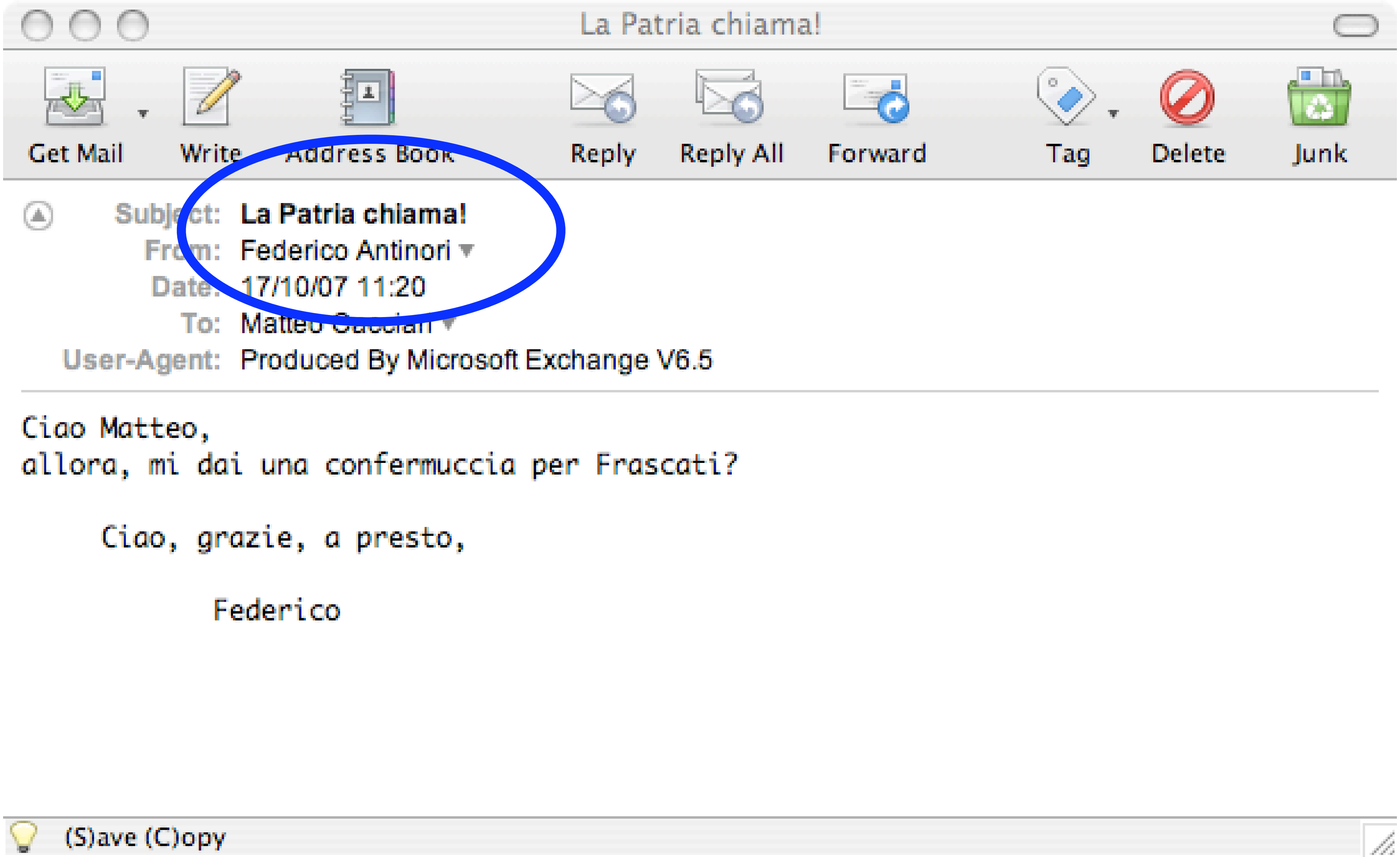
ALICE Italia
Frascati, 13 Novembre 2007

Jets, and Reconstruction with FastJet

Matteo Cacciari

LPTHE - Paris 6,7 and CNRS

In collaboration with
Gavin Salam and Gregory Soyez


La Patria chiama!
Get Mail
Write
Address Book
Reply
Reply All
Forward
Tag
Delete
Junk
Subject: La Patria chiama!
From: Federico Antinori
Date: 17/10/07 11:20
To: Matteo
User-Agent: Produced By Microsoft Exchange V6.5
Ciao Matteo,
allora, mi dai una confermuccia per Frascati?
Ciao, grazie, a presto,
Federico
(S)ave (C)opy

Goals

Consider **real** jet algorithms

i.e. jets that can be predicted from perturbative QCD

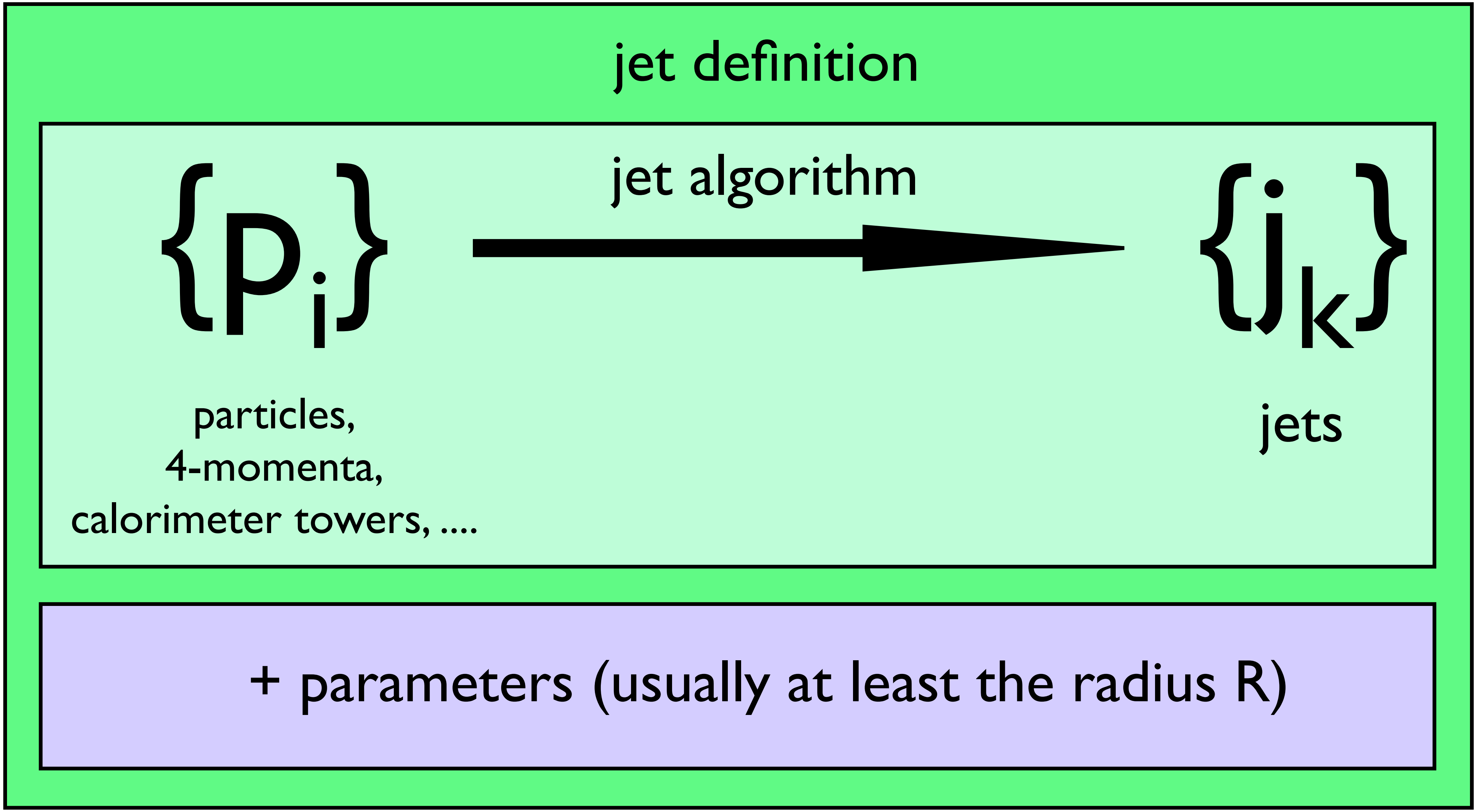
What tools are available?

Subtract diffuse soft background from hard jets

UE/pileup/heavy ions background

How do we do it?

Jet Definition



Reminder: running a jet definition gives a well defined physical observable, which we can measure and, hopefully, calculate

Jet Algorithm requirements

A jet algorithm **must** be

✔ infrared and collinear safe

soft emission shouldn't change jets
collinear splitting shouldn't change jets

✔ identically defined at parton and hadron level

so that perturbative calculations can be compared to experiments

It is **nice** if a jet algorithm is

✔ not too sensitive to hadronisation, underlying event, pile-up

(because we are not very good at modeling non-perturbative stuff)

✔ realistically applicable at detector level

(e.g. not too slow)

Jet Algorithms

Two main jet algorithm classes:
cone algorithms and **sequential clustering algorithms**

- **Cone-type** algorithms (JetClu, ILCA/MidPoint, ...) are mainly used at the Tevatron. They **identify energy flow into cones**. Detailed definition can be messy. Infrared/collinear safety must be carefully studied.

- **Sequential clustering** algorithms (kt, Cambridge/Aachen, Jade,...) are based on **pair-wise successive recombinations**. Widely used at LEP and HERA. Simple definition, safely infrared and collinear safe.

The first cone

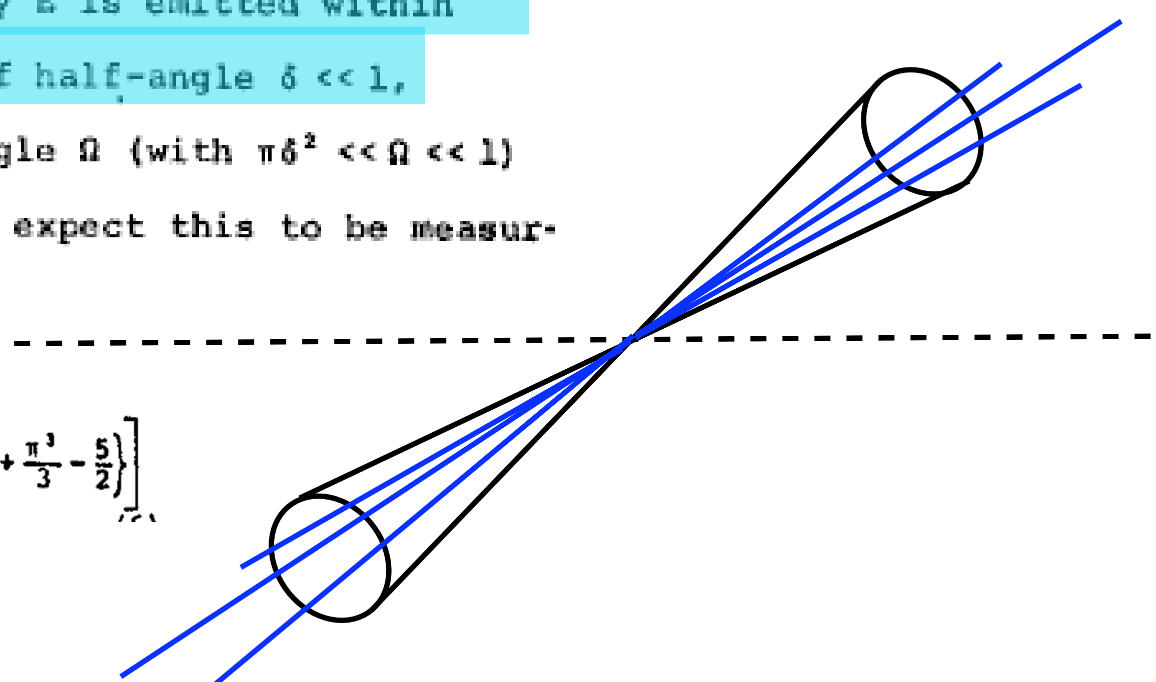
Jets are as old as the parton model (yes, even older than QCD...):

S.D. Drell, D.J. Levy and T.M. Yan, Phys. Rev. **187**, 2159 (1969) and **D1**, 1617 (1970)
N. Cabibbo, G. Parisi and M. Testa, Lett. Nuovo Cimento **4**, 35 (1970)
J.D. Bjorken and S. D. Brodsky, Phys. Rev. **D1**, 1416 (1970)
R.P. Feynman, Photon Hadron Interactions, p. 166 (1972)

The first rigorous definition of an **infrared and collinear safe** jet in QCD is due to Sterman and Weinberg, Phys. Rev. Lett. **39**, 1436 (1977):

To study jets, we consider the partial cross section $\sigma(E,\theta,\Omega,\epsilon,\delta)$ for e^+e^- hadron production events, in which all but a fraction $\epsilon \ll 1$ of the total e^+e^- energy E is emitted within some pair of oppositely directed cones of half-angle $\delta \ll 1$, lying within two fixed cones of solid angle Ω (with $\pi\delta^2 \ll \Omega \ll 1$) at an angle θ to the e^+e^- beam line. We expect this to be measur-

$$\sigma(E,\theta,\Omega,\epsilon,\delta) = (d\sigma/d\Omega)_0\,\Omega\left[1 - (g_E^2/3\pi^2)\left\{3\ln\delta + 4\ln\delta\,\ln 2\epsilon + \frac{\pi^3}{3} - \frac{5}{2}\right\}\right]$$



A modern cone algorithm

How do I decide where to place the cones?

- ☞ try an initial location
- ☞ sum 4-momenta of particles, find axis of cone
- ☞ use axis as a new trial location, and **iterate**
- ☞ stop when axis is stable
- ☞ merge overlapping cones, or split them into two

Issues:

☠ Where do I start?

Seedless (i.e. everywhere)?	Very slow
Some particles above a threshold?	Collinear unsafe
Calorimeter towers?	Expt. dependent

☠ How do I split/merge?

Complicated procedure, risky, not necessarily physical

k_t algorithm

The definition of a sequential clustering algorithm, on the other hand, is extremely simple.

For instance, take **the longitudinally invariant k_t:**

S. Catani, Y. Dokshitzer, M. Seymour and B. Webber, Nucl. Phys. B406 (1993) 187
S.D. Ellis and D.E. Soper, Phys. Rev. D48 (1993) 3160

- Calculate the distances between the particles: $d_{ij} = \min(k_{ti}^2, k_{tj}^2) \frac{\Delta y^2 + \Delta\phi^2}{R^2}$
- Calculate the beam distances: $d_{iB} = k_{ti}^2$
- Combine particles with smallest distance or, if d_{iB} is smallest, call it a jet
- Find again smallest distance and repeat procedure until no particles are left

[In most naive implementation, calculating all distances is an N^2 operation, to be repeated N times ⇒ $O(N^3)$]

Tools

Until two years ago:

- Cone algorithms: not really safe

  Typical cone algorithms (JetClu, MidPoint, etc) are not infrared safe: at some order in perturbation theory they will fail

- k_t algorithm: very slow for large N ($\sim N^3$)

  Clustering many particles takes a very long time (~ 1 day CPU time for one LHC heavy ion event)

Tools

<u>Now</u>:

- k_t and Cam/Aachen algorithms: very fast ($\sim N \ln N$)

MC, G. Salam, hep-ph/0512210

- Cone safe and reasonably fast (SISCone, $\sim N^2 \ln N$)

G. Salam, G. Soyez, arXiv:0704.0292

- Subtraction of background using jet areas

MC, G. Salam, arXiv:0707.1378

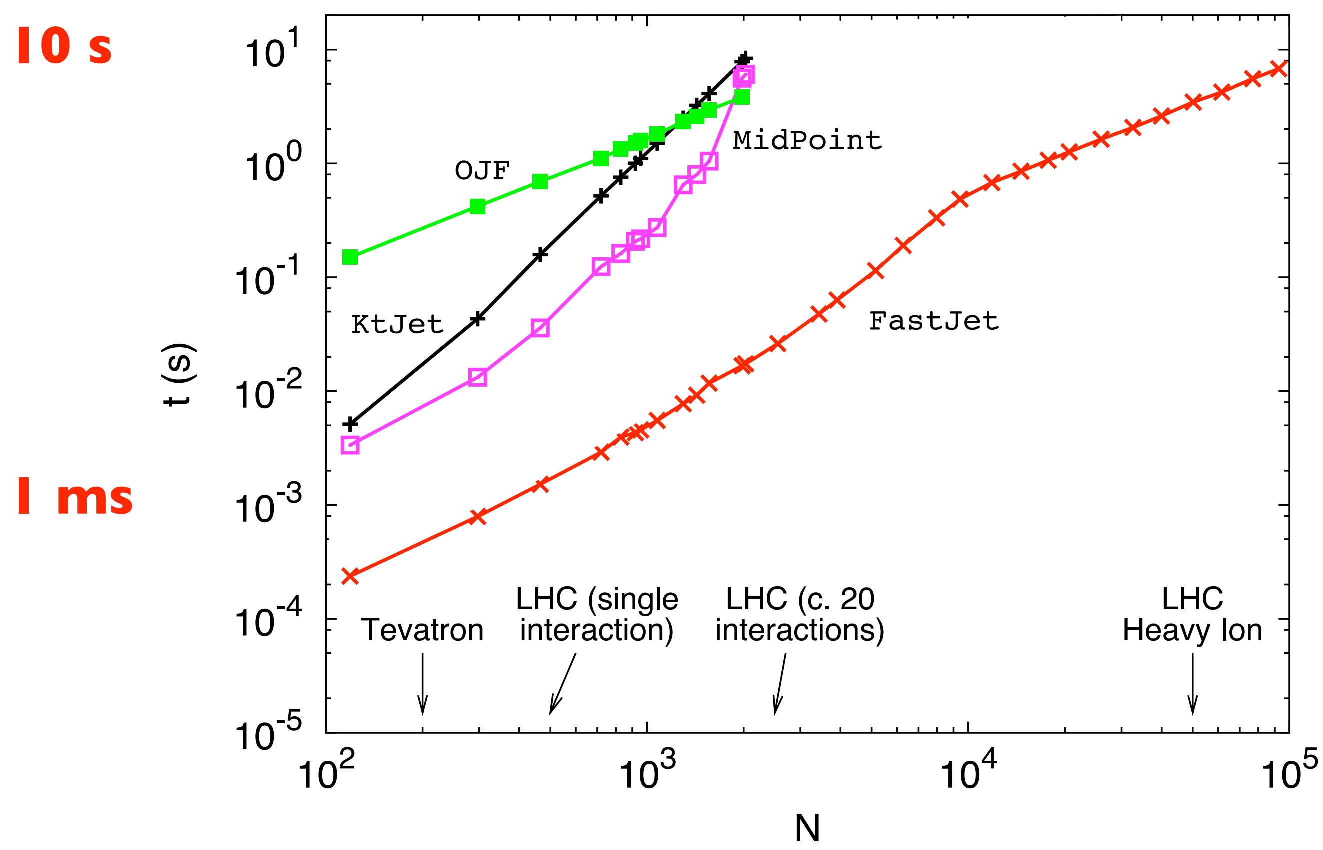
- anti-k_t algorithm (recombination algorithm, but gives perfect cones)

MC, G. Salam, G. Soyez, in preparation

FastJet: http://www.lpthe.jussieu.fr/~salam/fastjet

FastJet performance

Time taken to cluster N particles (k_t algorithm):



- Almost two orders of magnitude gain at small N (related $O(N^2)$ implementation)
- Large-N region now reachable (O(1 sec) rather than 1 day for heavy ion collisions)

What is anti-k_t?

A new recombination-type algorithm,
using a distance measure similar to k_t and Cambridge/Aachen:

$$d_{ij} = \min(k_{ti}^2, k_{tj}^2) \frac{\Delta y^2 + \Delta\phi^2}{R^2}$$

k_t

$$d_{ij} = \frac{\Delta y^2 + \Delta\phi^2}{R^2}$$

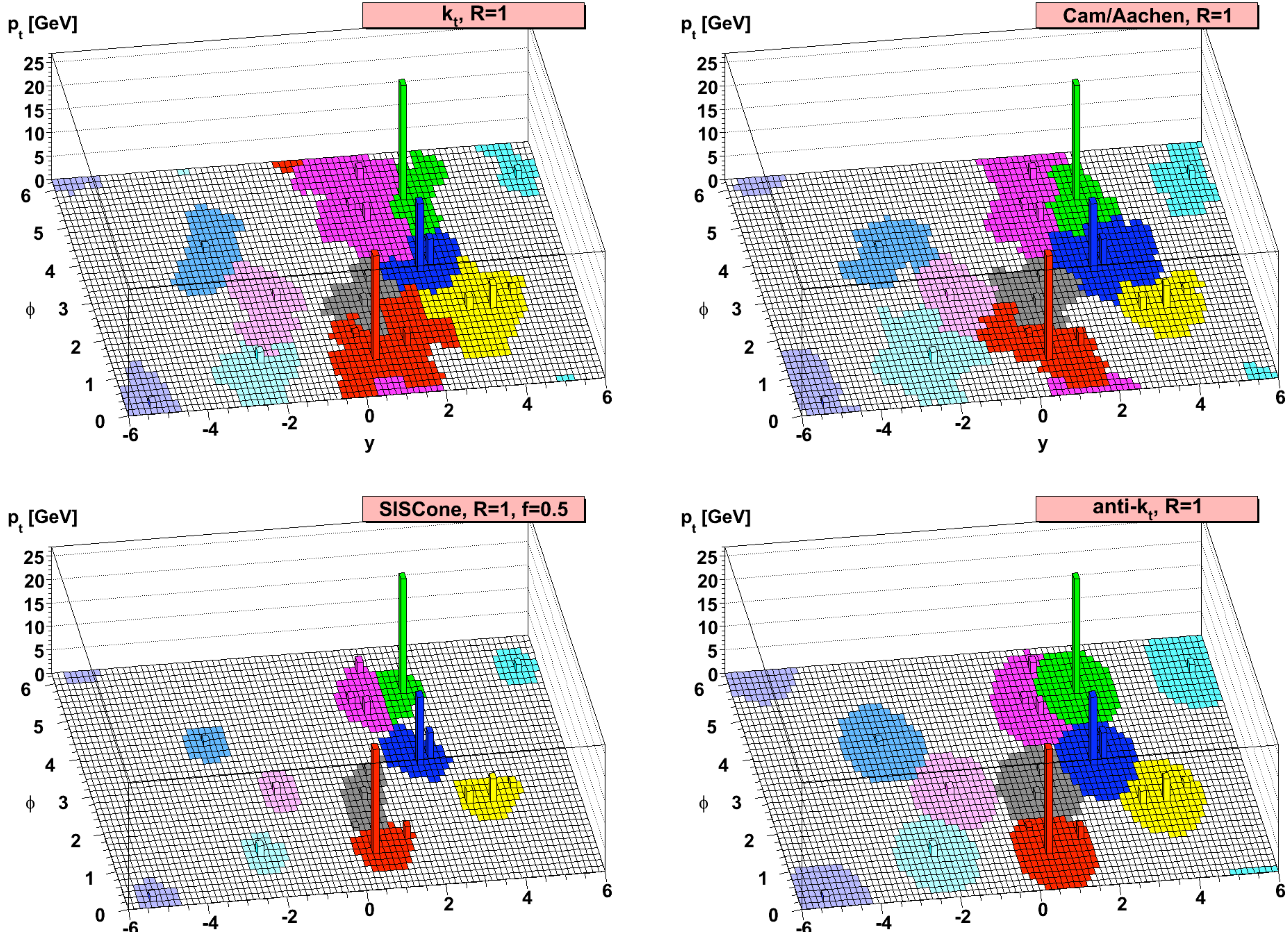
Cambridge/Aachen

$$d_{ij} = \min\left(\frac{1}{k_{ti}^2}, \frac{1}{k_{tj}^2}\right) \frac{\Delta y^2 + \Delta\phi^2}{R^2}$$

anti-k_t

Three members of a family of algorithms:

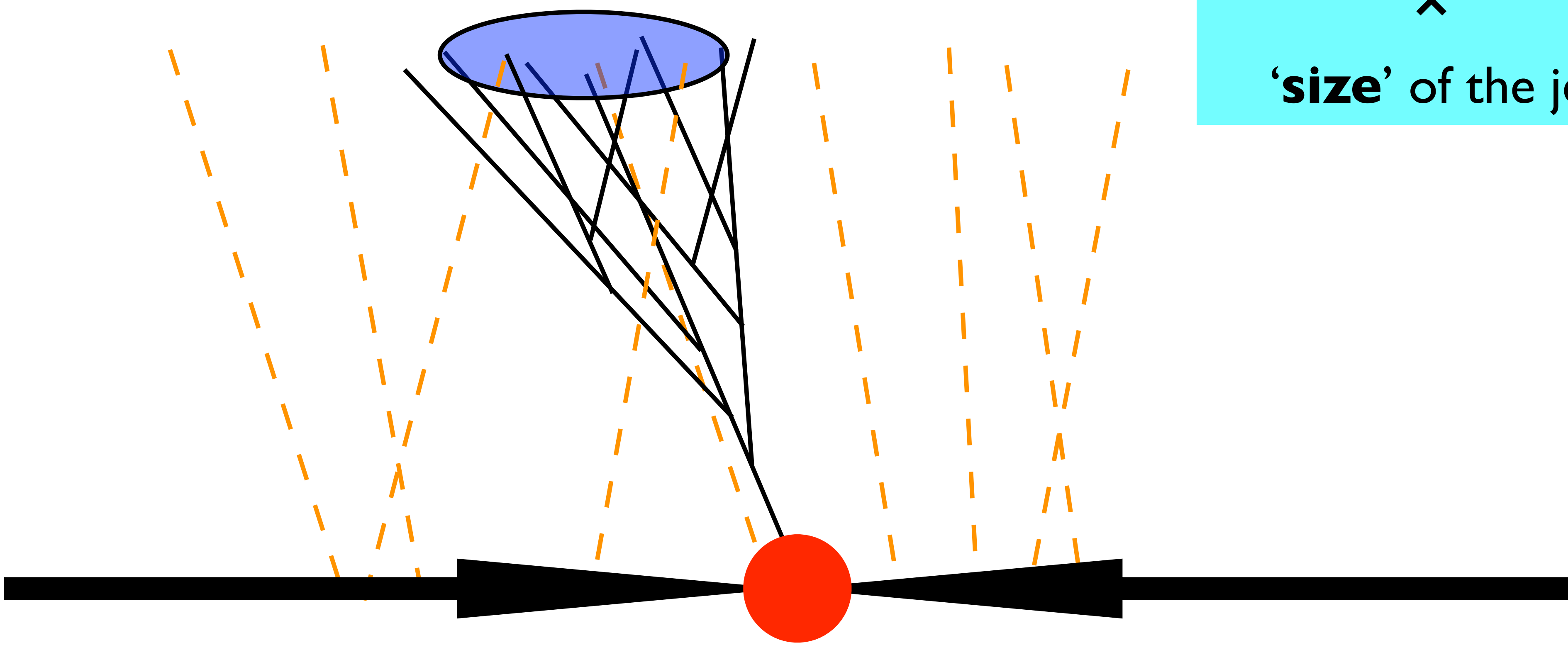
$$d_{ij} = \min(k_{ti}^{2p}, k_{tj}^{2p}) \frac{\Delta y^2 + \Delta\phi^2}{R^2} \quad \begin{cases} p=1 & k_t \\ p=0 & \text{Cam/Aachen} \\ p=-1 & \text{anti-}k_t \end{cases}$$



k_t, R=1
Cam/Aachen, R=1
SISCone, R=1, f=0.5
anti-k_t, R=1
p_t [GeV]
ϕ
y

A crowded event

p_T (jet) ~ p_T (parton) + Average underlying momentum density × '**size**' of the jet



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

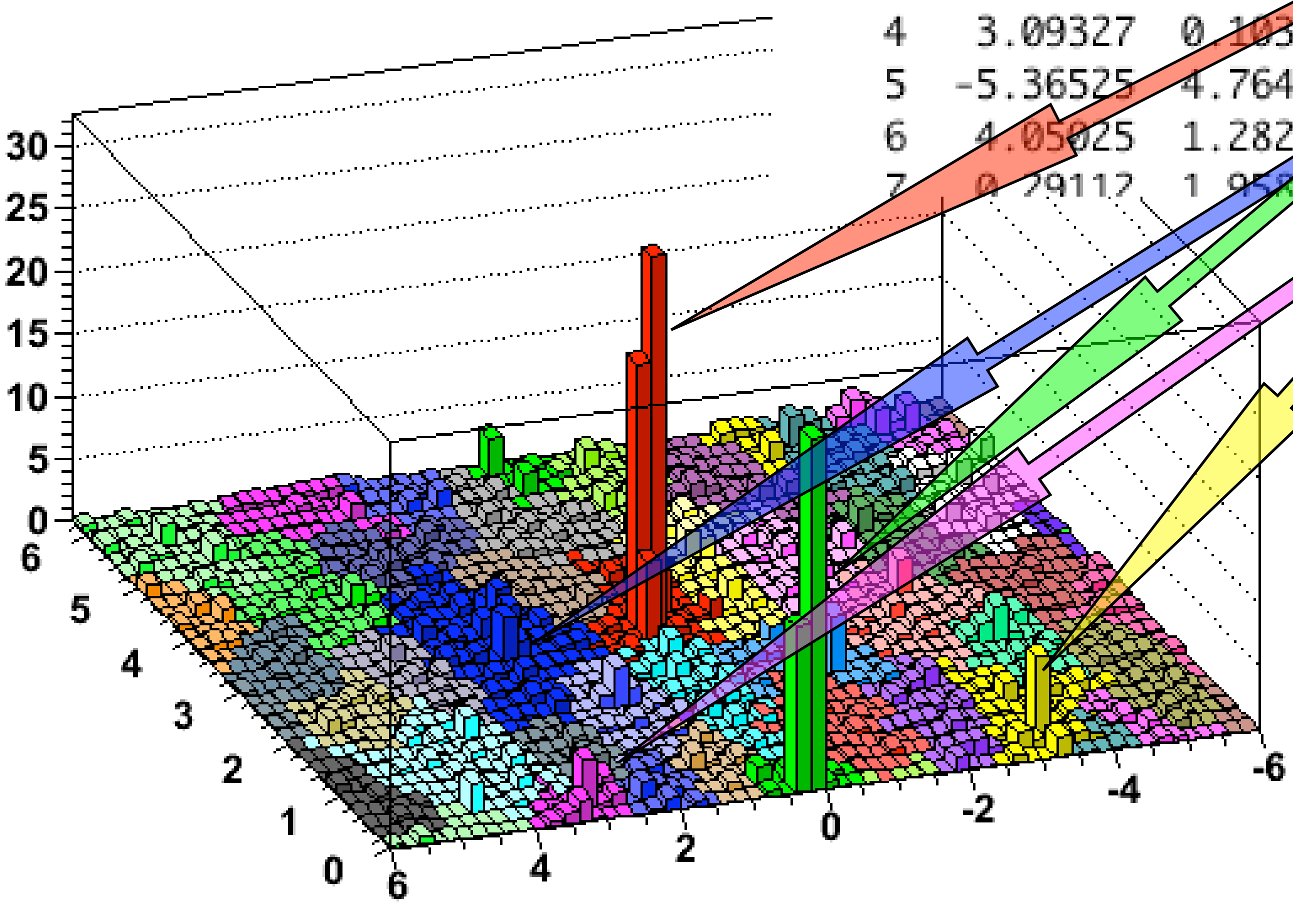
Can we subtract it from the jet to find the parton momentum?

What is the '**size**' of a jet??

The Active Jet Area

FastJet allows the calculation of the **areas** of the jets

```
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     19.598       2.780 +-  0.012
    6   4.05025   1.28279     15.561       3.592 +-  0.028
    7   0.79112   1.95845     14.566       2.114 +-  0.018
```



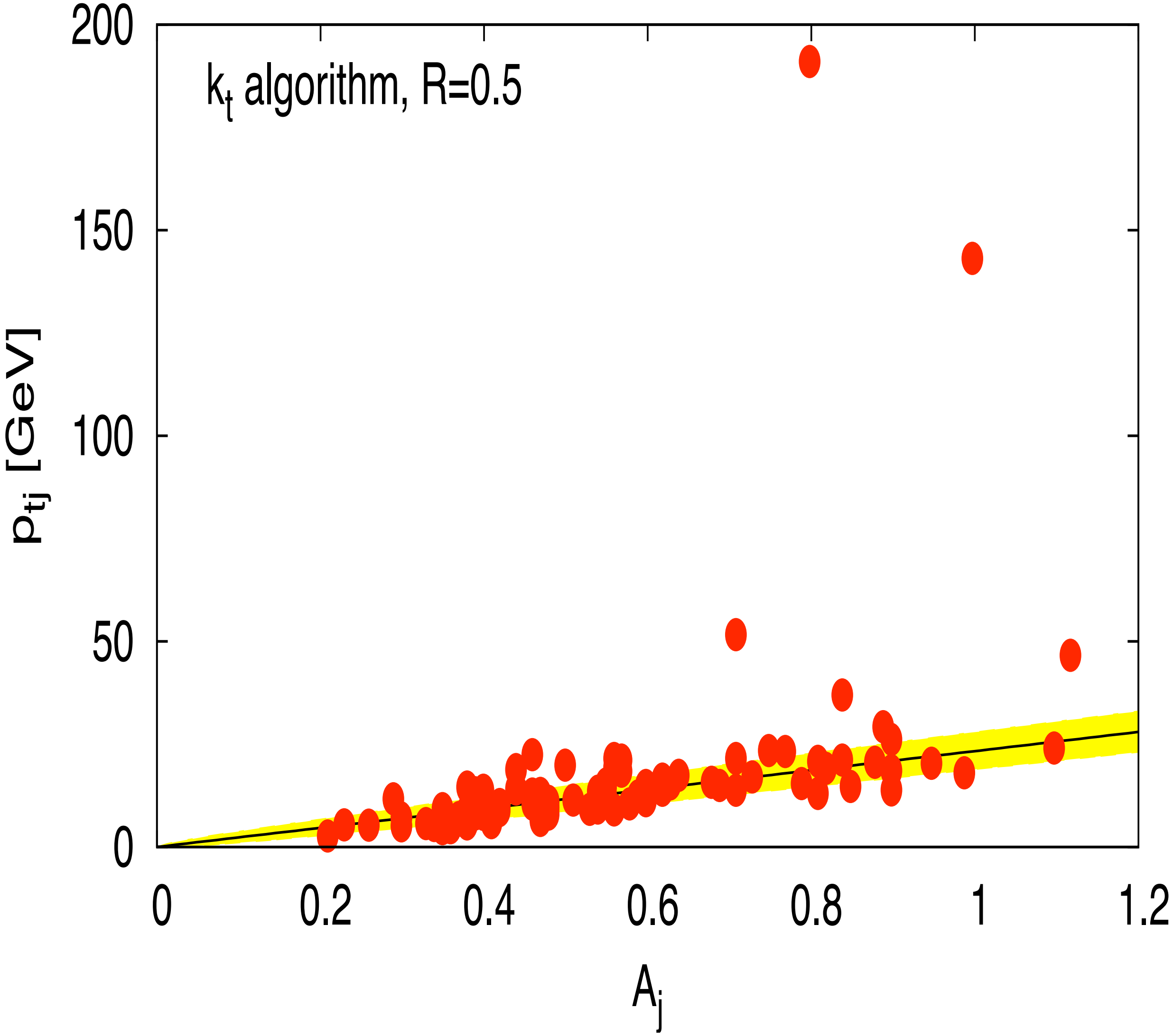
Try to estimate the **active area** of each jet

Fill event with many very soft particles, count how many are clustered into given jet

[NB. This is a **definition**]

Area vs. p_T

LHC: dijet event + high-lumi pilup

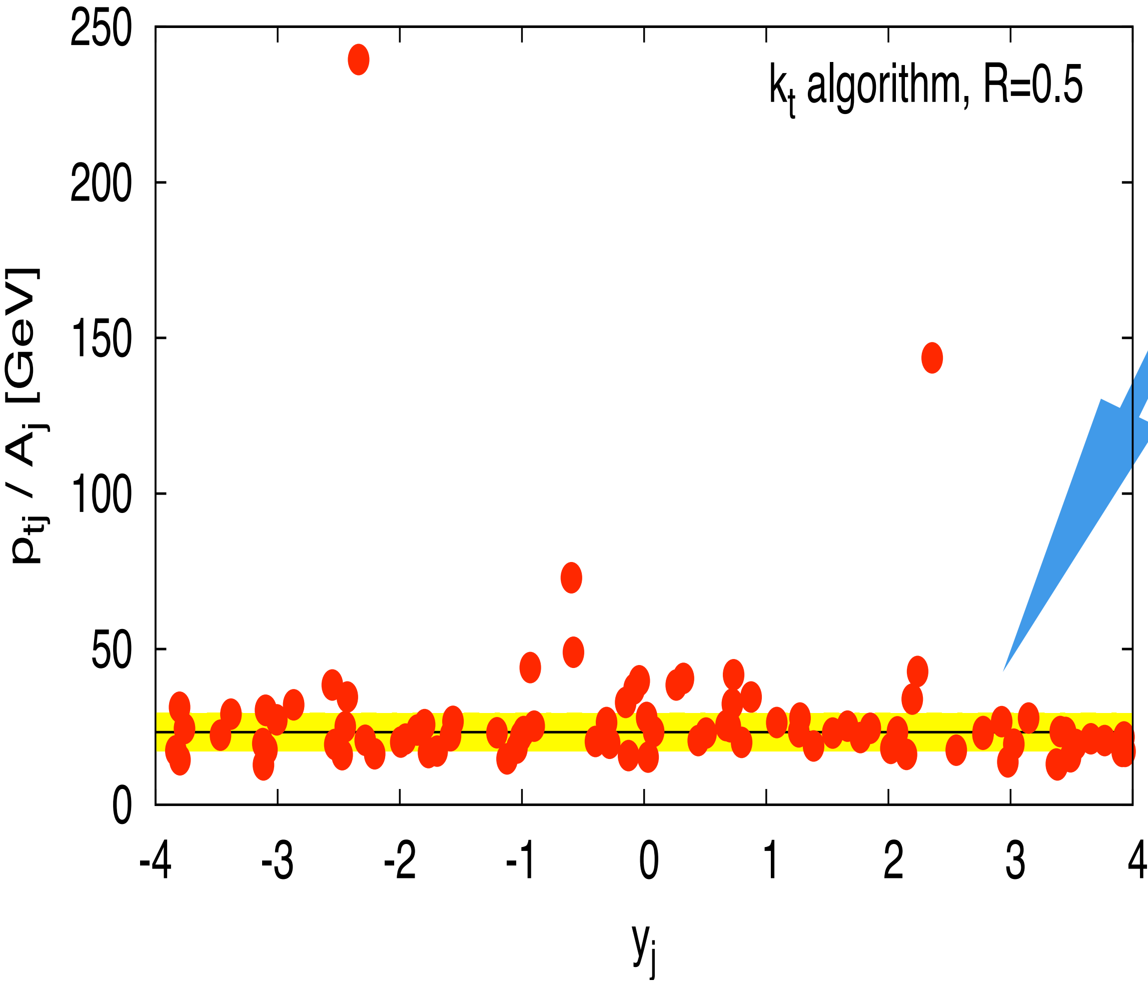


The jets adapt to the surrounding environment

They can have very different areas

Area vs. p_T

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



The distribution of background jets establishes its own average momentum density

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

<u>Dynamical selection</u>

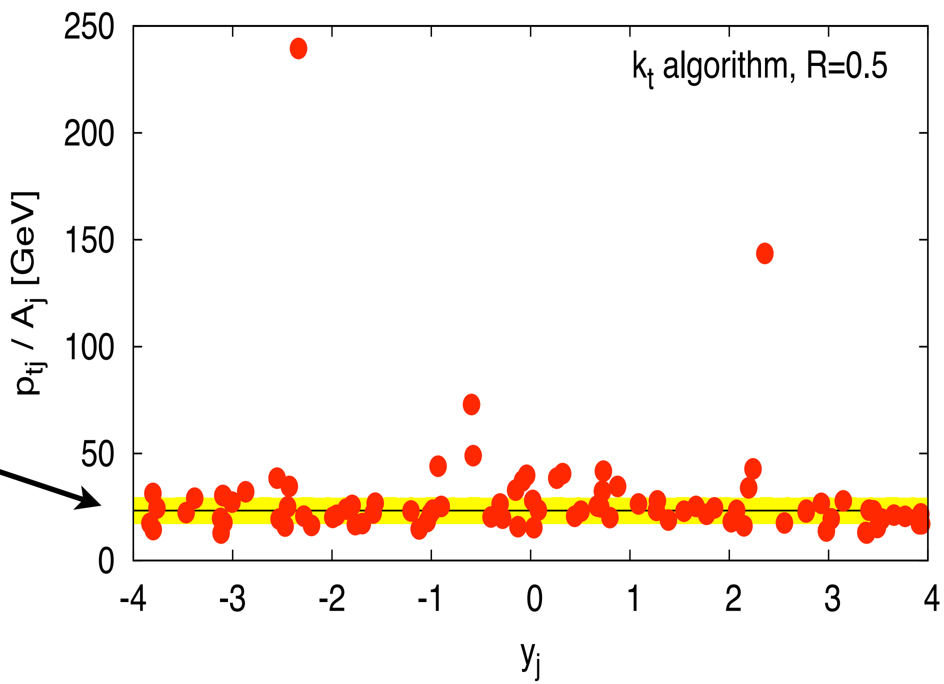
The jets are classified as belonging to the noise on the basis of their **characteristics**

Extraction of average noise momentum density

$$\rho \equiv \text{median}\left[\left\{\frac{p_t^{jet}}{\text{Area}_{jet}}\right\}\right]$$

(Taking the median of the distribution is a nice trick to get rid of the possible bias from the few hard jets)

One can also estimate the fluctuations (yellow band)



Subtraction

- A proper operative definition of **jet area** can be given
- When a hard event is superimposed on a **roughly uniformly distributed background**, study of **transverse momentum/area** of each jet allows one to 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:

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

NB. Procedure fully data driven.
No Monte Carlo corrections
needed in principle

The subtraction: ease of implementation

```
// the input particles' 4-momenta
vector<fastjet::PseudoJet> input_particles;

// choose the jet algorithm
fastjet::JetDefinition jet_def(kt_algorithm,R);

// define the kind of area
fastjet::GhostedAreaSpec ghosted_area_spec(ghost_etamax);
fastjet::AreaDefinition area_def(ghosted_area_spec);

// perform the clustering
fastjet::ClusterSequence cs(input_particles,jet_def,area_def);

// get the jets with pt > 0
vector<fastjet::PseudoJet> jets = cs.inclusive_jets();

// a jet transverse momentum, area, and area 4-vector
double pt = jets[0].perp();
double area = cs.area(jets[0]);
fastjet::Pseudojet area_4vector = cs.area_4vector(jets[0]);
```

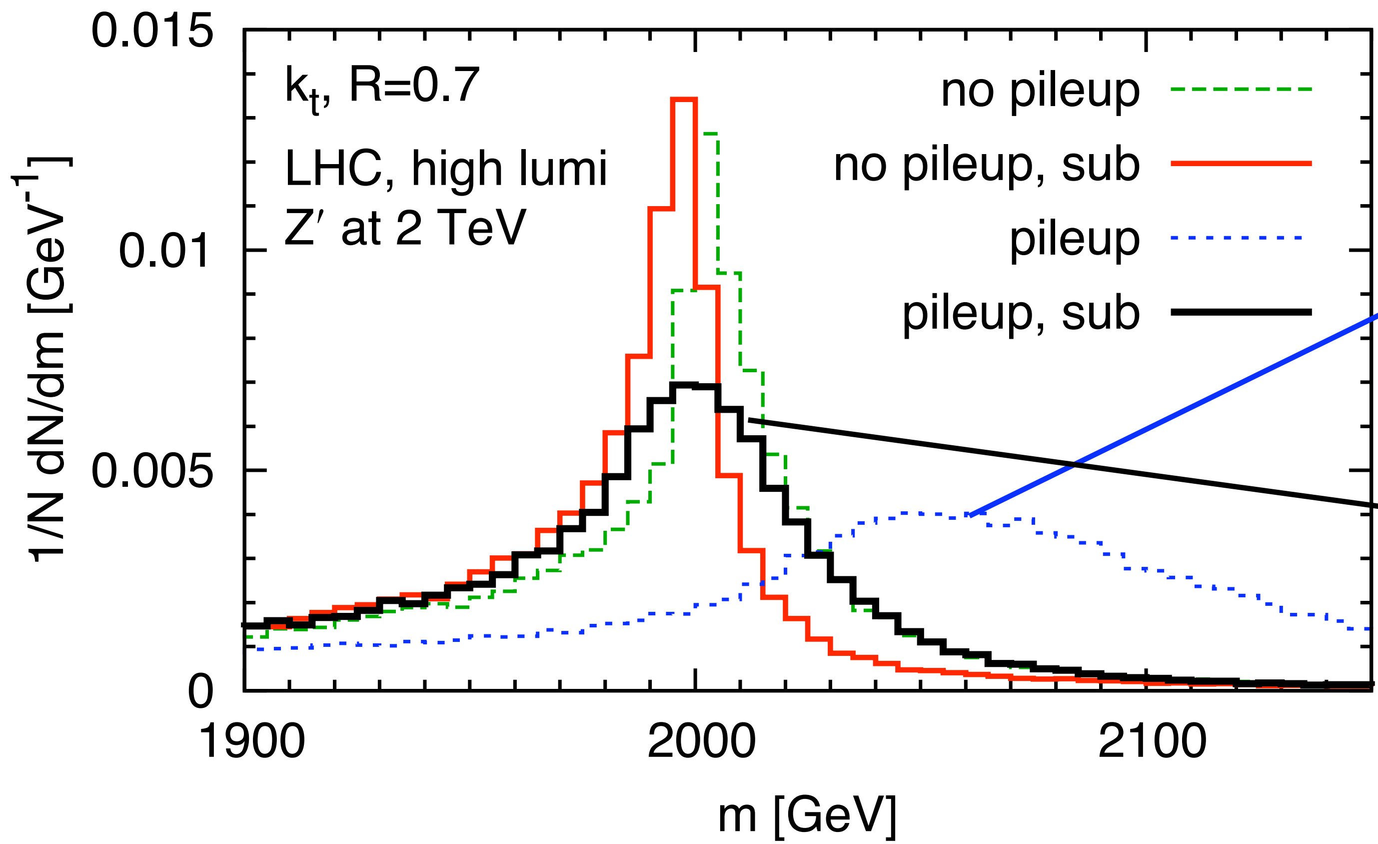
```
// get the median, i.e. rho
double rho = cs.median_pt_per_unit_area(rapmax); // or:
double rho_4v = cs.median_pt_per_unit_area_4vector(rapmax);

// subtract
double pt_sub = pt - rho * area;  // or:
fastjet::Pseudojet p_sub = jets[0] - rho_4v * area_4vector;
```

NB1. The "_4vector" variants also correct jet directions, and are better for large R

NB2. This is a pp case, but heavy ions is similar

Reconstructed Z' mass



Pileup shifts peak by ~ 50 GeV, and broadens it

Correct peak position and better resolution after subtraction

	k_t		Cam/Aachen		SISCone	
	m	Δm	m	Δm	m	Δm
no pileup	2003	10	2002	10	1998	10
no pileup, sub	1995	13	1995	8	1993	10
pileup	2065	60	2049	48	2030	33
pileup, sub	1998	25	1998	25	1997	20

Heavy Ion Collisions: PbPb @ LHC

Background much larger than LHC hi-lumi pileup:

HYDJET v1.1

$$\left.\frac{dN_{ch}}{dy}\right|_{y=0} = 1600 \quad \Longrightarrow \quad \rho_{background} \equiv \frac{dp_T}{dy\,d\phi} \sim 250 \text{ GeV}$$

Hence, a jet with R = 0.4 on average gets an additional

$$\Delta p_T \simeq \rho_{background}\,\pi R^2 \sim 100 \text{ GeV}$$

and yet, not so much the size of this background, but rather its **fluctuations**, are the real obstacle to its subtraction

Heavy Ion Jet Algorithms

Standard approach: correct **before/during** clustering.

- p_t cut ~ 1-2 GeV

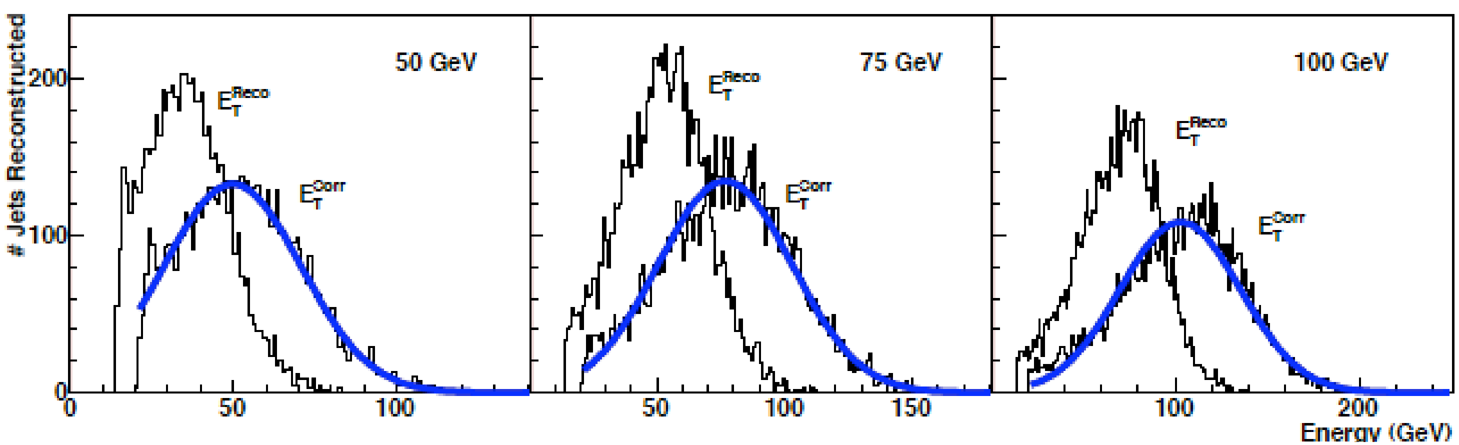
Eliminates most background, but not collinear safe.
Requires a posteriori correction.
How does it affect determination of quenching?

- subtract energy from calorimeter cells

Negative energy cells?
Experiment dependent

Example: **A Cone Jet-Finding Algorithm for Heavy-Ion Collisions at LHC Energies** nucl-ex/0609023

S-L Blyth[1,2], M J Horner[1,2], T Awes[3], T Cormier[4], H Gray[1,2], J L Klay[5], S R Klein[1], M van Leeuwen[1], A Morsch[6], G Odyniec[1] and A Pavlinov[4]

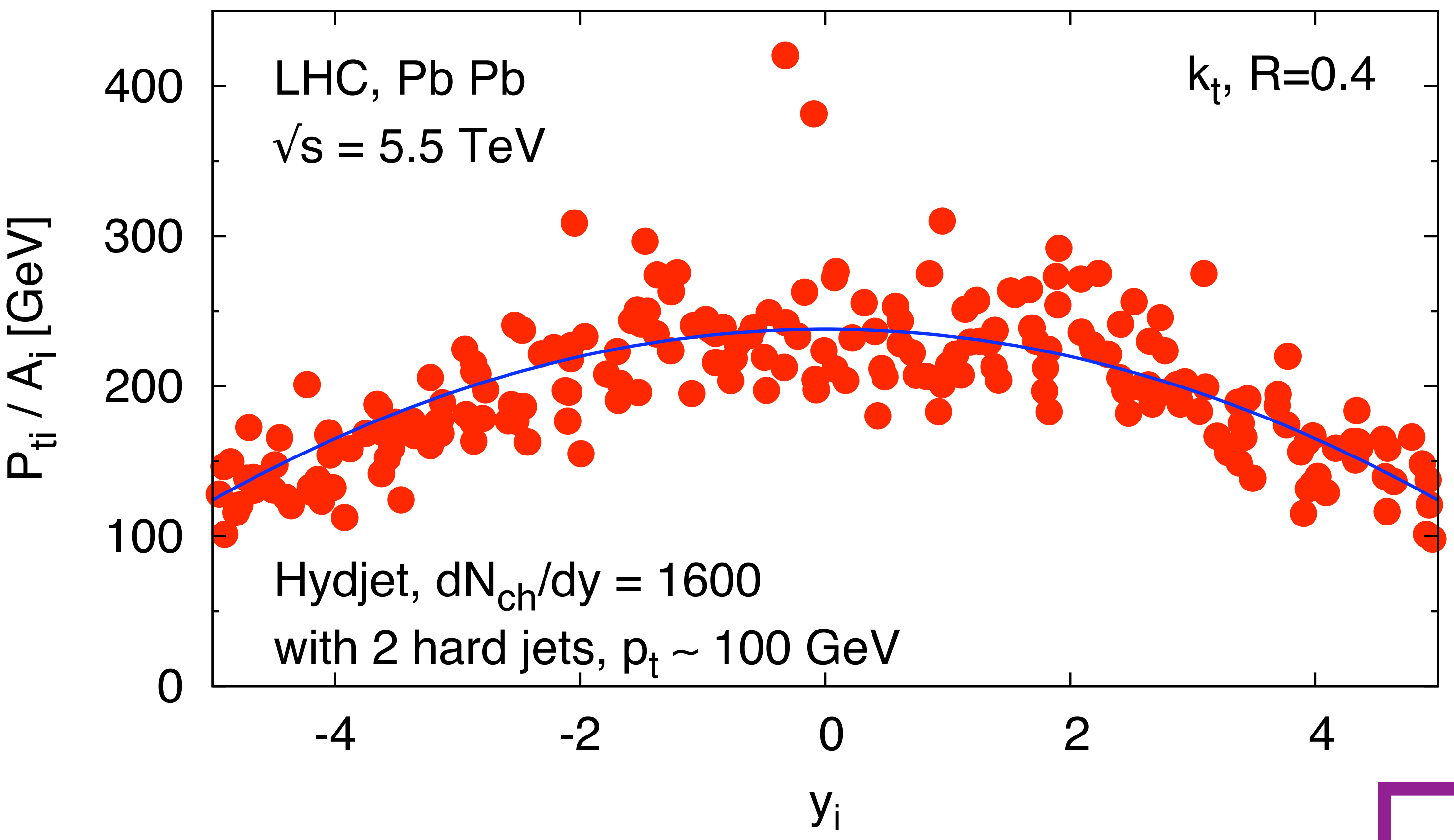


	50 GeV jets + HIJING	75 GeV jets + HIJING	100 GeV jets + HIJING
$\langle E_T^{Reco} \rangle \pm \sigma$ (GeV)	34 ± 14	52 ± 18	70 ± 22
$\langle E_T^{Corr} \rangle \pm \sigma$ (GeV)	50 ± 21	77 ± 26	103 ± 33

Proposal for Heavy Ion Collisions

Use the same approach (area-based) proposed for pileup:

- study transverse momentum/area of each jet
- subtract contribution <u>proportional to area of each jet</u>

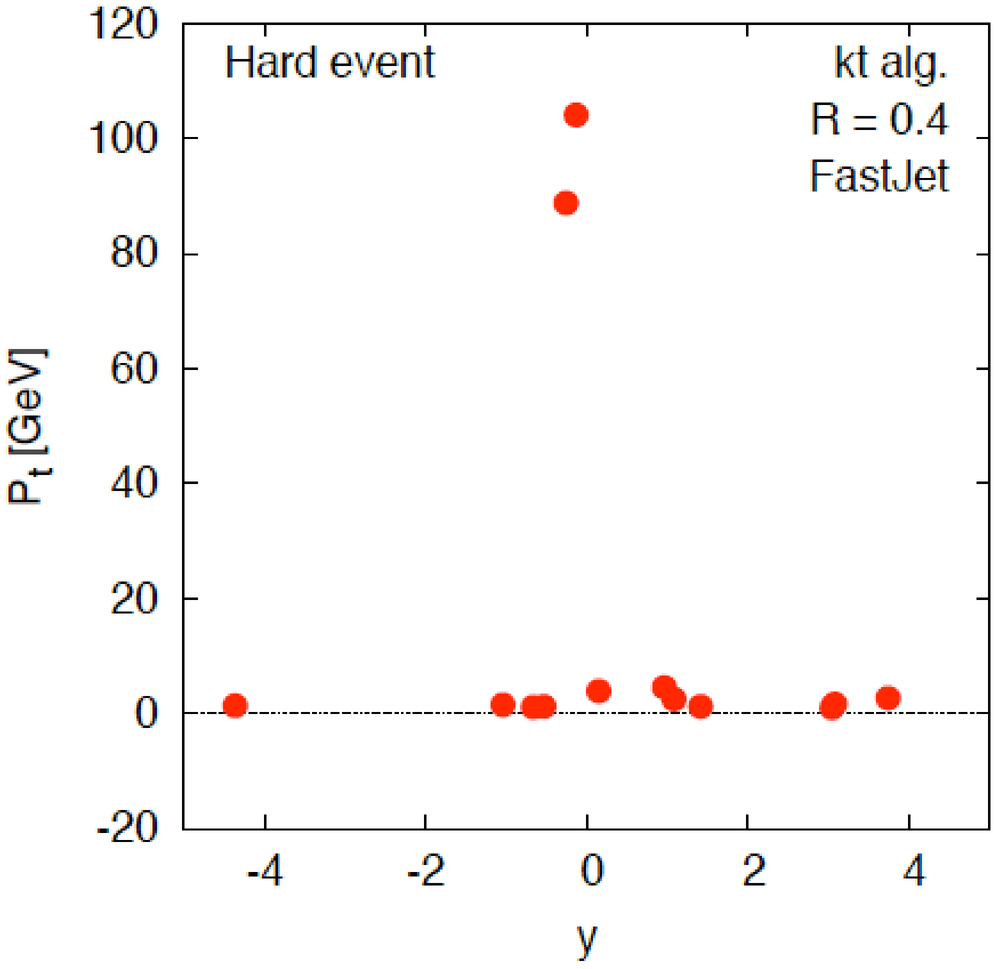


This is what improves the resolution

NB. No minimum p_t cut ever used

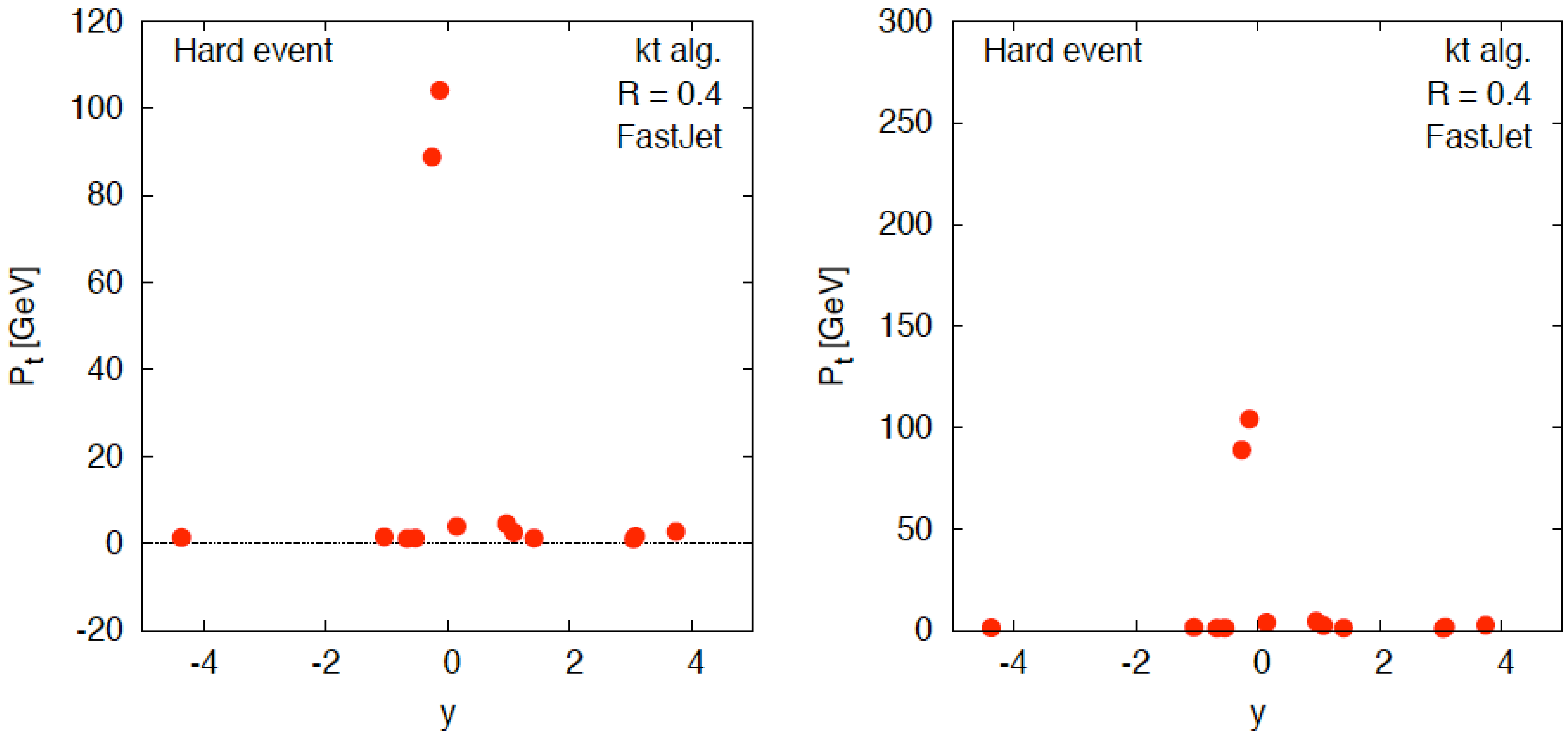
Minor modification: fit a parabola (or any appropriate shape).
One can also study a subregion and extract the local background level.

Background subtraction in HI event



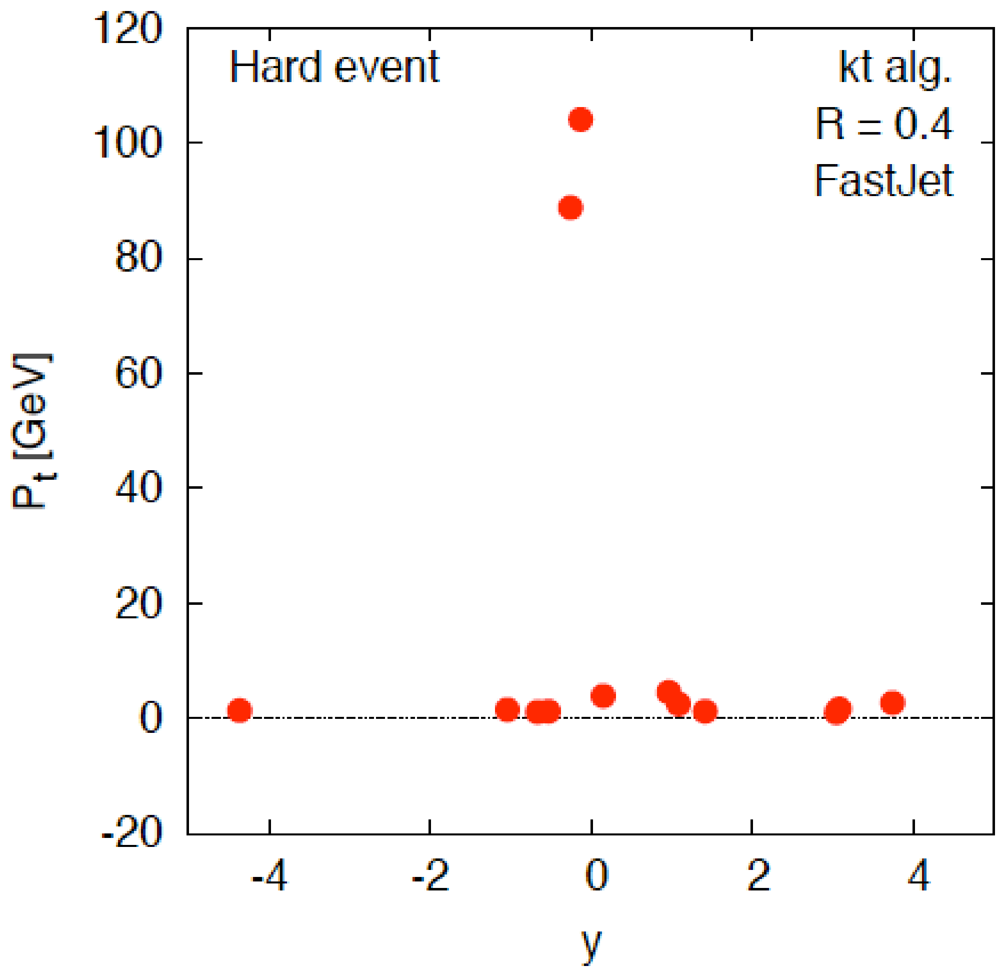
Start with a hard dijet event

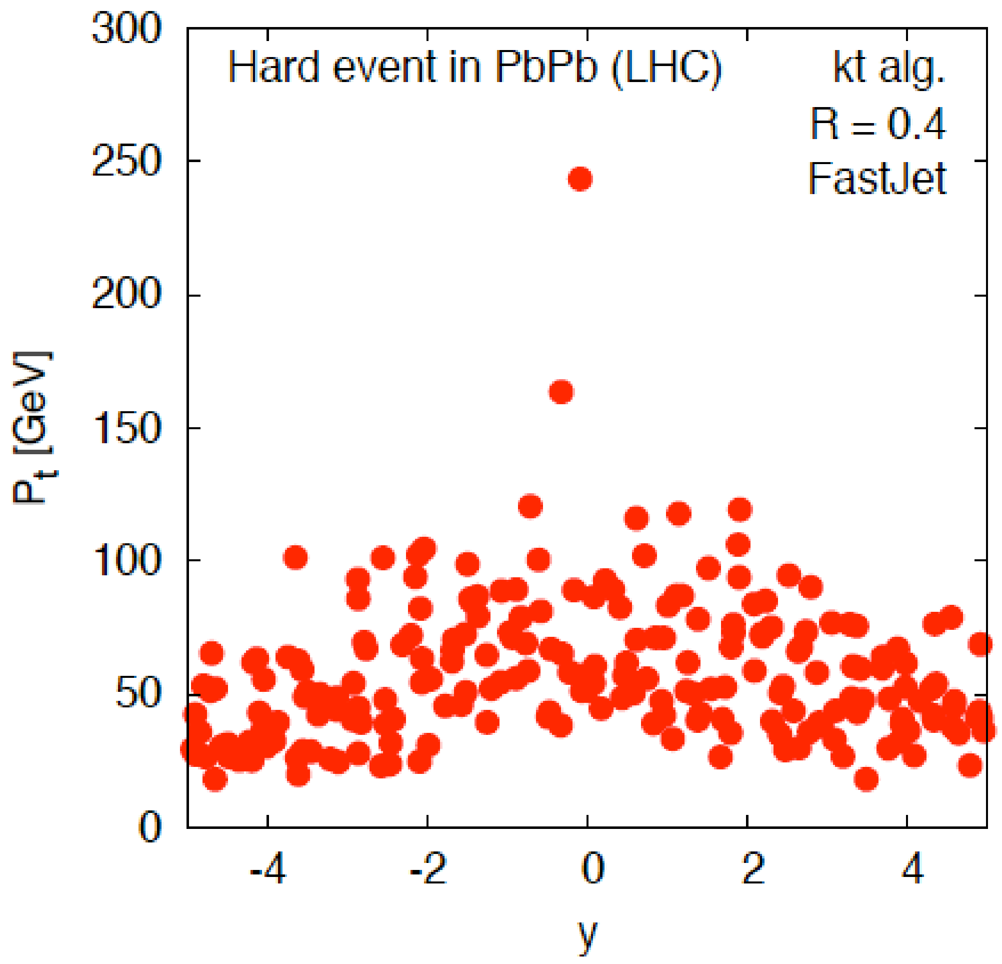
Background subtraction in HI event



Same event on a different scale

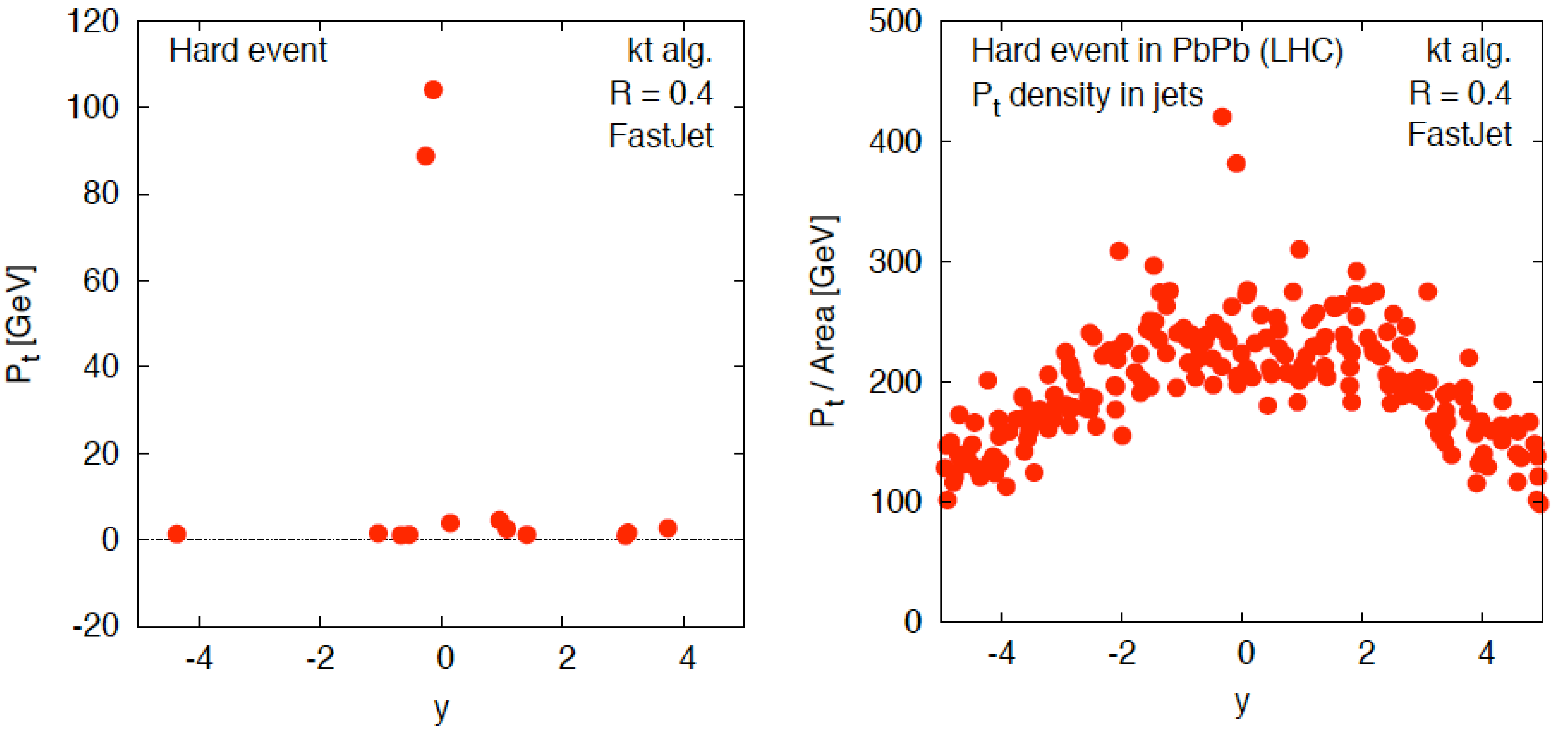
Background subtraction in HI event





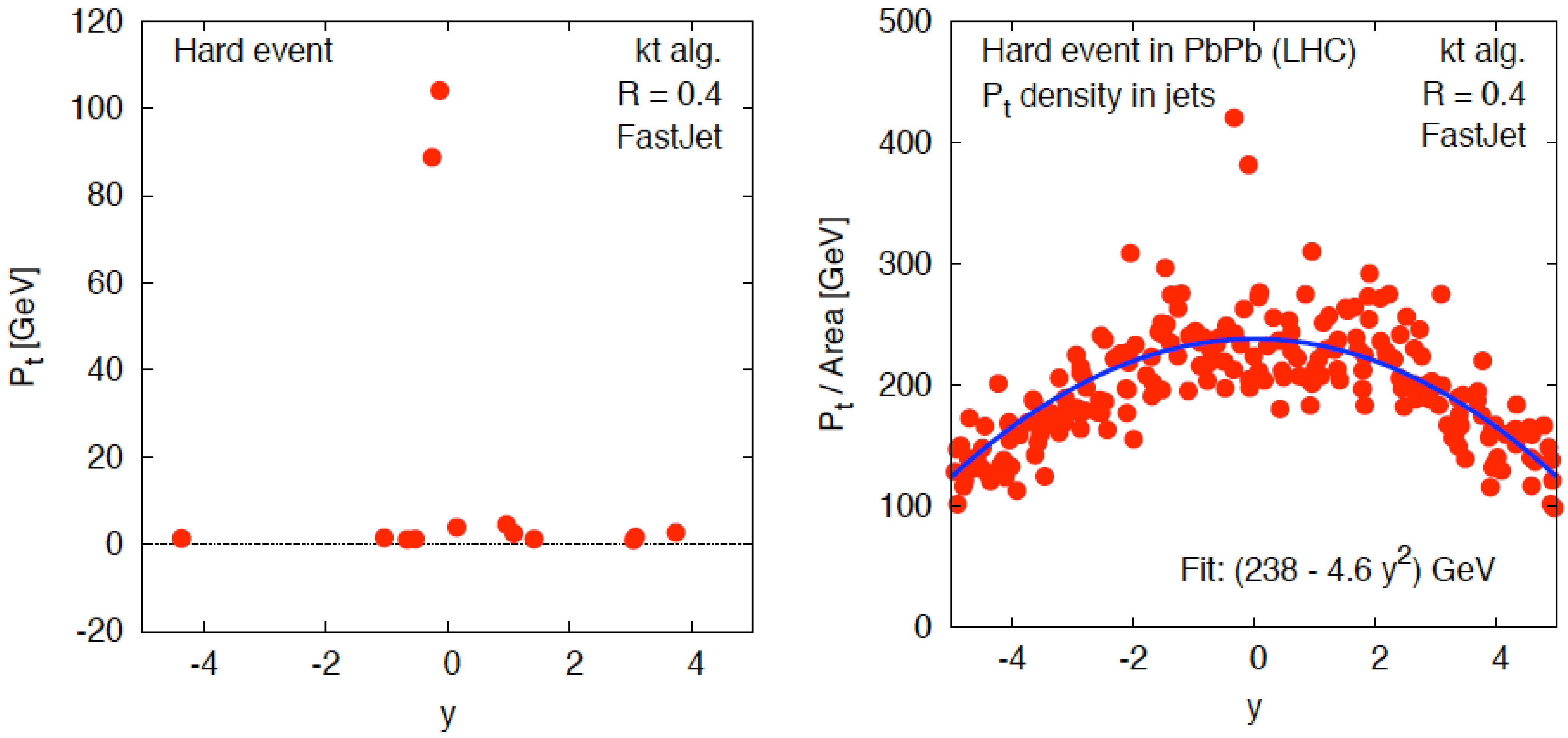
Embed it into a central Hydjet Pb Pb event

Background subtraction in HI event



Look at P_t/Area for each jet

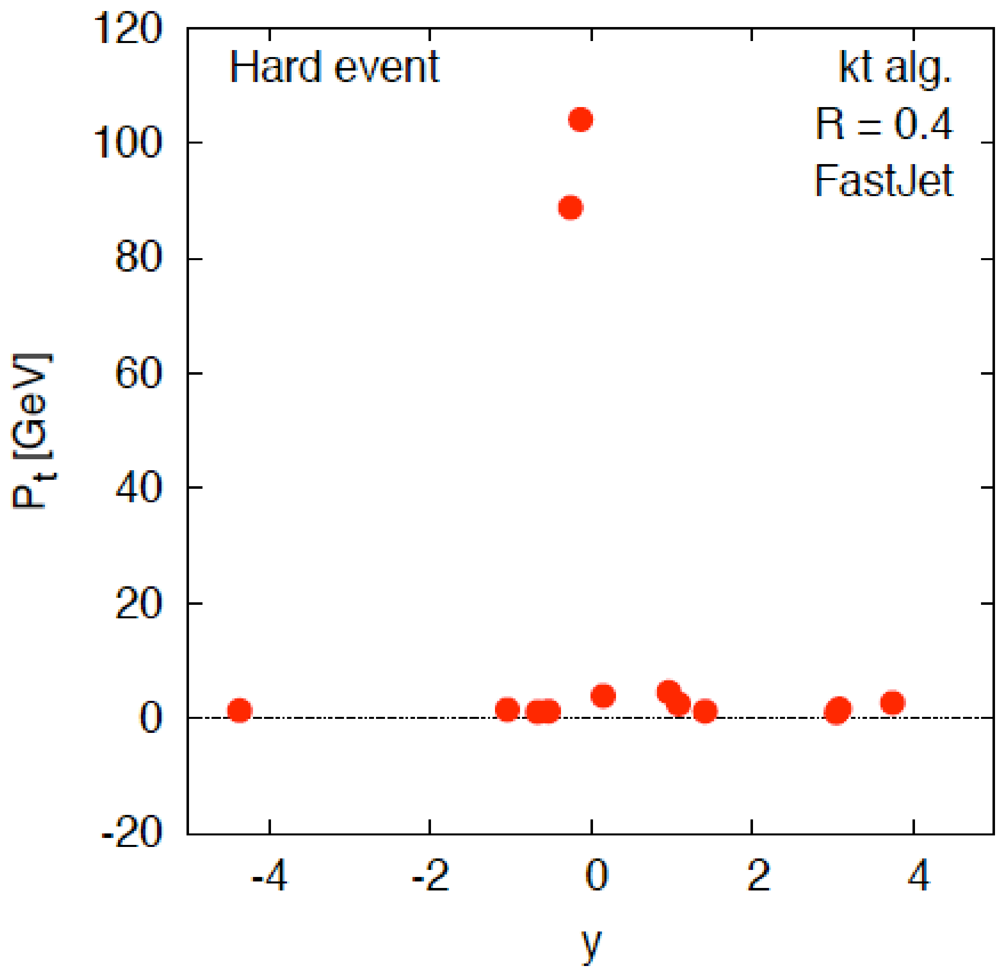
Background subtraction in HI event

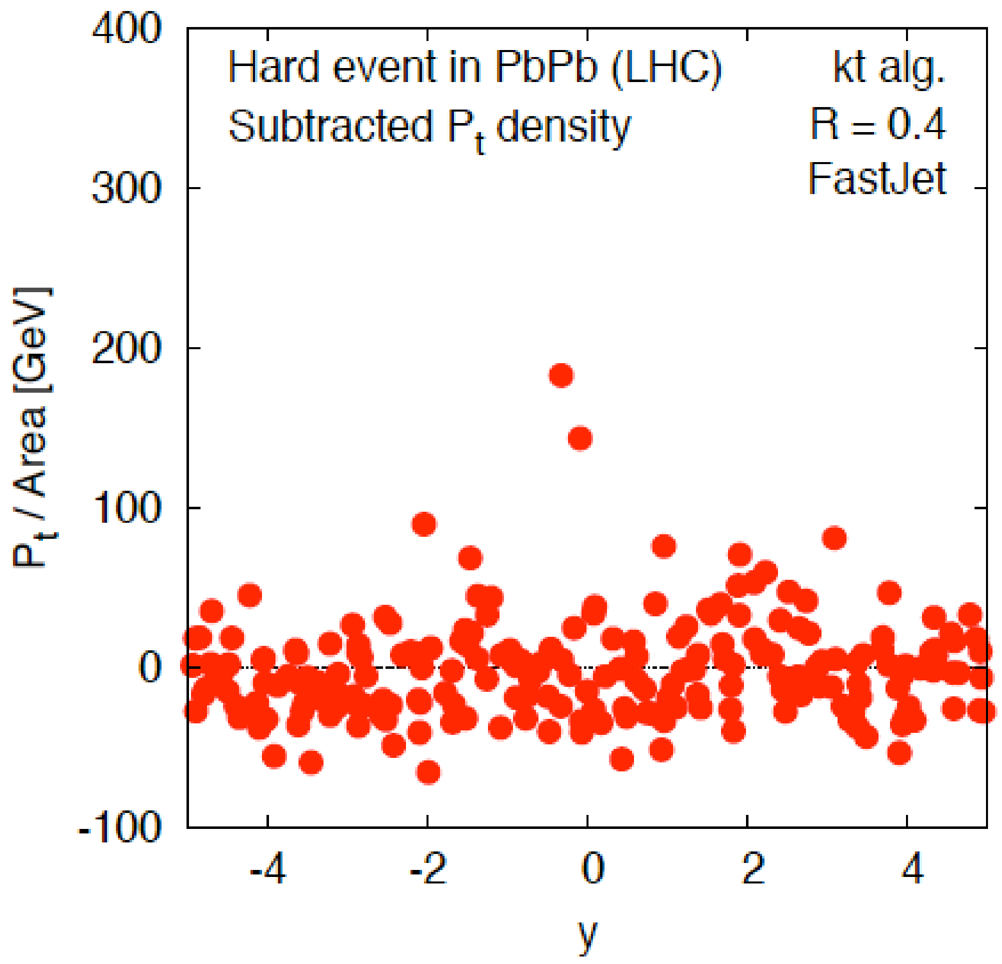


Fit the background $\rho(y)$

[NB: more general functional form needs investigating]

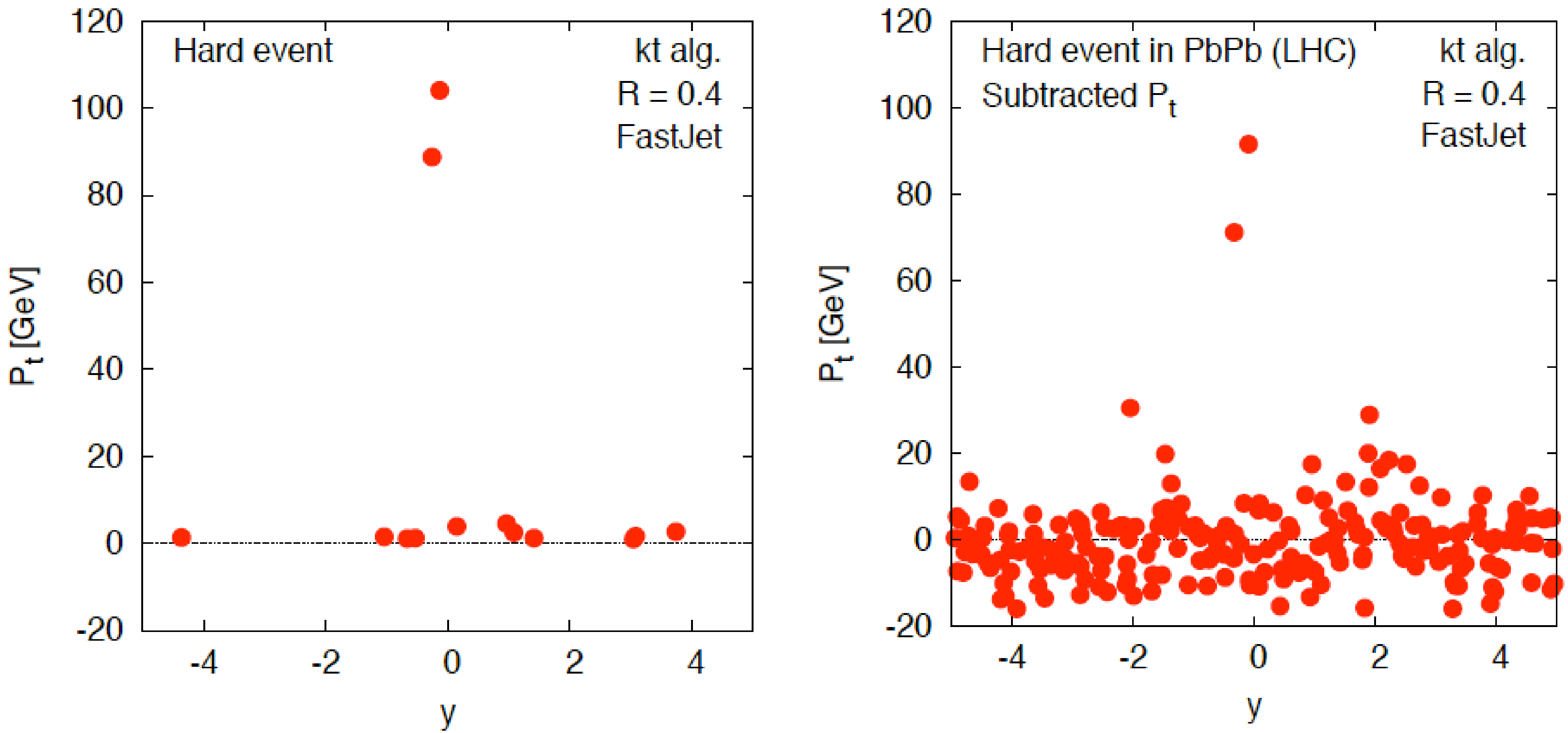
Background subtraction in HI event





Subtract $\rho(y)$ from P_t/Area for each jet

Background subtraction in HI event

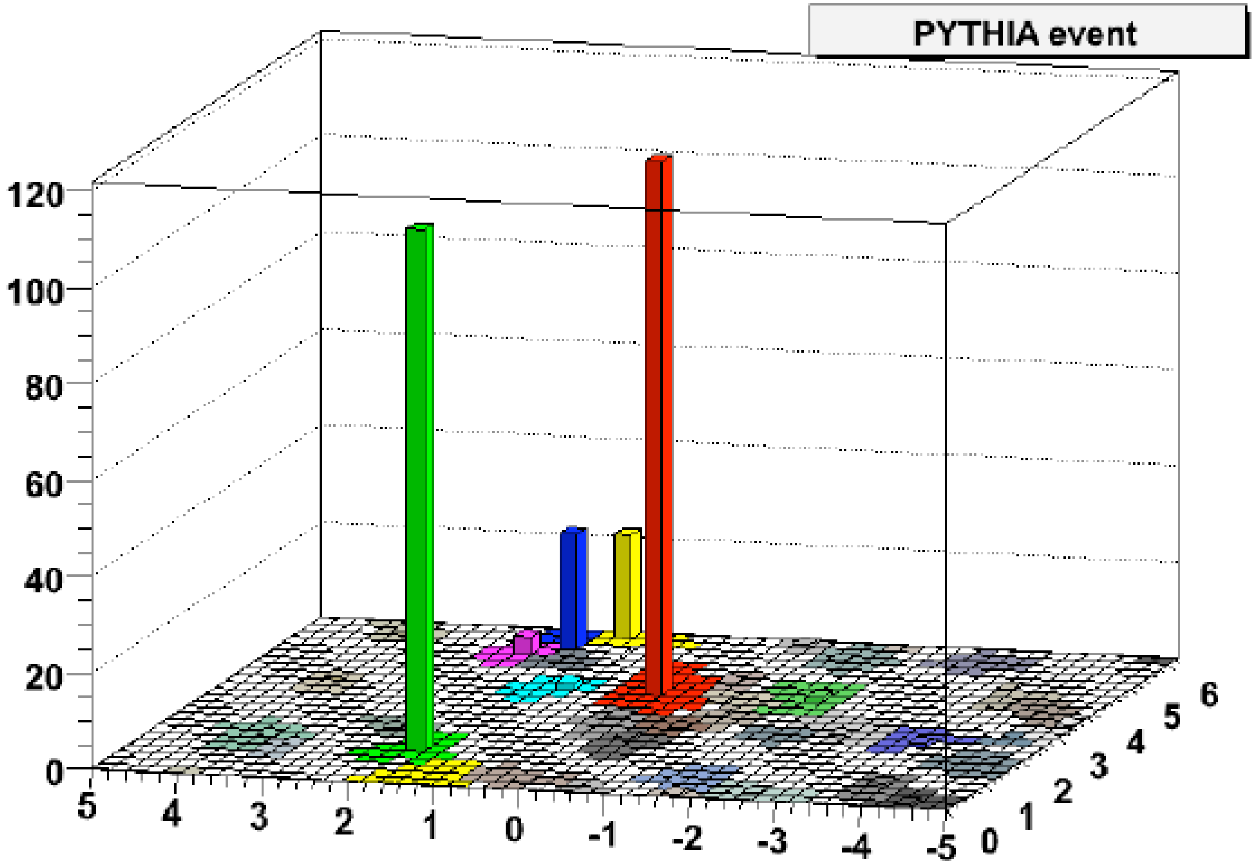


Look at resulting corrected $P_t = P_{t,orig} - \rho(y) \times \text{Area}$

Hard jets with roughly correct P_t and y emerge clearly!

A less easy piece

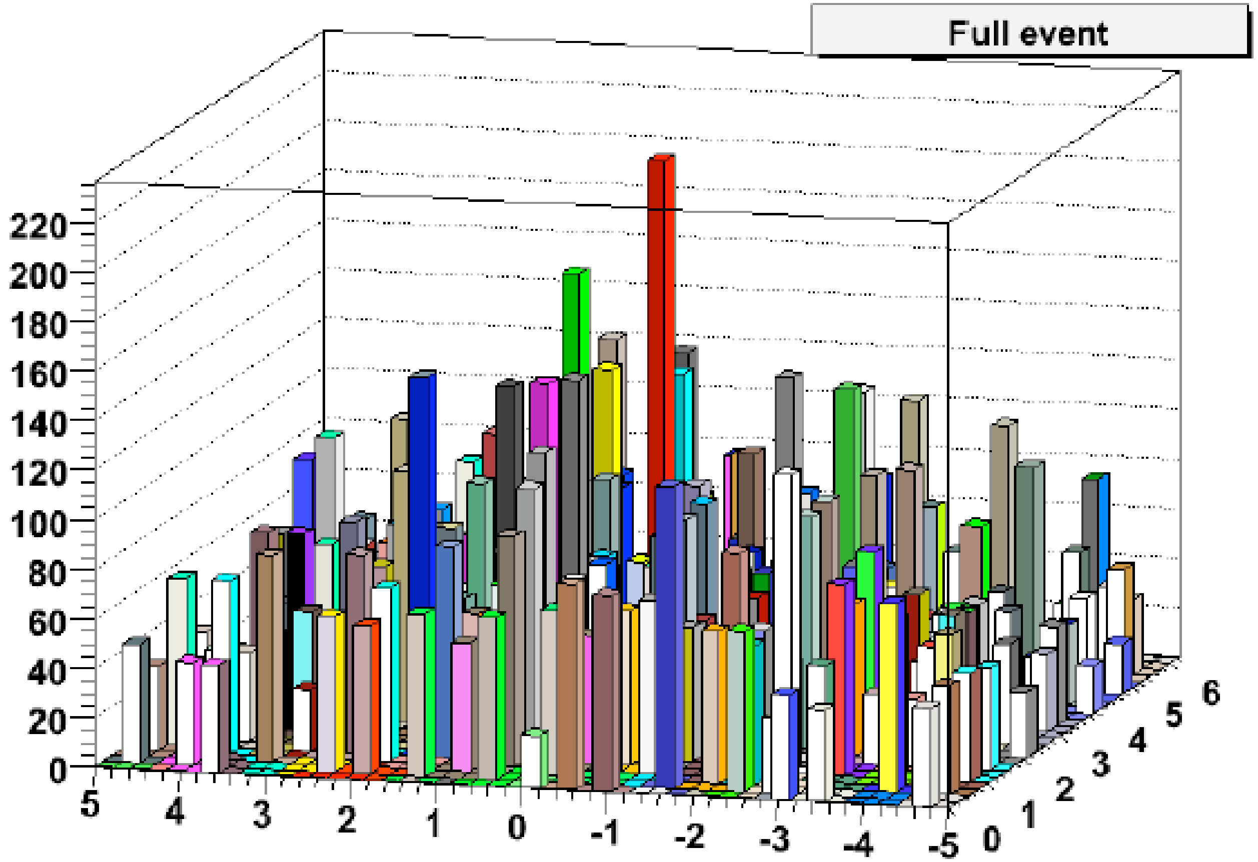
A **100 GeV** dijet PYTHIA event embedded in a HYDJET Pb-Pb one at the LHC



1. The *pp* hard event generated by PYTHIA only
2. The same event embedded in the whole Pb-Pb collisions
3. The result of the subtraction of the background

A less easy piece

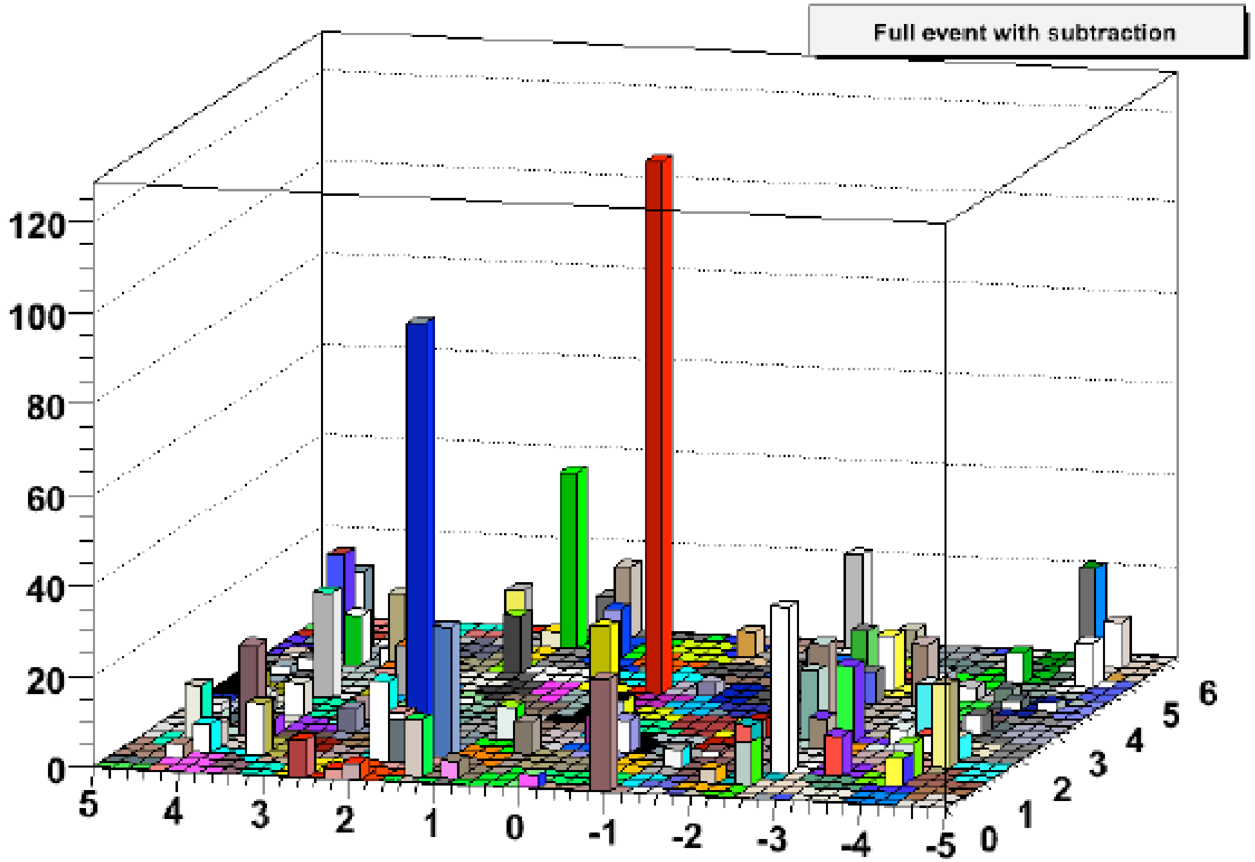
A **100 GeV** dijet PYTHIA event embedded in a HYDJET Pb-Pb one at the LHC



1. The *pp* hard event generated by PYTHIA only
2. The same event embedded in the whole Pb-Pb collisions
3. The result of the subtraction of the background

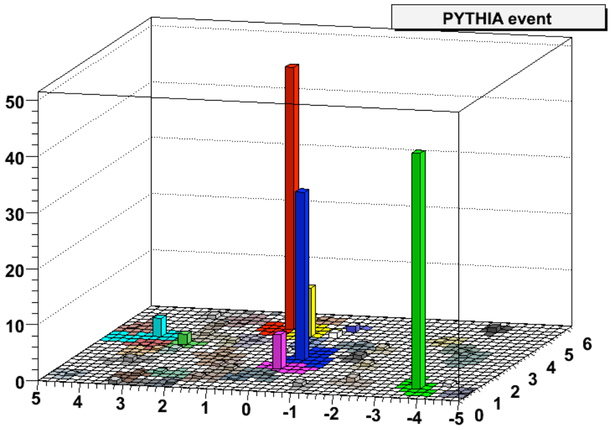
A less easy piece

A **100 GeV** dijet PYTHIA event embedded in a HYDJET Pb-Pb one at the LHC



1. The *pp* hard event generated by PYTHIA only
2. The same event embedded in the whole Pb-Pb collisions
3. The result of the subtraction of the background

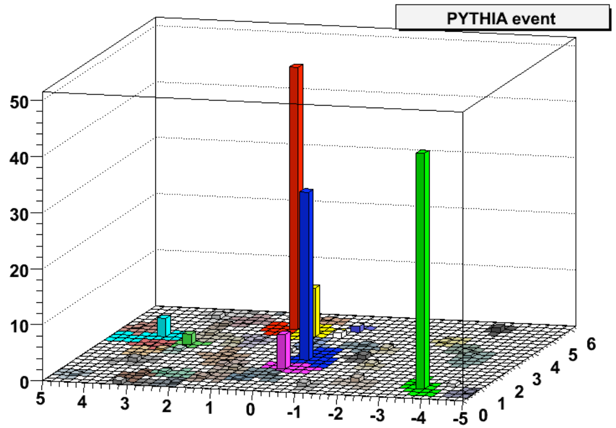
40 GeV jets

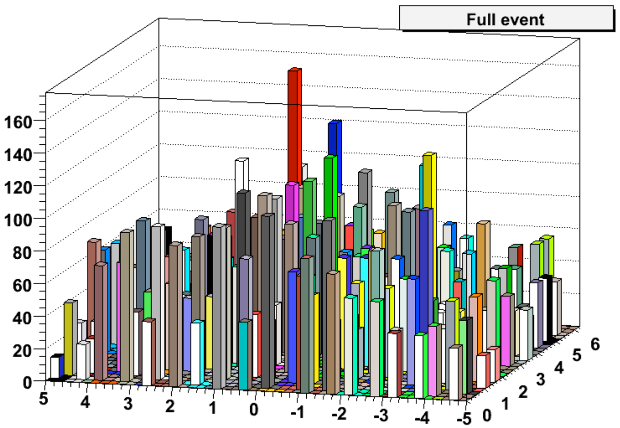



```
## hard event
# ijet      rap      phi       Pt      area
   0     1.138    4.990    46.696    0.807
   1    -3.693    0.982    41.942    0.813
   2    -0.166    2.638    29.912    1.143
   3     0.599    4.654     8.716    0.934
   4     0.054    1.967     6.157    0.553
   5     3.880    3.941     3.238    1.073
   6     3.001    3.589     1.840    0.622
   7    -0.256    5.169     1.126    0.413

## full event subtraction
#
```

40 GeV jets

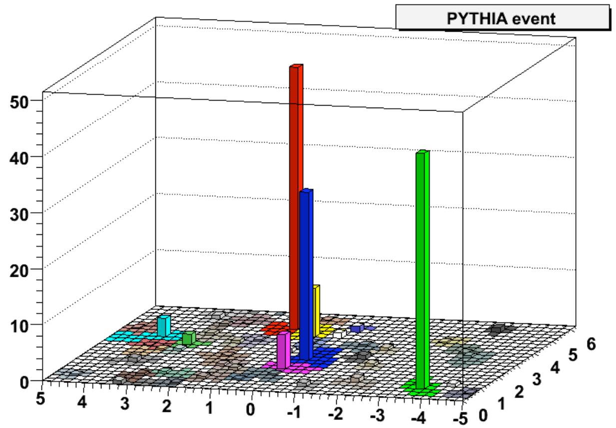


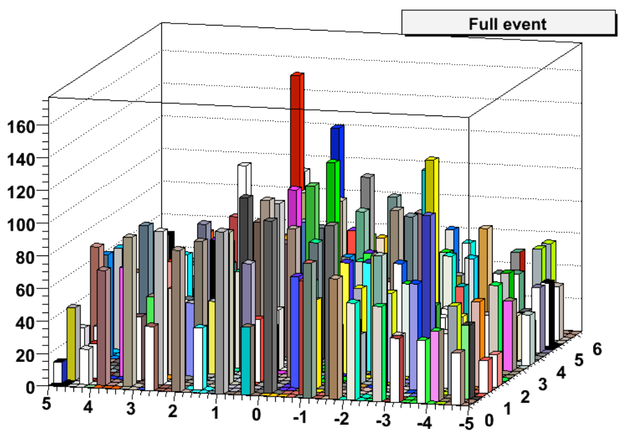


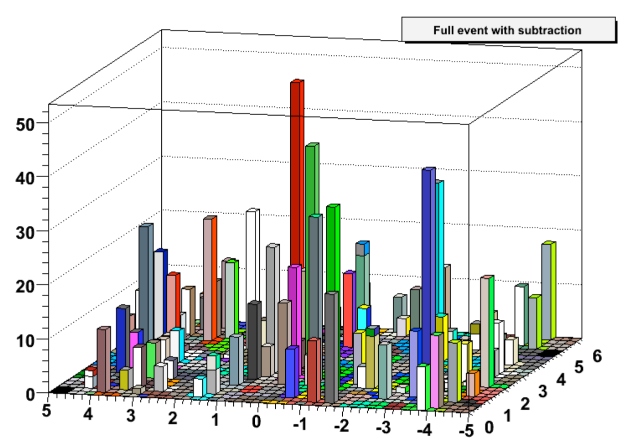

```
## hard event
# ijet      rap       phi          Pt     area
   0     1.138     4.990      46.696    0.807
   1    -3.693     0.982      41.942    0.813
   2    -0.166     2.638      29.912    1.143
   3     0.599     4.654       8.716    0.934
   4     0.054     1.967       6.157    0.553
   5     3.880     3.941       3.238    1.073
   6     3.001     3.589       1.840    0.622
   7    -0.256     5.169       1.126    0.413

## full event subtraction
# ijet      rap       phi          Pt     area
   0     1.094     4.945     160.668    0.521
   1    -1.179     1.530     134.888    0.470
   2     0.259     5.016     127.540    0.635
   3    -2.602     3.506     123.007    0.635
   4     0.081     2.138     111.034    0.406
   5    -2.250     4.253     110.653    0.406
   6    -0.156     2.741     109.869    0.305
   7    -3.620     0.856     109.479    0.457
   8     2.107     4.399     107.934    0.419
   9     0.588     1.614     107.771    0.470
  10     1.082     1.989     106.716    0.432
  11    -0.165     0.441     105.569    0.495
  12    -1.498     0.482     104.687    0.406
  13    -0.489     4.917     100.438    0.406
  14     0.949     0.054      99.661    0.445
  15     0.649     2.721      98.143    0.343
  16     1.406     5.541      95.984    0.457
  17     2.521     0.778      93.890    0.483
```

40 GeV jets







hard event

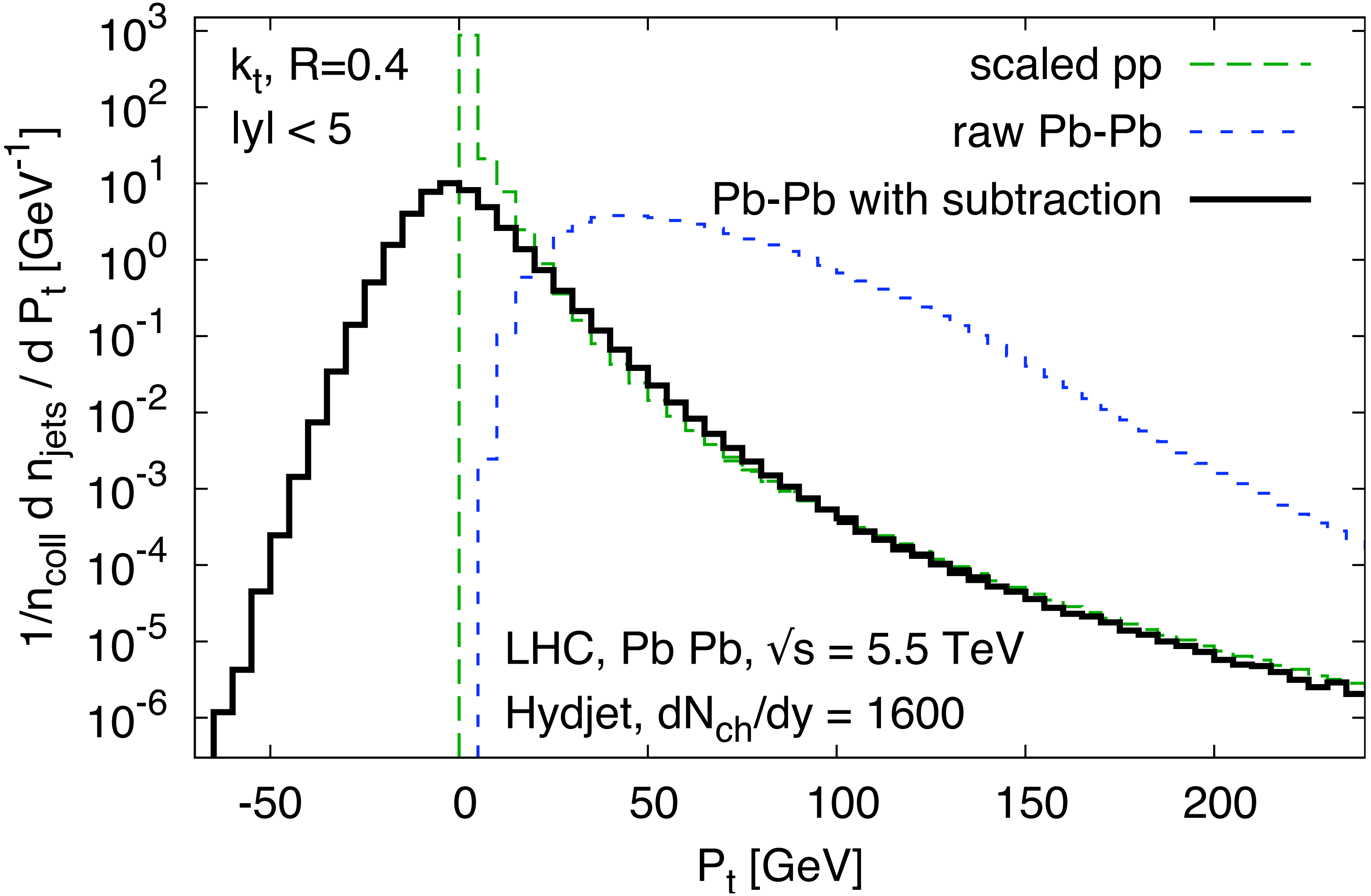
# ijet	rap	phi	Pt	area
0	1.138	4.990	46.696	0.807
1	-3.693	0.982	41.942	0.813
2	-0.166	2.638	29.912	1.143
3	0.599	4.654	8.716	0.934
4	0.054	1.967	6.157	0.553
5	3.880	3.941	3.238	1.073
6	3.001	3.589	1.840	0.622
7	-0.256	5.169	1.126	0.413

full event subtraction

# ijet	rap	phi	Pt	area	[illegible]	(rap corr	phi corr	Pt corr)ex
0	1.094	4.945	160.668	0.521	[illegible]	1.073	4.937	48.411
1	-1.179	1.530	134.888	0.470	[illegible]	-1.203	1.539	33.736
2	0.259	5.016	127.540	0.635	[illegible]	100000.000	0.000	0.000
3	-2.602	3.506	123.007	0.635	[illegible]	-2.546	3.974	8.290
4	0.081	2.138	111.034	0.406	[illegible]	0.109	2.211	20.413
5	-2.250	4.253	110.653	0.406	[illegible]	-2.308	4.313	32.292
6	-0.156	2.741	109.869	0.305	[illegible]	-0.143	2.696	41.763
7	-3.620	0.856	109.479	0.457	[illegible]	-3.689	0.987	42.644
8	2.107	4.399	107.934	0.419	[illegible]	2.074	4.369	25.428
9	0.588	1.614	107.771	0.470	[illegible]	0.773	2.281	5.529
10	1.082	1.989	106.716	0.432	[illegible]	1.023	2.011	13.914
11	-0.165	0.441	105.569	0.495	[illegible]	100000.000	0.000	0.000
12	-1.498	0.482	104.687	0.406	[illegible]	-1.463	0.427	20.104
13	-0.489	4.917	100.438	0.406	[illegible]	-0.276	5.166	10.850
14	0.949	0.054	99.661	0.445	[illegible]	0.685	5.813	4.184
15	0.649	2.721	98.143	0.343	[illegible]	0.664	2.617	23.367
16	1.406	5.541	95.984	0.457	[illegible]	1.290	0.769	4.452
17	2.521	0.778	93.890	0.483	[illegible]	2.503	0.641	4.706

NB. Second and third hardest jets are down in 6th and 7th position in full event but they are recovered after subtraction

Inclusive jets in PbPb at LHC



The scaled pp cross section is recovered after subtraction

NB. No minimum pt cut
No a posteriori Monte Carlo correction

Conclusions

- ☑ 'Proper' jet algorithms can (and probably should) be used in heavy ion collisions
- ☑ Given a proper jet algorithm, jet areas can be defined
- ☑ They can be used to **estimate** the level of a uniformly distributed soft background
- ☑ They can be used to **subtract** the background contribution from the hard jets