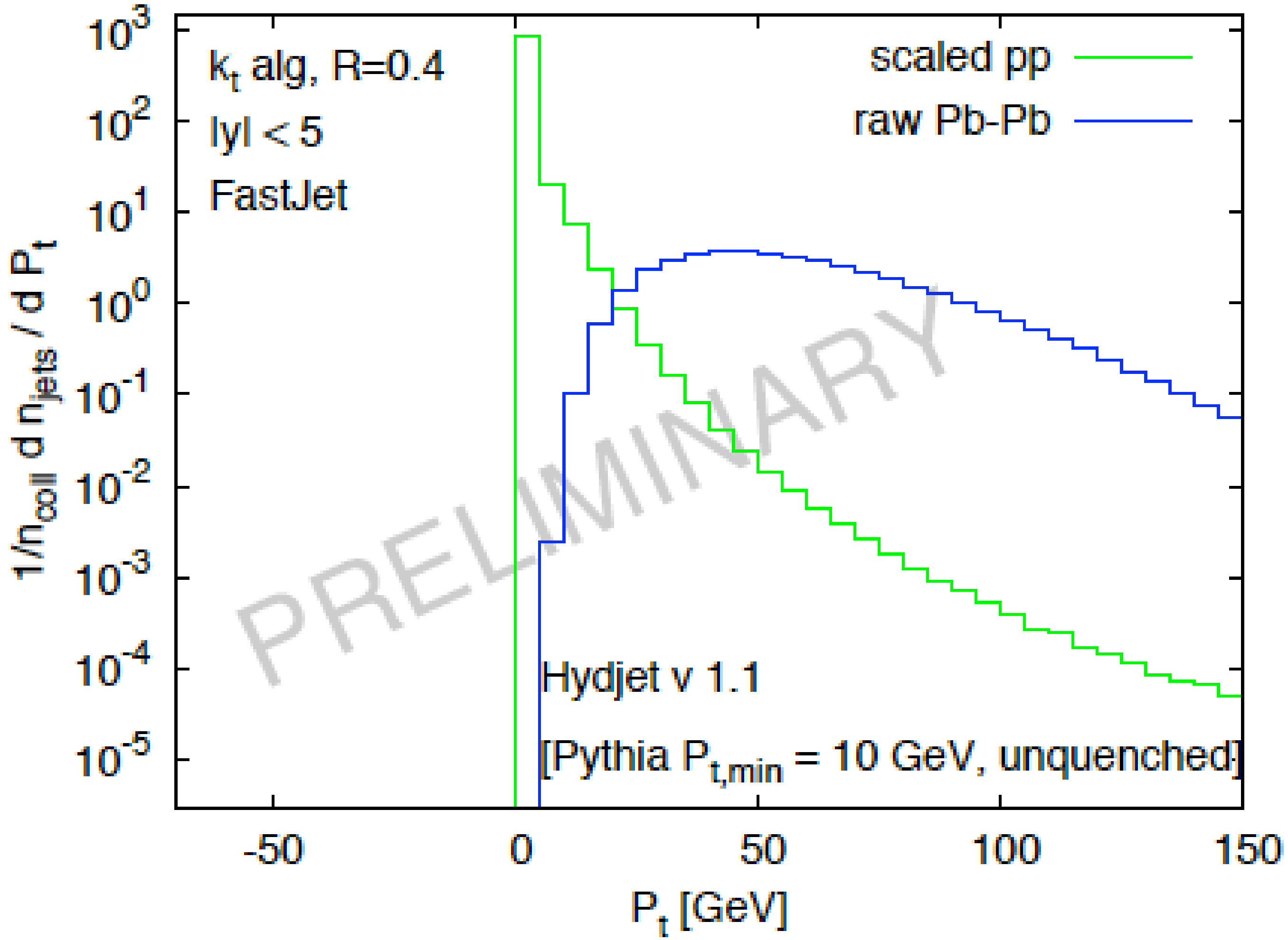


The top and W mass distributions get shifted, but they can be recovered after correction with good resolution

Inclusive jet distribution in HIC

The momentum density of simulated events is measured to be ~ 250 GeV per unit area

Hence, with R = 0.4 a jet on average gets ~ 125 GeV of additional transverse momentum

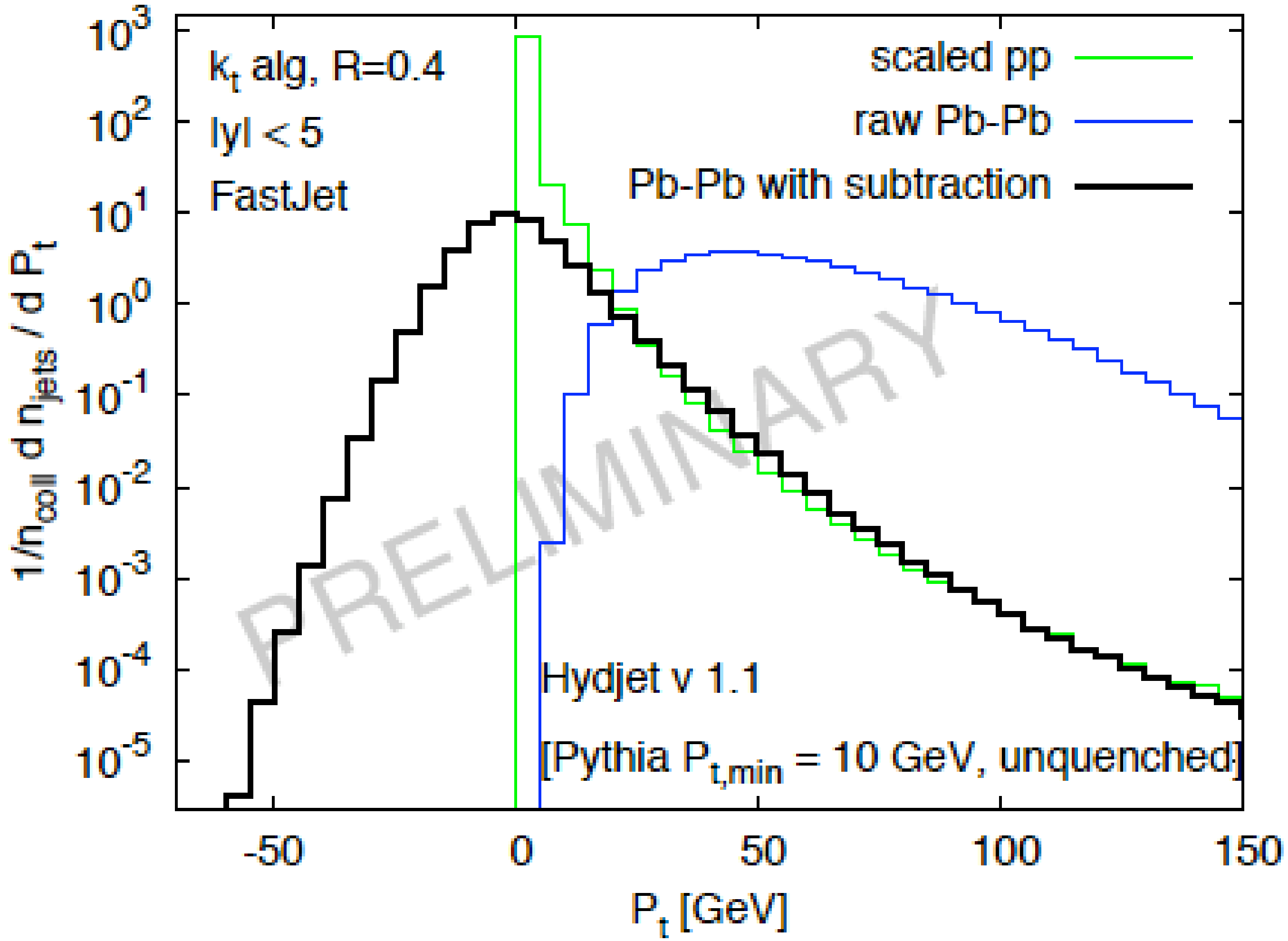


The jet distribution is completely distorted by the huge background......

Inclusive jet distribution in HIC

The momentum density of simulated events is measured to be ~ 250 GeV per unit area

Hence, with R = 0.4 a jet on average gets ~ 125 GeV of additional transverse momentum



The jet distribution is completely distorted by the huge background......

...but it can be recovered down to fairly low p_T

Conclusions

- ☑ Given a proper jet-finder, jet areas can be **defined**

Conclusions

- Given a proper jet-finder, jet areas can be **defined**
- They can be used to **estimate** the level of a uniformly distributed noise

Conclusions

- Given a proper jet-finder, jet areas can be **defined**
- They can be used to **estimate** the level of a uniformly distributed noise
- They can be used to **subtract** the background contribution from the hard jets.
  Everything is data driven: no cuts, no Monte Carlo corrections

Conclusions

- Given a proper jet-finder, jet areas can be **defined**

- They can be used to **estimate** the level of a uniformly distributed noise

- They can be used to **subtract** the background contribution from the hard jets.
  Everything is data driven: no cuts, no Monte Carlo corrections

- Preliminary Monte Carlo tests look promising.
  Full ‘experimental’ tests are now needed

Work in progress
To be published soon