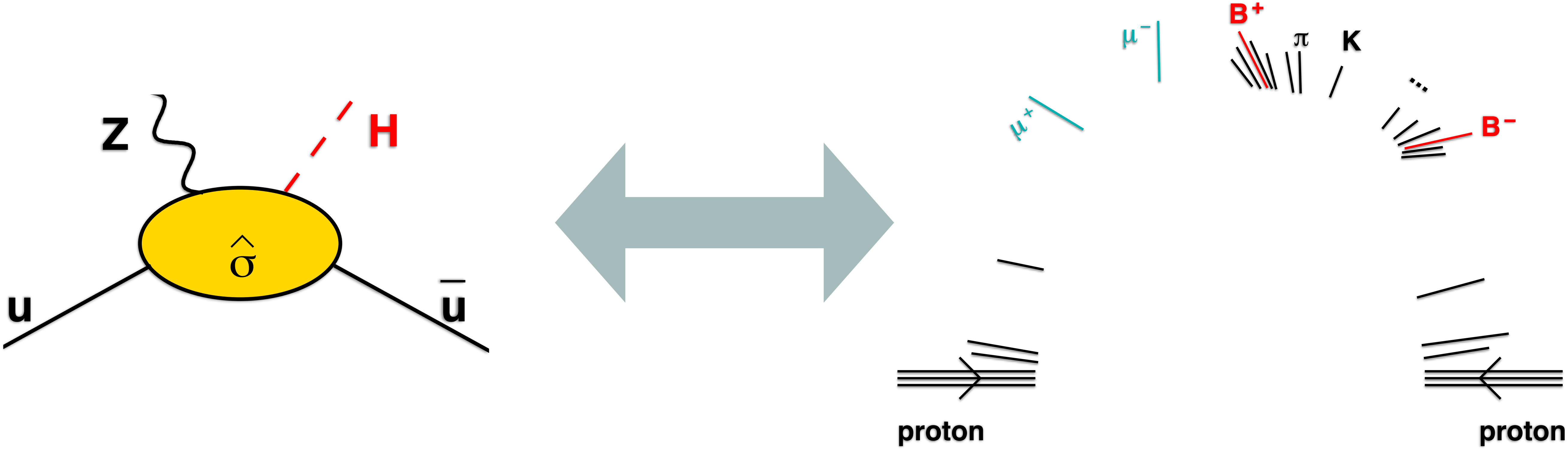
QCD (for Colliders)
Lecture 3

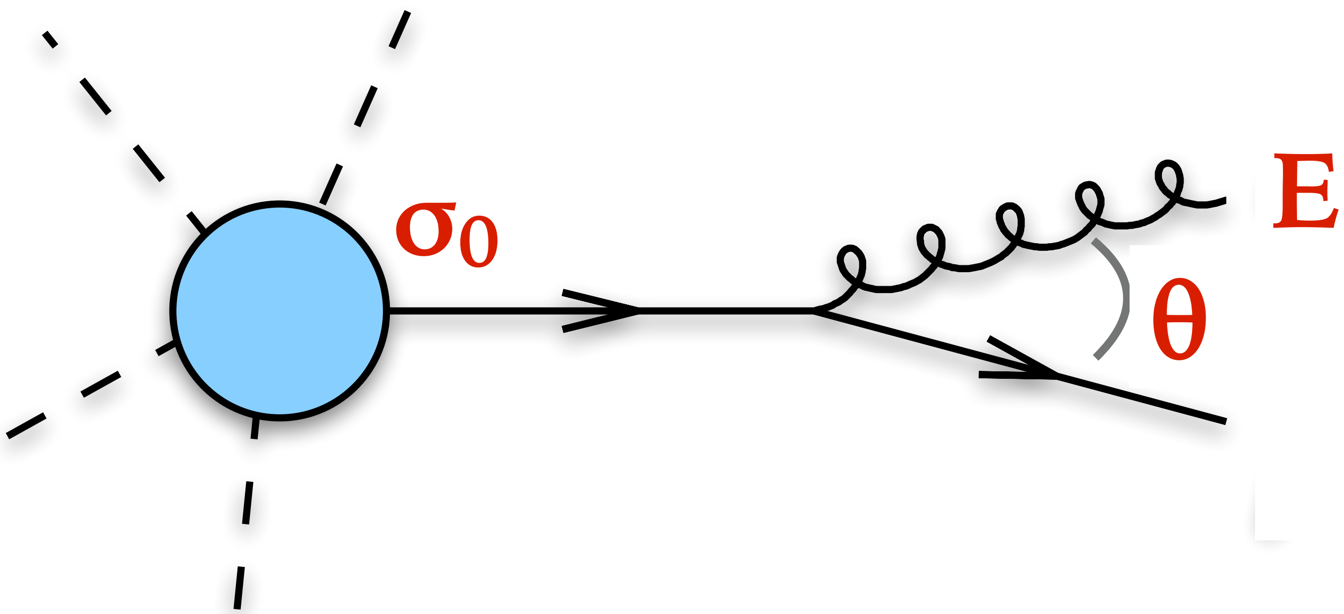
Gavin Salam, CERN

Fourth Asia-Europe-Pacific School of High-Energy Physics
September 2018, Qhy Nhon, Vietnam

the real world?



GLUON EMISSION FROM A QUARK



Consider an emission with

- energy $E \ll \sqrt{s}$ ("soft")
- angle $\theta \ll 1$ ("collinear" wrt quark)

Examine correction to some hard process with cross section σ_0

$$d\sigma \simeq \sigma_0 \times \frac{2\alpha_s C_F}{\pi} \frac{dE}{E} \frac{d\theta}{\theta}$$

This has a divergence when $E \to 0$ or $\theta \to 0$

[in some sense because of quark propagator going on-shell]

Suppose we're not inclusive — e.g. calculate probability of emitting a gluon

Probability P_g of emitting gluon from a quark with energy Q:

$$P_g \simeq \frac{2\alpha_s C_F}{\pi} \int^Q \frac{dE}{E} \int^1 \frac{d\theta}{\theta} \Theta(E\theta > Q_0)$$

We cut off the integral for transverse momenta ($p_T \simeq E\,\theta$) below some non-perturbative threshold Q_0.

On the grounds that perturbation theory doesn't apply for $p_T \sim \Lambda_{QCD}$
i.e. language of quarks and gluons becomes meaningless

With this cutoff, the result is

$$P_g \simeq \frac{\alpha_s C_F}{\pi} \ln^2 \frac{Q}{Q_0} + \mathcal{O}\left(\alpha_s \ln Q\right)$$

this is called a "double logarithm"
[it crops up all over the place in QCD]

Suppose we're not inclusive — e.g. calculate probability of emitting a gluon

Probability P_g of emitting gluon from a quark with energy Q:

$$P_g \simeq \frac{2\alpha_s C_F}{\pi} \int_{Q_0}^{Q} \frac{dE}{E} \int_{\frac{Q_0}{E}}^{1} \frac{d\theta}{\theta}$$

We cut off the integral for transverse momenta ($p_T \simeq E\,\theta$) below some non-perturbative threshold Q_0.

On the grounds that perturbation theory doesn't apply for $p_T \sim \Lambda_{QCD}$
i.e. language of quarks and gluons becomes meaningless

With this cutoff, the result is

$$P_g \simeq \frac{\alpha_s C_F}{\pi} \ln^2 \frac{Q}{Q_0} + \mathcal{O}\left(\alpha_s \ln Q\right)$$

this is called a "double logarithm"

[it crops up all over the place in QCD]

Suppose we're not inclusive — e.g. calculate probability of emitting a gluon

Suppose we take $Q_0 \sim \Lambda_{QCD}$, what do we get?

Let's use $\alpha_s = \alpha_s(Q) = 1/(2b \ln Q/\Lambda)$
[Actually over most of integration range this is optimistically small]

$$P_g \simeq \frac{\alpha_s C_F}{\pi} \ln^2 \frac{Q}{Q_0} \;\to\; \frac{C_F}{2b\pi} \ln \frac{Q}{\Lambda_{QCD}} \;\to\; \frac{C_F}{4b^2\pi\,\alpha_s}$$

Put in some numbers:

$Q = 100$ GeV, $\Lambda_{QCD} \simeq 0.2$ GeV, $C_F = 4/3$, $b(\equiv b_0) \simeq 0.6$

$$P_g \simeq 2.2$$

This is supposed to be an $O(\alpha_s)$ correction.

But the final result $\sim 1/\alpha_s$

QCD hates to not emit gluons!

correct way of doing it: with running coupling inside the integral

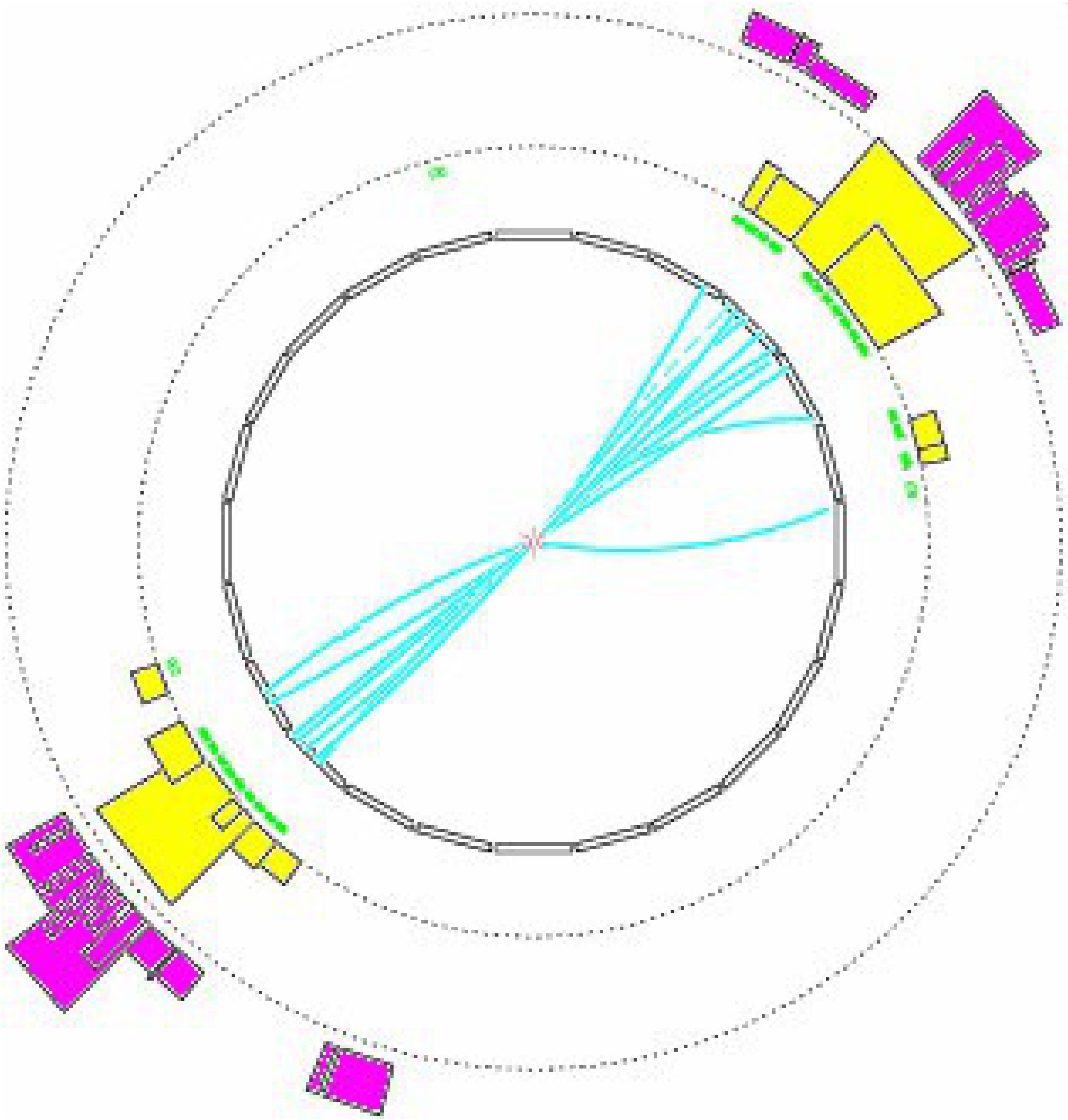
Adding running coupling is straightforward: just use $\alpha_s(p_t)$ with $p_t \simeq E\theta$, in the integrand:

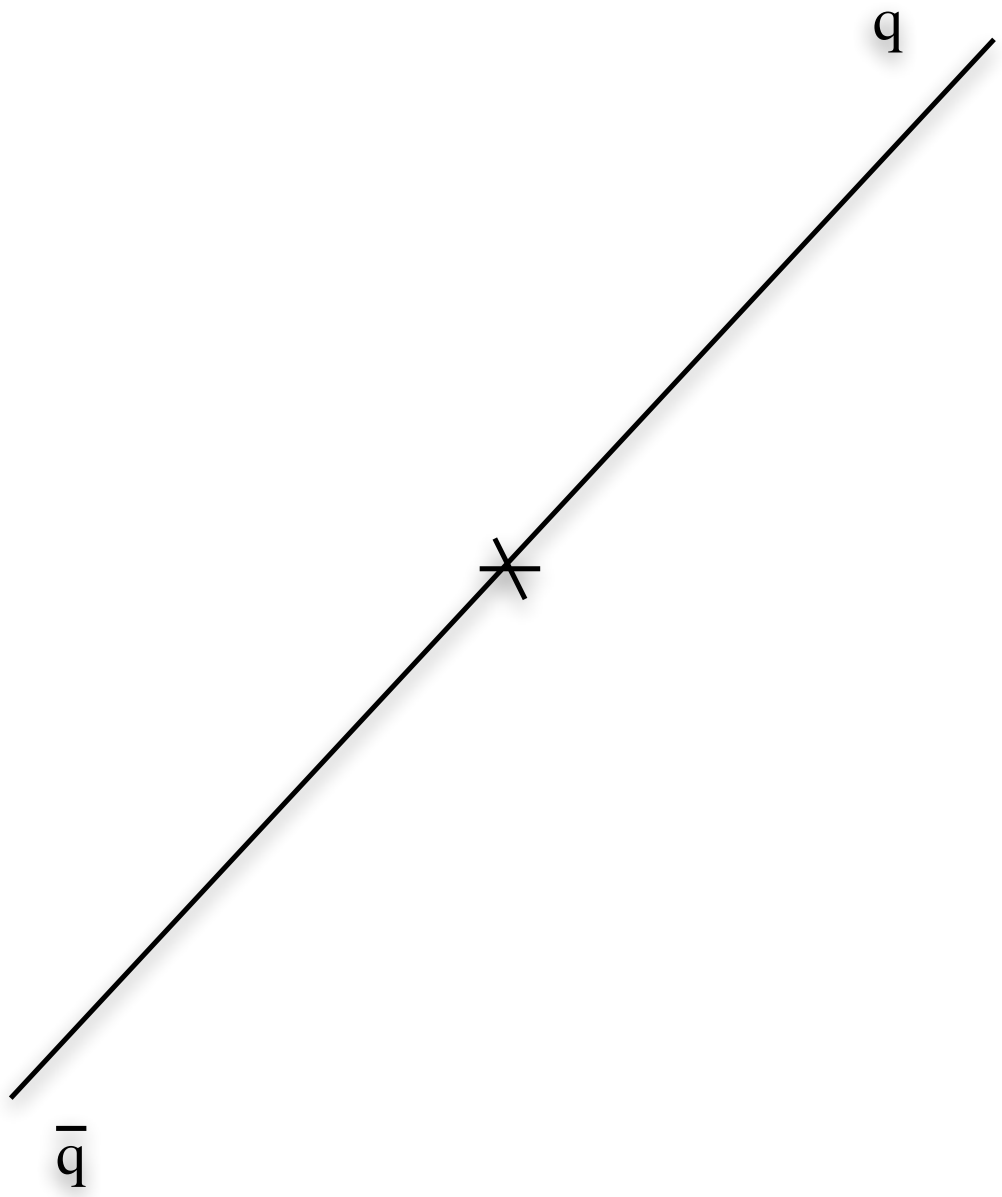
$$P_g = \frac{2C_F}{\pi}\int_{Q_0}^{Q}\frac{dp_t}{p_t}\,\alpha_s(p_t)\int_{p_t/Q}^{1}\frac{dz}{z} = \frac{C_F}{\pi b_0}\left(\ln\frac{Q}{\Lambda}\ln\ln\frac{Q}{\Lambda}+\cdots\right)$$

Structure of answer changes a bit: it's larger than $1/\alpha_s(Q)$, by a factor $\ln\ln Q/\Lambda$.

But to keep expressions simple in these lectures we'll often restrict ourselves to a fixed-coupling approximation.

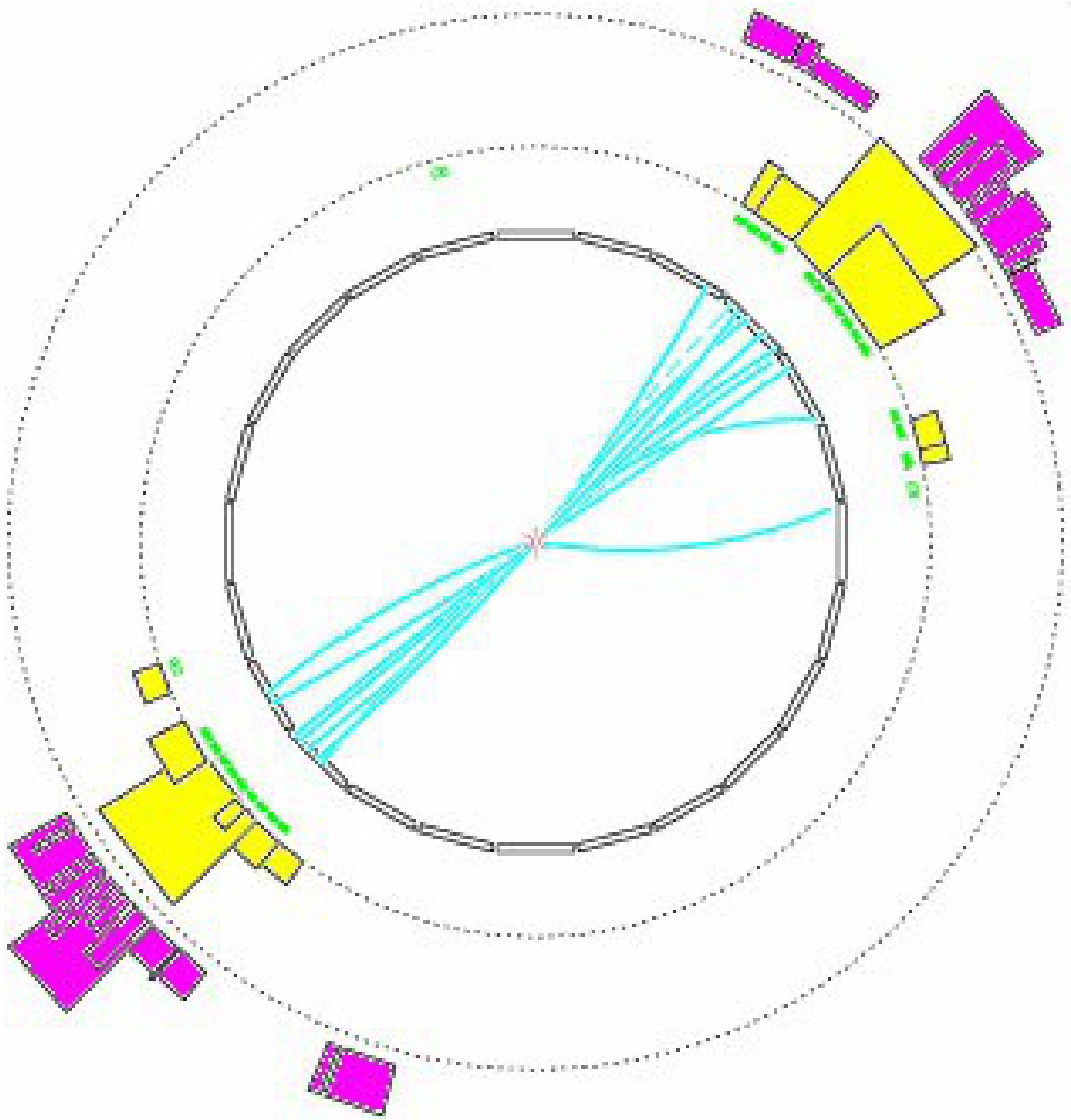
Picturing a QCD event

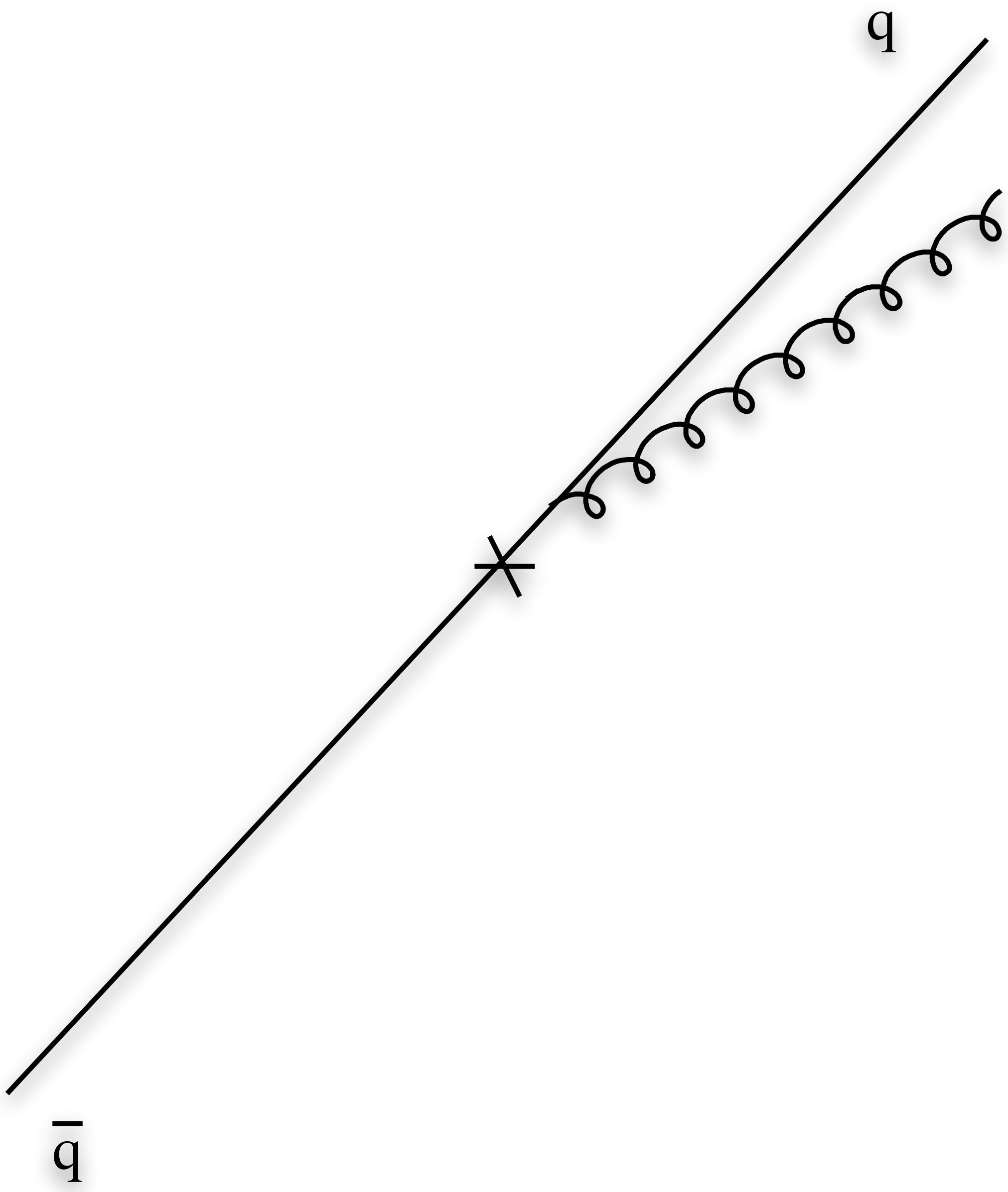




Start off with a qqbar system

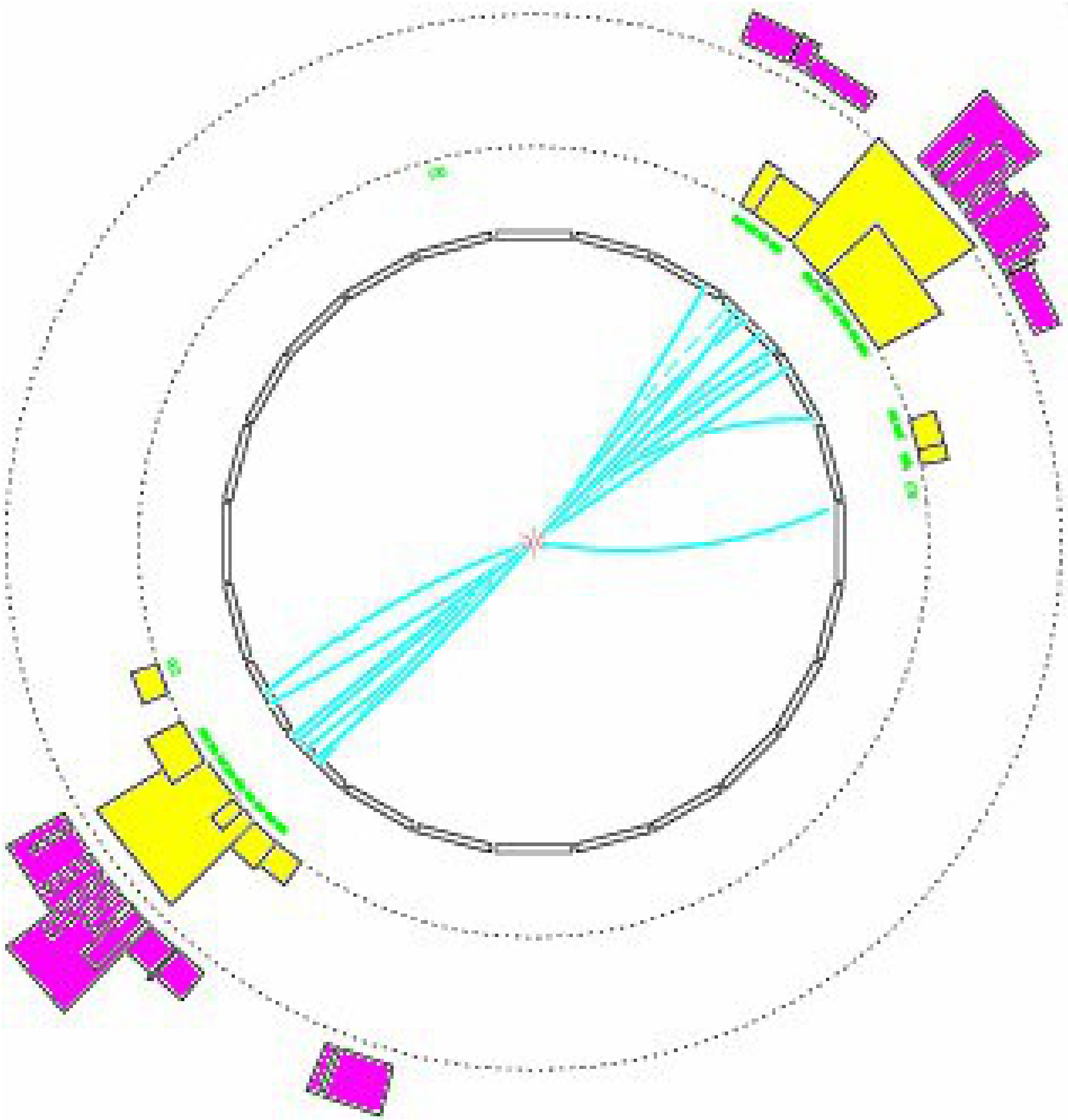
Picturing a QCD event

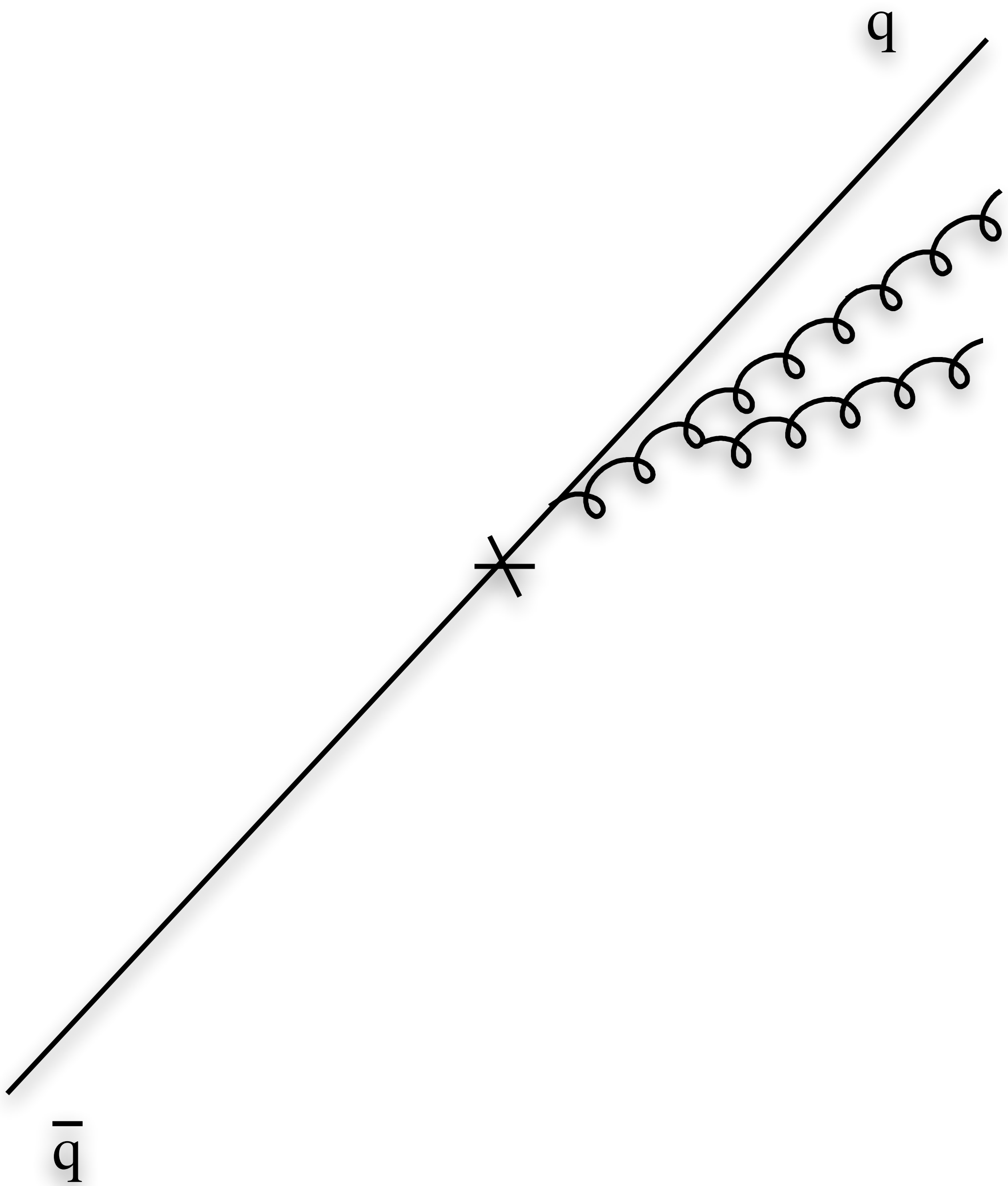




a gluon gets emitted at small angles

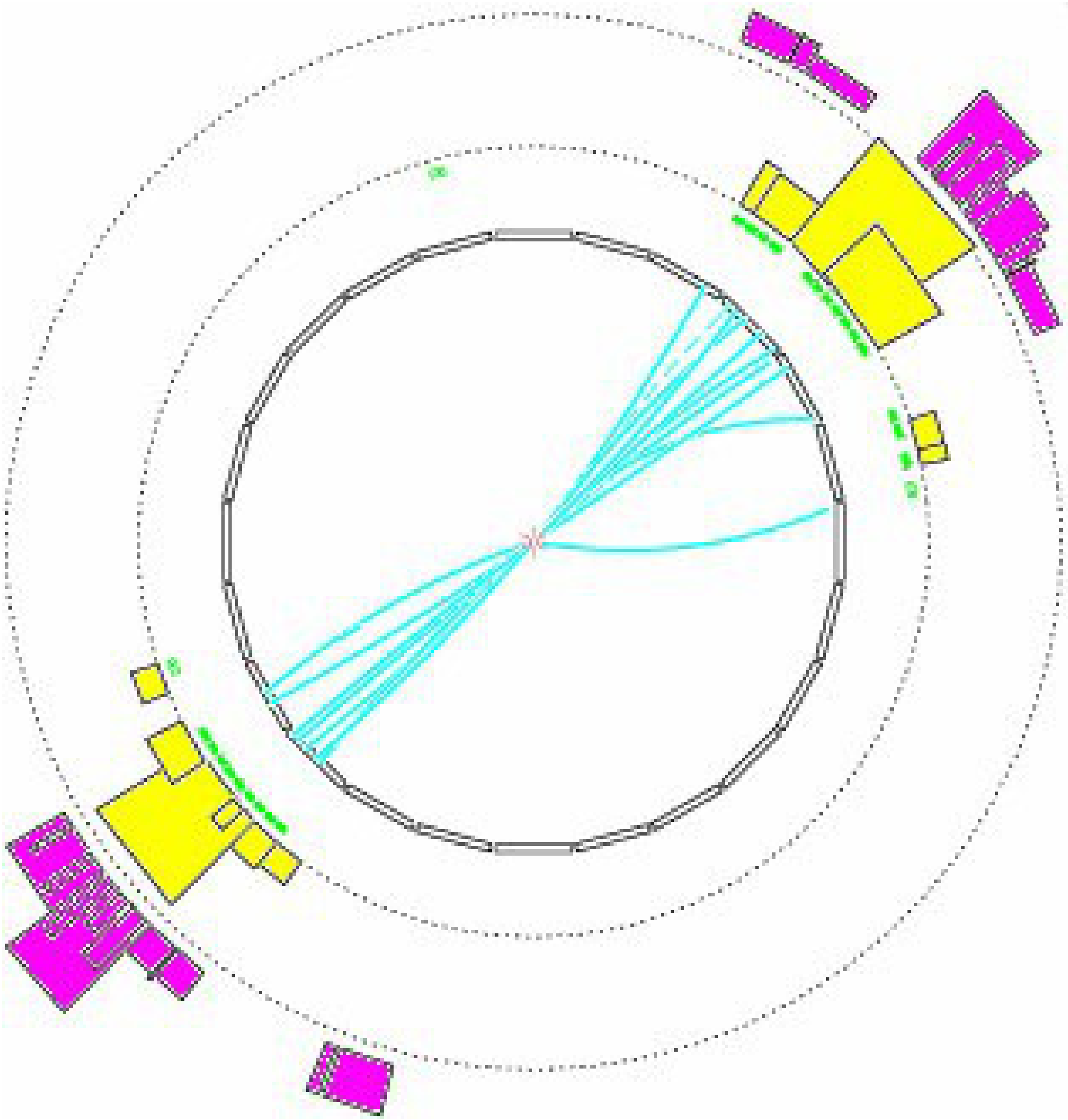
Picturing a QCD event

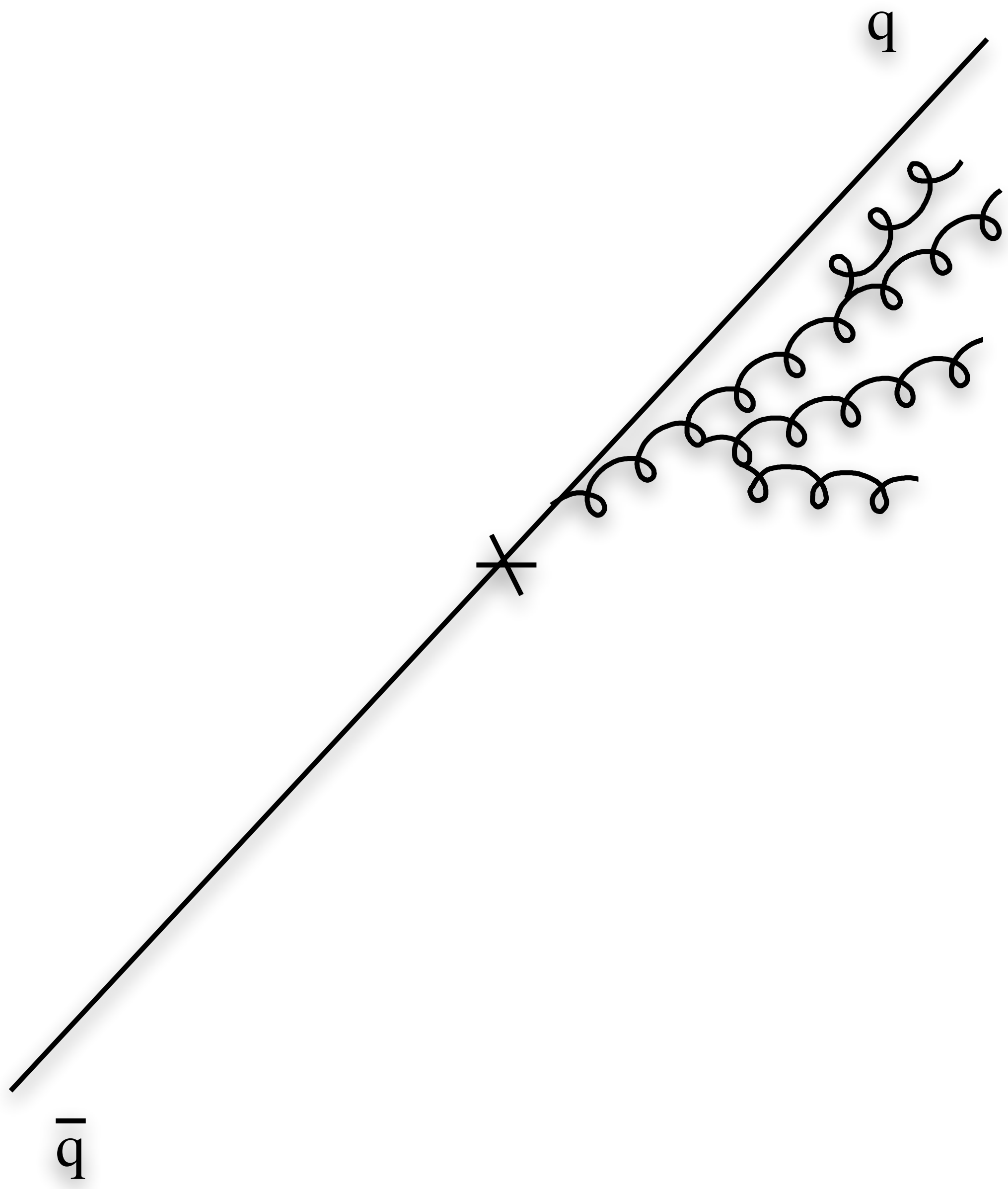




it radiates a further gluon

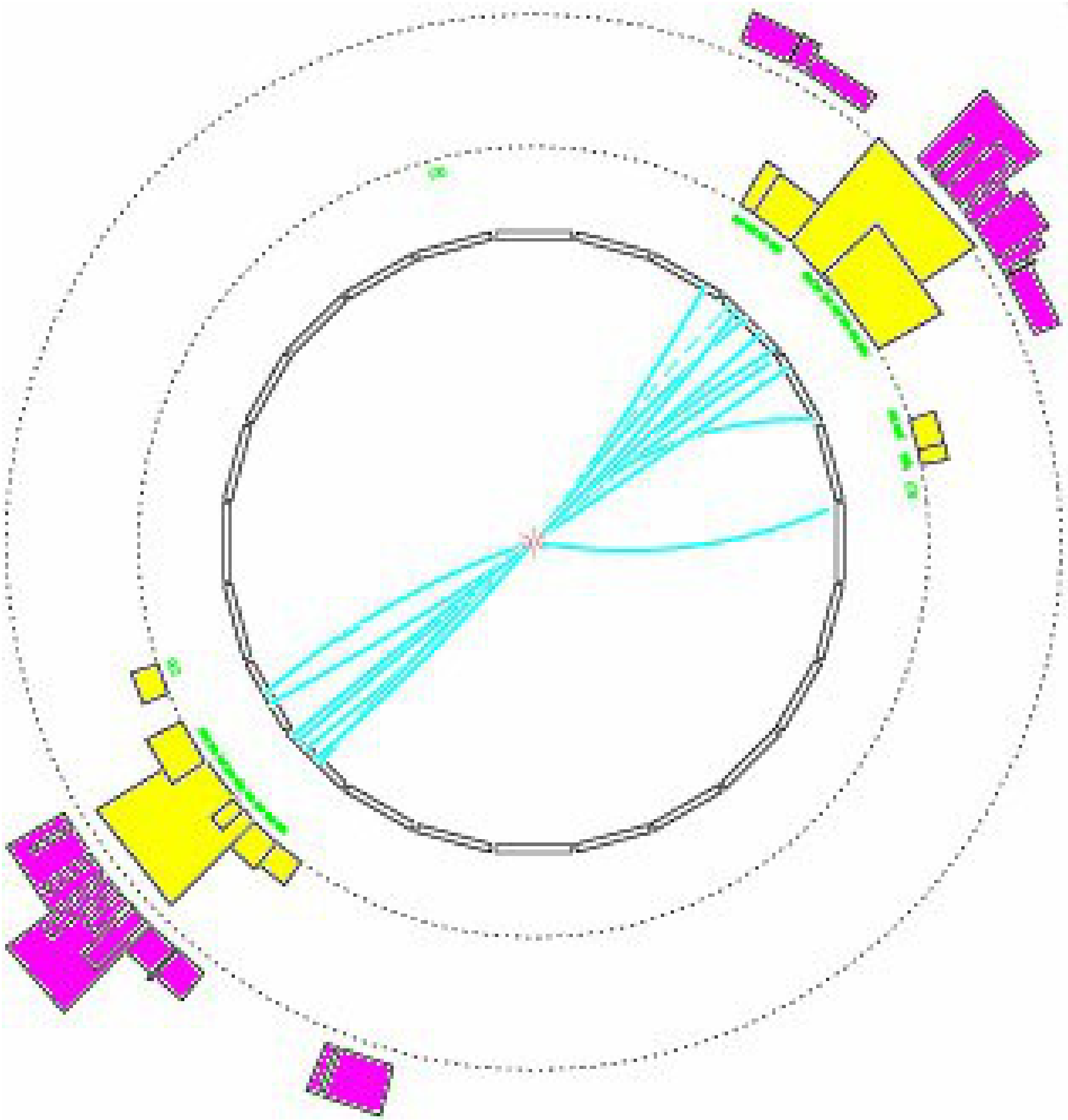
Picturing a QCD event

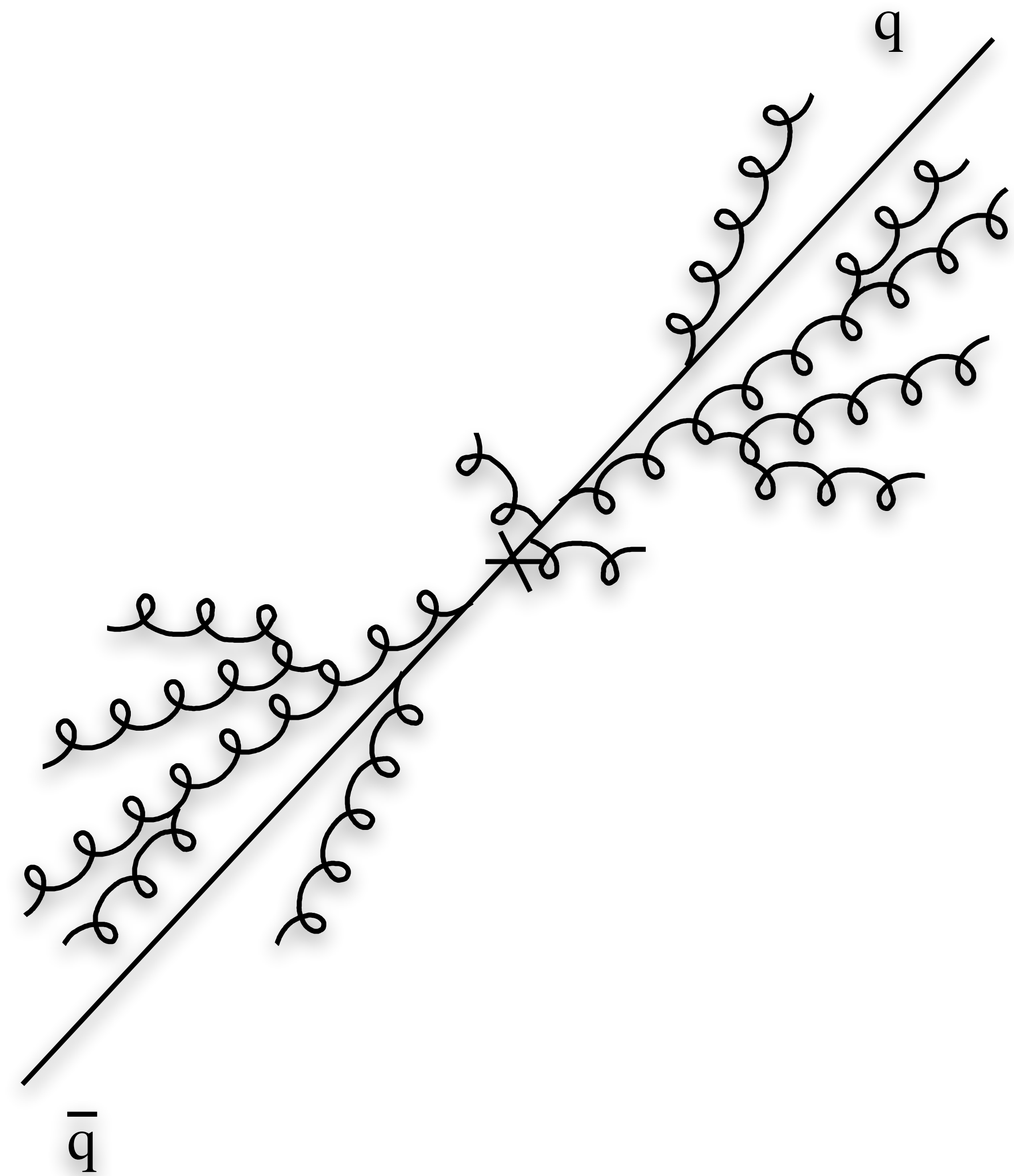




and so forth

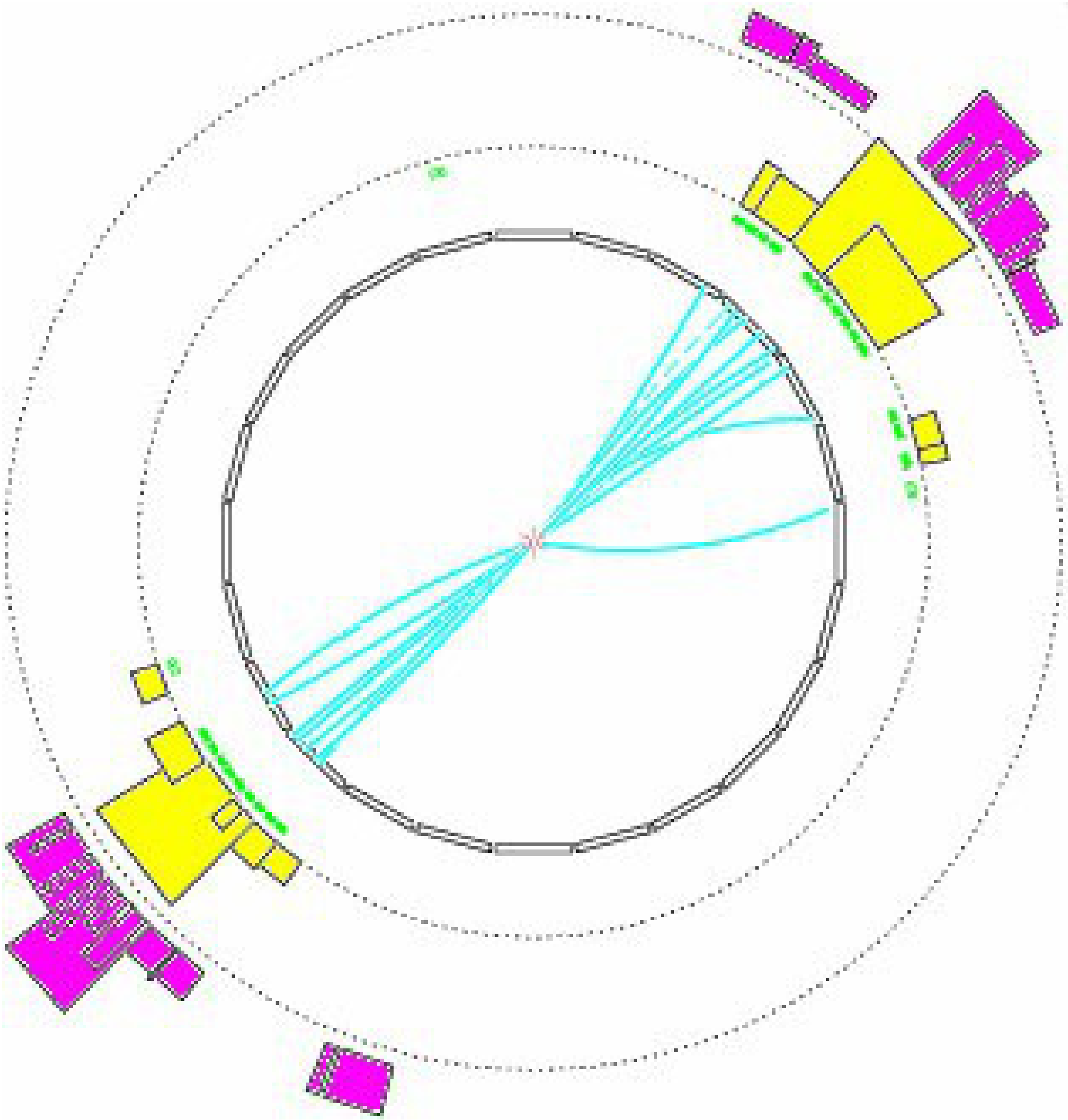
Picturing a QCD event





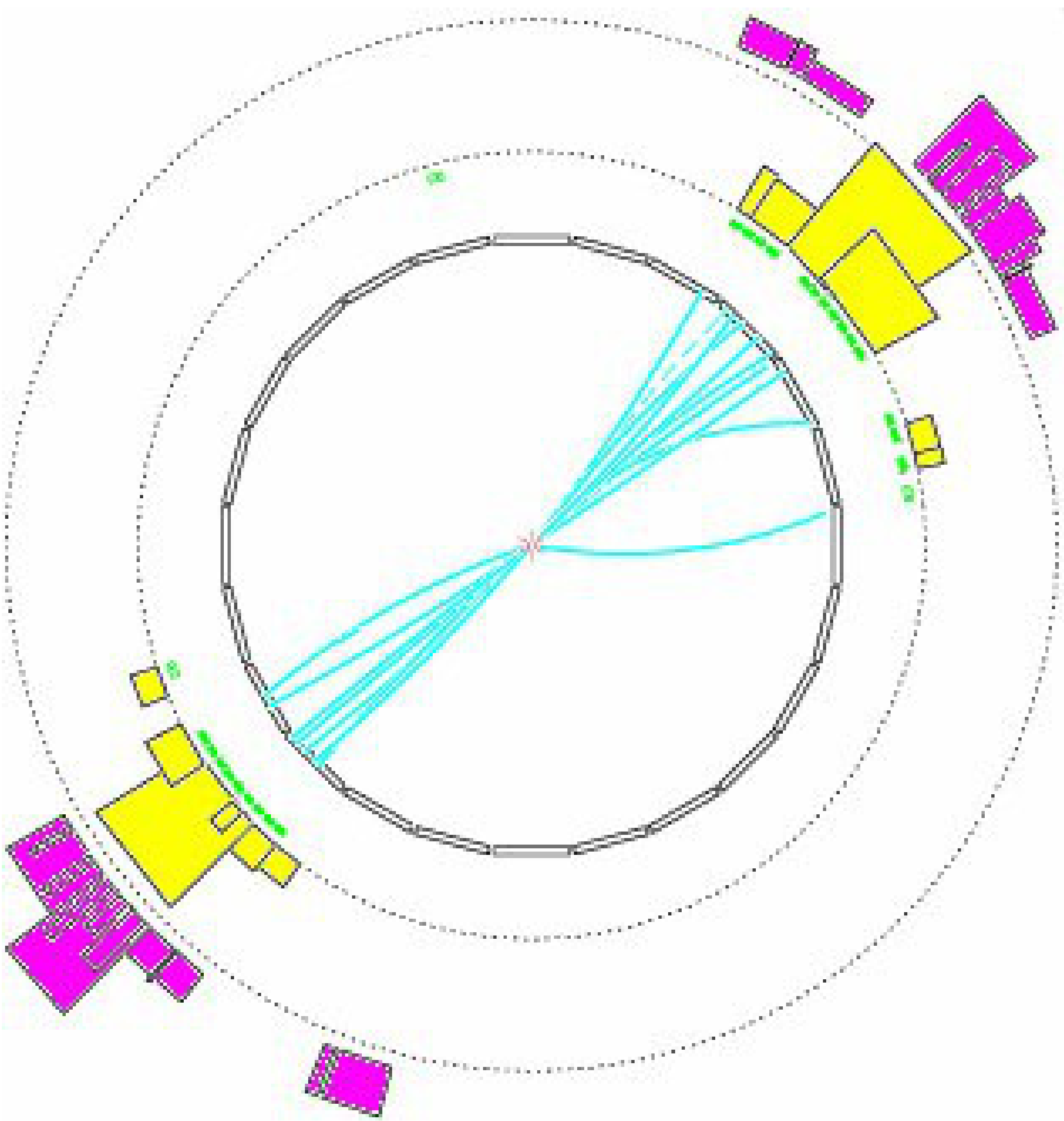
meanwhile the same happened on the other side

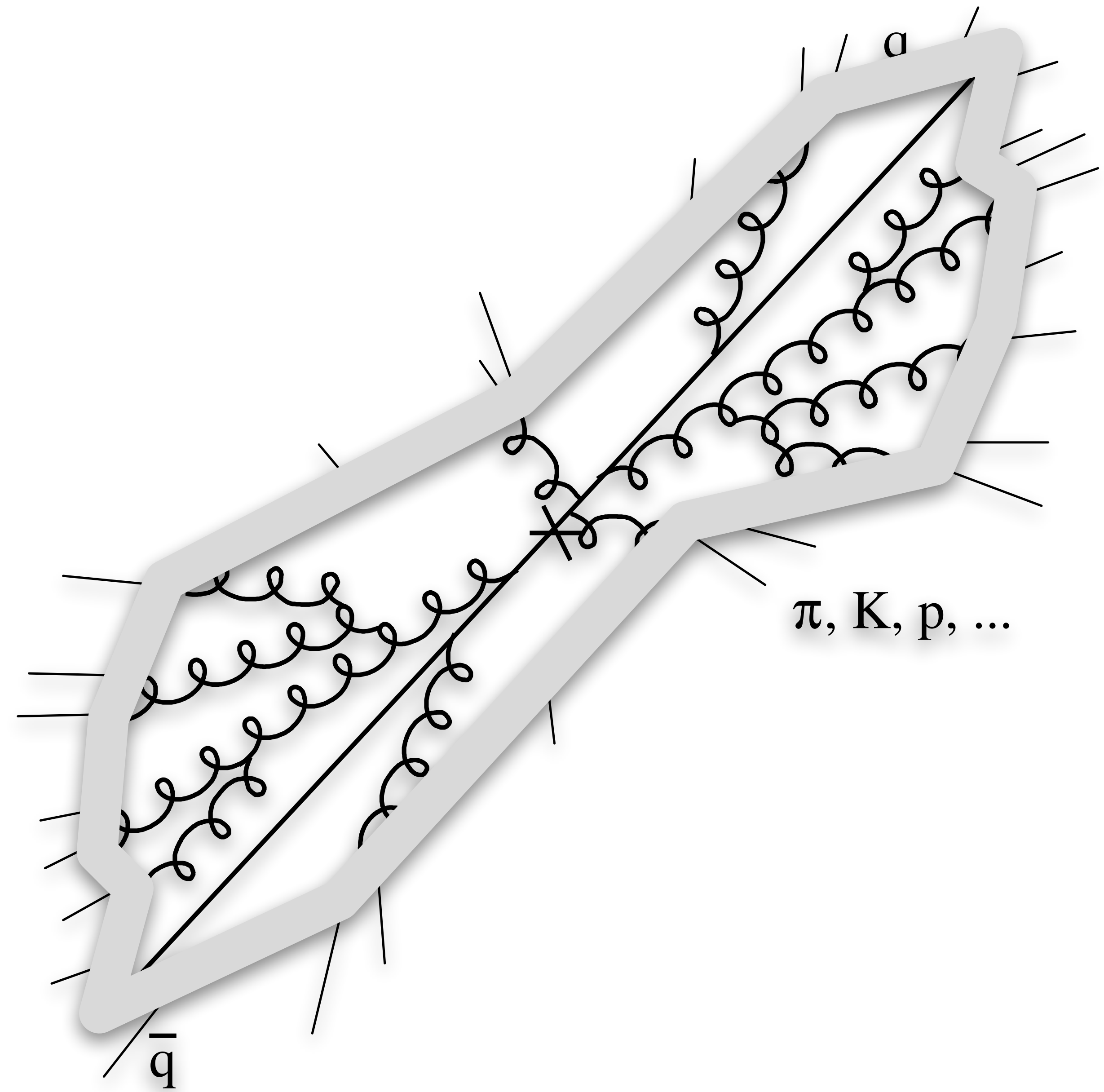
Picturing a QCD event



then a non-perturbative transition occurs

Picturing a QCD event





giving a pattern of hadrons that "remembers" the gluon branching

(hadrons mostly produced at small angles wrt qqbar directions — two **"jets"**)

resummation and parton showers

the previous slides applied in practice

Resummation

Analytical, or semi-numerical, calculation of dominant logarithmically enhanced terms, to all orders in the strong coupling. **Applies when you place a strong constraint on an observable.**

Calculations are often specific to a single observable.

Parton shower Monte Carlo

Simulation of emission of arbitrary number of particles, usually ordered in angle or p_t.

Underlying algorithm should reproduce many of the singular limits of multi-particle QCD amplitudes, including virtual corrections.

Can be used to calculate arbitrary observables.

Resummation: one way of seeing the underlying key idea

Calculate cross section for some **observable v(p₁,...pₘ)**, a function of the event momenta, to be less than some cut V.

Illustrate structure in soft limit, fixed coupling, ignore secondary emissions from soft gluons.

$$\sigma(v(\text{emissions}) < V) =$$
$$\sigma_0 \lim_{\epsilon\to 0} \sum_{m=0}^{\infty} \frac{1}{m!} \prod_{i=1}^{m} \left(\frac{2\alpha_s C_F}{\pi} \int_\epsilon \frac{dE_i}{E_i} \int_\epsilon \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right) \times$$
$$\times \sum_{n=0}^{\infty} \frac{1}{n!} \prod_{i=m+1}^{m+n} \left(-\frac{2\alpha_s C_F}{\pi} \int_\epsilon \frac{dE_i}{E_i} \int_\epsilon \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right) \times$$
$$\times \Theta\left(V - v(p_1, \ldots, p_m)\right)$$

Resummation: one way of seeing the underlying key idea

Calculate cross section for some **observable v(p₁,…pₘ)**, a function of the event momenta, to be less than some cut V.

Illustrate structure in soft limit, fixed coupling, ignore secondary emissions from soft gluons.

Any number of real gluons (independent of each other if angles are all very different)

$$\sigma(v(\text{emissions}) < V) =$$

$$\sigma_0 \lim_{\epsilon\to 0} \boxed{\sum_{m=0}^{\infty} \frac{1}{m!} \prod_{i=1}^{m} \left(\frac{2\alpha_s C_F}{\pi} \int_\epsilon \frac{dE_i}{E_i} \int_\epsilon \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right)} \times$$

$$\times \sum_{n=0}^{\infty} \frac{1}{n!} \prod_{i=m+1}^{m+n} \left(-\frac{2\alpha_s C_F}{\pi} \int_\epsilon \frac{dE_i}{E_i} \int_\epsilon \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right) \times$$

$$\times\, \Theta\left(V - v(p_1, \dots, p_m)\right)$$

Resummation: one way of seeing the underlying key idea

Calculate cross section for some **observable v(p$_1$,...p$_m$)**, a function of the event momenta, to be less than some cut V.

Illustrate structure in soft limit, fixed coupling, ignore secondary emissions from soft gluons.

Any number of real gluons (independent of each other if angles are all very different)

$$\sigma(v(\text{emissions}) < V) =$$

$$\sigma_0 \lim_{\epsilon \to 0} \boxed{\sum_{m=0}^{\infty} \frac{1}{m!} \prod_{i=1}^{m} \left(\frac{2\alpha_s C_F}{\pi} \int_\epsilon \frac{dE_i}{E_i} \int_\epsilon \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right)} \times$$

$$\times \boxed{\sum_{n=0}^{\infty} \frac{1}{n!} \prod_{i=m+1}^{m+n} \left(-\frac{2\alpha_s C_F}{\pi} \int_\epsilon \frac{dE_i}{E_i} \int_\epsilon \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right)} \times$$

$$\times \Theta\left(V - v(p_1, \dots, p_m)\right)$$

Any number of virtual gluons

Resummation: one way of seeing the underlying key idea

Calculate cross section for some **observable v(p$_1$,...p$_m$)**, a function of the event momenta, to be less than some cut V.

Illustrate structure in soft limit, fixed coupling, ignore secondary emissions from soft gluons.

Any number of real gluons (independent of each other if angles are all very different)

$$\sigma(v(\text{emissions}) < V) =$$

$$\sigma_0 \lim_{\epsilon \to 0} \boxed{\sum_{m=0}^{\infty} \frac{1}{m!} \prod_{i=1}^{m} \left(\frac{2\alpha_s C_F}{\pi} \int_\epsilon \frac{dE_i}{E_i} \int_\epsilon \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right)} \times$$

$$\times \boxed{\sum_{n=0}^{\infty} \frac{1}{n!} \prod_{i=m+1}^{m+n} \left(-\frac{2\alpha_s C_F}{\pi} \int_\epsilon \frac{dE_i}{E_i} \int_\epsilon \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right)} \times$$

$$\boxed{\times \, \Theta\left(V - v(p_1, \ldots, p_m)\right)}$$

Any number of virtual gluons

constraint from observable, involves just real gluons

Resumation example result

- It's common to ask questions like *"what is the probability that a Z boson is produced with transverse momentum < p_T"*
- Answer is given (~) by a "**Sudakov form factor**", i.e. the probability of not emitting any gluons with transverse momentum > p_T.

$$P(Z \text{ trans.mom.} < p_T) \simeq \exp\left[-\frac{2\alpha_s C_F}{\pi}\ln^2\frac{M_Z}{p_T}\right]$$

- when p_T is small, the logarithm is large and compensates for the smallness of α_s — so you need to resum log-enhanced terms to all orders in α_s.

What do we know about resummation?

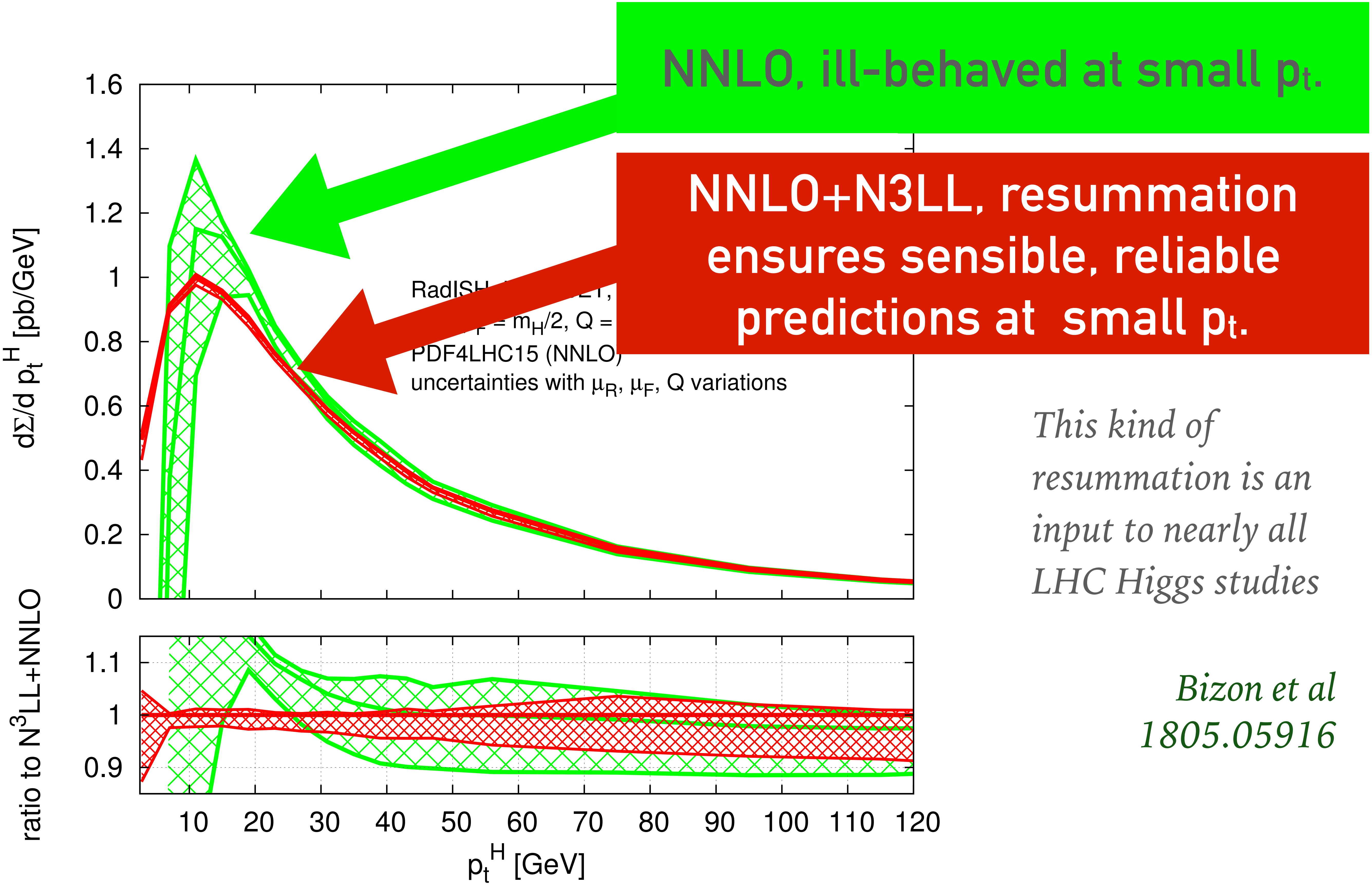
- You'll sometimes see mention of "NNLL" or similar
- This means next-next-to-leading logarithmic
- Most common definition of Leading logarithmic (LL): you sum all terms with p=n+1 (for n=1…∞) in

$$\exp\left[-\sum_{n,p} \alpha_s^n \ln^p \frac{M_H}{p_T}\right]$$

- NLL: include all terms with p=n (for n=1…∞)
- NNLL: include all terms with p=n–1 (for n=1…∞)

In real life, the function that appears in the resummation is sometimes instead a Fourier or Mellin transform of an exponential

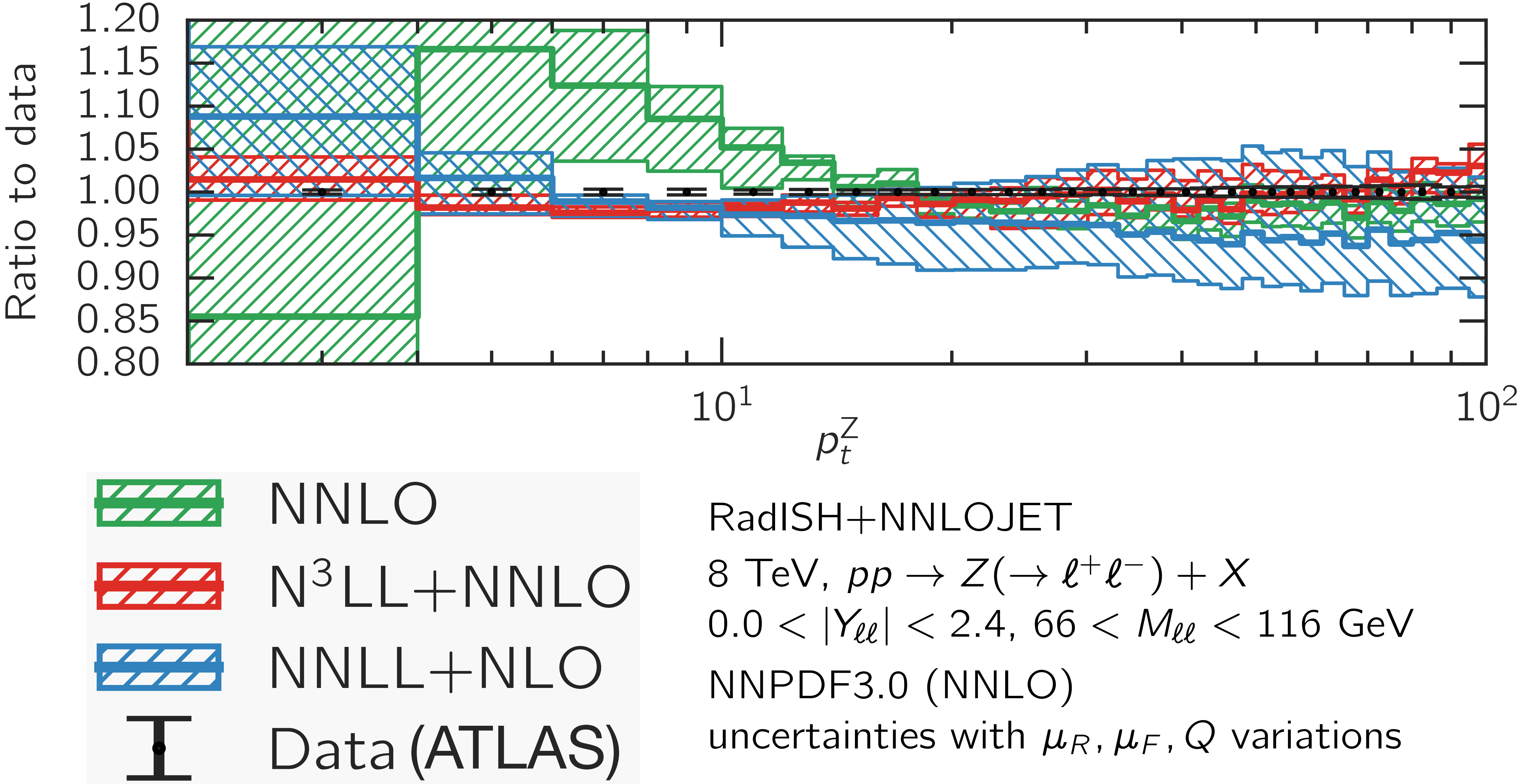
Resummation of Higgs p_T spectrum (same formula, with $C_F \to C_A$)



This kind of resummation is an input to nearly all LHC Higgs studies

Bizon et al 1805.05916

Resummation of Z p_T spectrum v. data



Bizon et al
1805.05916

- If you produce a system near threshold, i.e. mass M close to the pp centre-of-mass energy √s, there are so-called “threshold logarithms”, which modify the total cross section.
- Steeply falling PDFs may also result in threshold logarithms. Driven by a quantity N,

$$N = \frac{d \ln \sigma}{d \ln s}$$

- Resummation involves terms $(\alpha_s \ln^2 N)^n$, often enhance σ
- Valid if $N \gg 1$.
- In practice, for some applications (e.g. top, Higgs production), M « √s and $N \sim 2$. Opinions differ about validity of threshold resummation in this regime

resummation v. parton showers (the basic idea, ignoring secondary emsn. from gluons)

- a resummation predicts **one observable** to high accuracy
- a parton shower takes the same idea of a Sudakov form factor and uses it to generate emissions
- from probability of not emitting gluons above a certain p_T, you can deduce p_T distribution of first emission

1. use a random number generator (r) to sample that p_T distribution

$$\text{deduce } p_T \text{ by solving } r = \exp\left[-\frac{2\alpha_s C_A}{\pi}\ln^2\frac{p_{T,\max}^2}{p_T^2}\right]$$

2. repeat for next emission, etc., until p_T falls below some non-perturbative cutoff

very similar to radioactive decay, with time ~ $1/p_T$ and a decay rate ~ $p_T \log 1/p_T$

A toy shower

https://github.com/gavinsalam/zuoz2016-toy-shower

(fixed coupling, primary branching only, only p_T, no energy conservation, no PDFs, etc.)

```
#!/usr/bin/env python
# an oversimplified (QED-like) parton shower
# for Zuoz lectures (2016) by Gavin P. Salam
from random import random
from math import pi, exp, log, sqrt

ptHigh = 100.0
ptCut  = 1.0
alphas = 0.12
CA=3

def main():
    for iev in range(0,10):
        print "\nEvent", iev
        event()

def event():
    # start with maximum possible value of Sudakov
    sudakov  = 1
    while (True):
        # scale it by a random number
        sudakov *= random()
        # deduce the corresponding pt
        pt = ptFromSudakov(sudakov)
        # if pt falls below the cutoff, event is finished
        if (pt < ptCut): break
        print "  primary emission with pt = ", pt

def ptFromSudakov(sudakovValue):
    """Returns the pt value that solves the relation
       Sudakov = sudakovValue (for 0 < sudakovValue < 1)
    """
    norm = (2*CA/pi)
    # r = Sudakov = exp(-alphas * norm * L^2)
    # --> log(r) = -alphas * norm * L^2
    # --> L^2 = log(r)/(-alphas*norm)
    L2 = log(sudakovValue)/(-alphas * norm)
    pt = ptHigh * exp(-sqrt(L2))
    return pt

main()
```

A toy shower

https://github.com/gavinsalam/zuoz2016-toy-shower

(fixed coupling, primary branching only, only p_T, no energy conservation, no PDFs, etc.)

```
#!/usr/bin/env python
# an oversimplified (QED-like) parton shower
# for Zuoz lectures (2016) by Gavin P. Salam
from random import random
from math import pi, exp, log, sqrt

ptHigh = 100.0
ptCut  = 1.0
alphas = 0.12
CA=3

def main():
    for iev in range(0,10):
        print "\nEvent", iev
        event()

def event():
    # start with maximum possible value of Sudakov
    sudakov  = 1
    while (True):
        # scale it by a random number
        sudakov *= random()
        # deduce the corresponding pt
        pt = ptFromSudakov(sudakov)
        # if pt falls below the cutoff, event is finished
        if (pt < ptCut): break
        print "  primary emission with pt = ", pt

def ptFromSudakov(sudakovValue):
    """Returns the pt value that solves the relation
       Sudakov = sudakovValue (for 0 < sudakovValue < 1)
    """
    norm = (2*CA/pi)
    # r = Sudakov = exp(-alphas * norm * L^2)
    # --> log(r) = -alphas * norm * L^2
    # --> L^2 = log(r)/(-alphas*norm)
    L2 = log(sudakovValue)/(-alphas * norm)
    pt = ptHigh * exp(-sqrt(L2))
    return pt

main()
```

```
% python ./toy-shower.py

Event 0
  primary emission with pt =  58.4041962726
  primary emission with pt =  3.61999582015
  primary emission with pt =  2.31198814996

Event 1
  primary emission with pt =  32.1881228375
  primary emission with pt =  10.1818306204
  primary emission with pt =  10.1383134201
  primary emission with pt =  7.24482350383
  primary emission with pt =  2.35709074796
  primary emission with pt =  1.0829758034

Event 2
  primary emission with pt =  64.934992001
  primary emission with pt =  16.4122436094
  primary emission with pt =  2.53473253194

Event 3
  primary emission with pt =  37.6281171491
  primary emission with pt =  22.7262873764
  primary emission with pt =  12.0255817868
  primary emission with pt =  4.73678636215
  primary emission with pt =  3.92257832288

Event 4
  primary emission with pt =  21.5359449851
  primary emission with pt =  4.01438733798
  primary emission with pt =  3.33902663941
  primary emission with pt =  2.02771620824
  primary emission with pt =  1.05944759028

. . .
```

A toy shower

https://github.com/gavinsalam/zuoz2016-toy-shower

(fixed coupling, primary branching only, only p_T, no energy conservation, no PDFs, etc.)

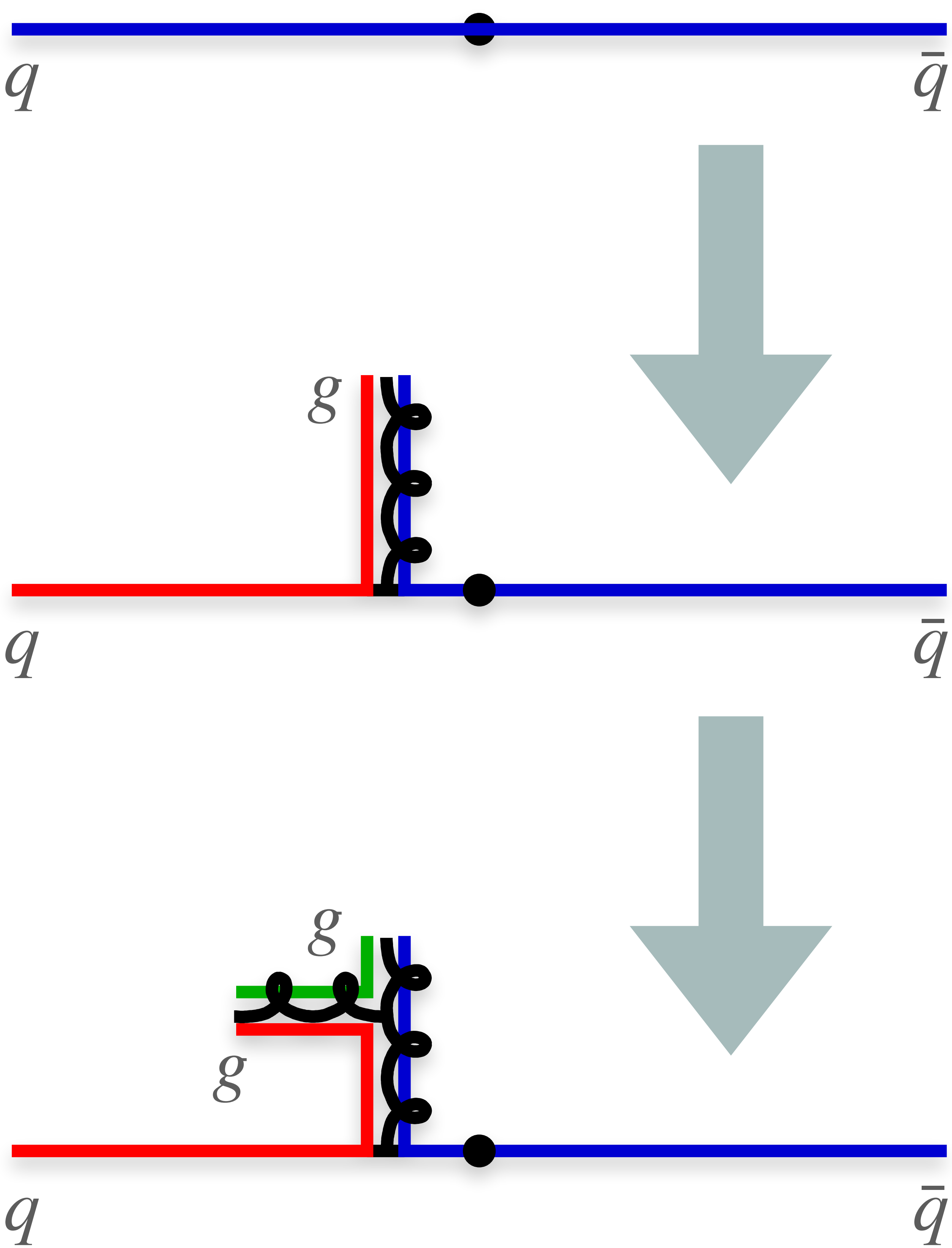
```python
#!/usr/bin/env python
# an oversimplified (QED-like) parton shower
# for Zuoz lectures (2016) by Gavin P. Salam
from random import random
from math import pi, exp, log, sqrt

ptHigh = 100.0
ptCut  = 1.0
alphas = 0.12
CA=3

def main():
    for iev in range(0,10):
        print "\nEvent", iev
        event()

def event():
    # start with maximum possible value of Sudakov
    sudakov  = 1
    while (True):
        # scale it by a random number
        sudakov *= random()
        # deduce the corresponding pt
        pt = ptFromSudakov(sudakov)
        # if pt falls below the cutoff, event is finished
        if (pt < ptCut): break
        print "  primary emission with pt = ", pt

def ptFromSudakov(sudakovValue):
    """Returns the pt value that solves the relation
       Sudakov = sudakovValue (for 0 < sudakovValue < 1)
    """
    norm = (2*CA/pi)
    # r = Sudakov = exp(-alphas * norm * L^2)
    # --> log(r) = -alphas * norm * L^2
    # --> L^2 = log(r)/(-alphas*norm)
    L2 = log(sudakovValue)/(-alphas * norm)
    pt = ptHigh * exp(-sqrt(L2))
    return pt

main()
```

```
% python ./toy-shower.py

Event 0
  primary emission with pt =  58.4041962726
  primary emission with pt =  3.61999582015
  primary emission with pt =  2.31198814996

Event 1
  primary emission with pt =  32.1881228375
  primary emission with pt =  10.1818306204
  primary emission with pt =  10.1383134201
  primary emission with pt =  7.24482350383
  primary emission with pt =  2.35709074796
  primary emission with pt =  1.0829758034

Event 2
  primary emission with pt =  64.934992001
  primary emission with pt =  16.4122436094
  primary emission with pt =  2.53473253194
```

If you want to play: replace C_A=3 (emission from gluons) with C_F=4/3 (emission from quarks) and see how pattern of emissions changes (multiplicity, p_T of hardest emission, etc.)

Secondary, tertiary gluons: many showers use colour dipoles (Pythia, Sherpa & option in Herwig)

Original dipole MC: Ariadne (90's)



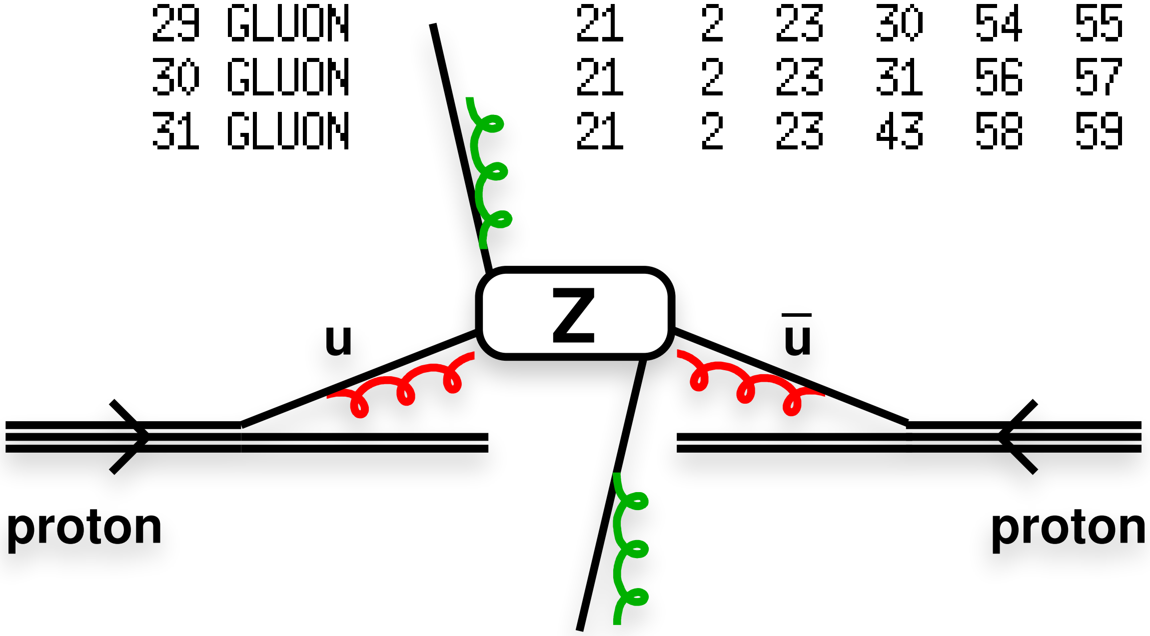
- Use large-N_C idea of colour structure
- Initial $q\bar{q}$ event = 1 colour dipole.
- Radiated gluon turns 1 dipole → 2 dipoles
- Each dipole then radiates independently (different colour ≡ no interference), creating new colour dipoles at each step

Event record from a real-world shower (Herwig6 — old shower with compact record)

IHEP	ID	IDPDG	IST	MO1	MO2	DA1	DA2	P-X	P-Y	P-Z	ENERGY	MASS
9	UQRK	94	141	4	6	11	16	2.64	-9.83	592.2	590.2	-49.07
10	CONE	0	100	4	5	0	0	-0.27	0.96	0.1	1.0	0.00
11	GLUON	21	2	9	12	32	33	-1.02	3.59	5.6	6.7	0.75-
12	GLUON	21	2	9	13	34	35	0.25	1.46	3.6	4.0	0.75-
13	GLUON	21	2	9	14	36	37	-0.87	1.62	4.7	5.1	0.75-
14	GLUON	21	2	9	15	38	39	-0.81	4.17	3611.7	3611.7	0.75-
15	GLUON	21	2	9	16	40	41	-0.19	-1.01	1727.7	1727.7	0.75-
16	UD	2101	2	9	25	42	41	0.00	0.00	1054.6	1054.6	0.32-
17	GLUON	94	142	5	6	19	21	-2.23	0.44	-233.5	232.8	-18.36
18	CONE	0	100	5	8	0	0	0.77	0.64	0.2	1.0	0.00
19	GLUON	21	2	17	20	43	44	1.60	0.58	-2.1	2.8	0.75
20	UD	2101	2	17	21	45	44	0.00	0.00	-2687.6	2687.6	0.32
21	UQRK	2	2	17	32	46	45	0.63	-1.02	-4076.9	4076.9	0.32
22	Z0/GAMA*	23	195	7	22	251	252	-257.66	-219.68	324.8	477.5	88.56
23	UQRK	94	144	8	6	25	31	258.06	210.29	33.9	345.5	86.10
24	CONE	0	100	8	5	0	0	0.21	0.17	-1.0	1.0	0.00
25	UQRK	2	2	23	26	47	42	26.82	24.33	23.7	43.3	0.32
26	GLUON	21	2	23	27	48	49	8.50	8.18	6.0	13.3	0.75
27	GLUON	21	2	23	28	50	51	73.27	61.24	12.0	96.2	0.75
28	GLUON	21	2	23	29	52	53	73.66	58.54	-6.3	94.3	0.75
29	GLUON	21	2	23	30	54	55	67.58	52.13	-7.3	85.7	0.75
30	GLUON	21	2	23	31	56	57	6.98	4.60	2.3	8.7	0.75
31	GLUON	21	2	23	43	58	59	1.24	1.26	3.6	4.1	0.75

INITIAL STATE SHOWER

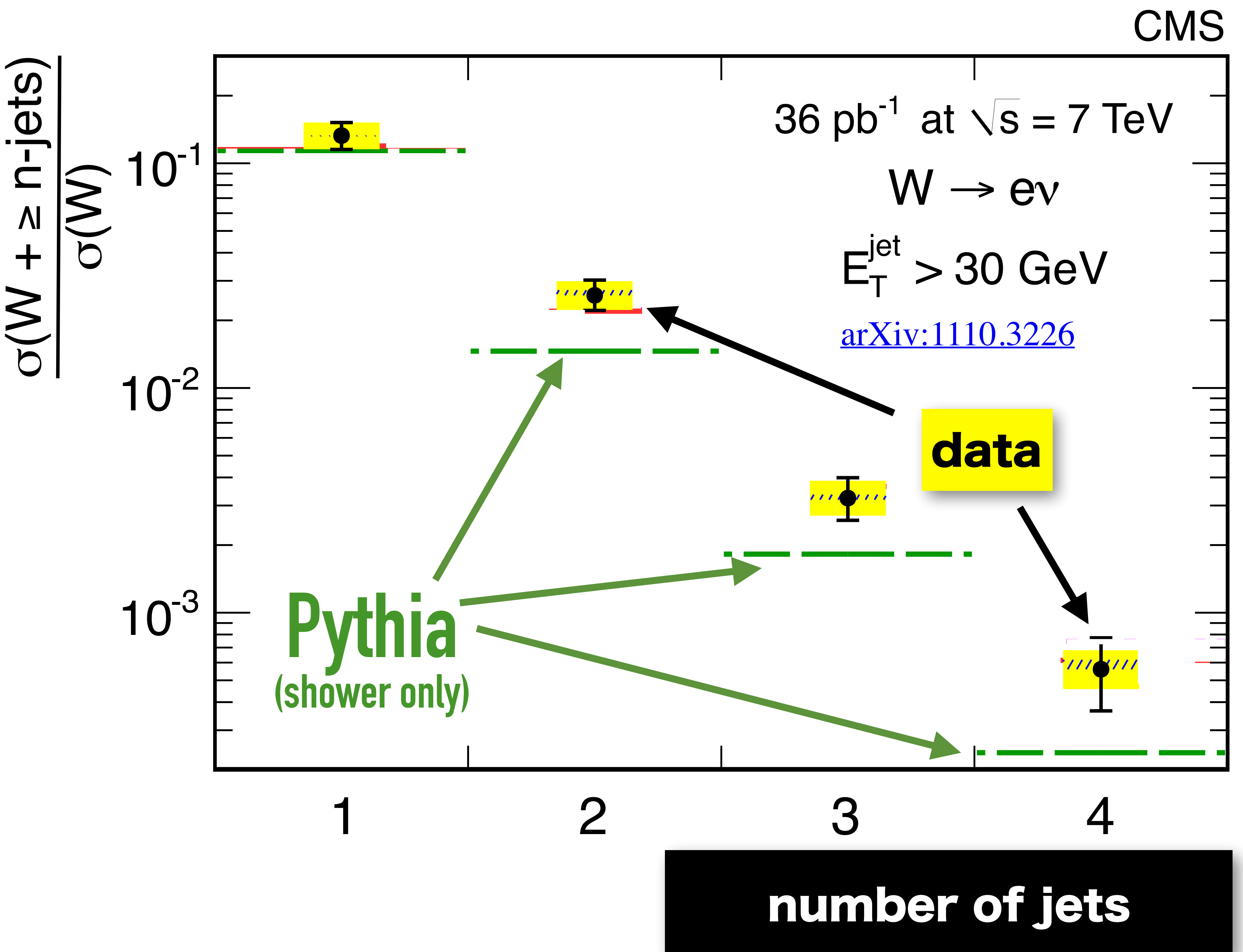
FINAL STATE SHOWER



combining showers & fixed order

essential for accurate cross sections & multijet states

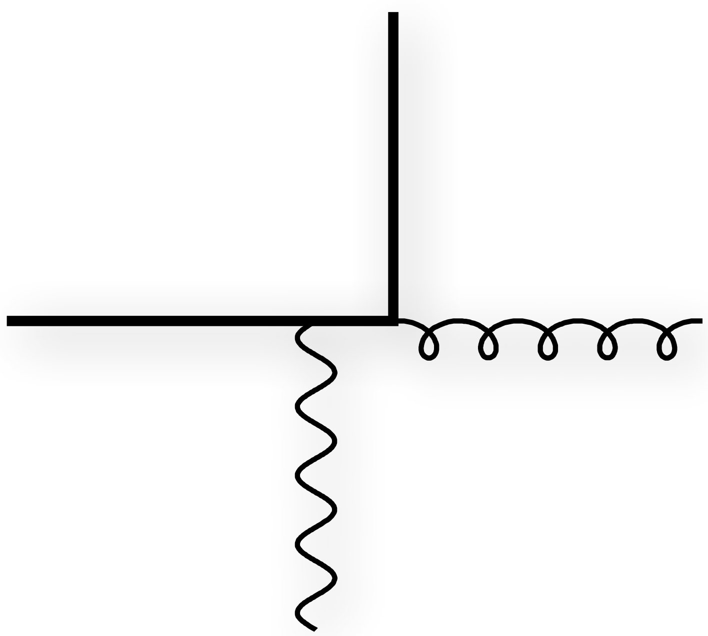
E.g. jet multiplicity in events with a W v. Pythia



shower MCs on their own cannot reproduce pattern of hard multijet states

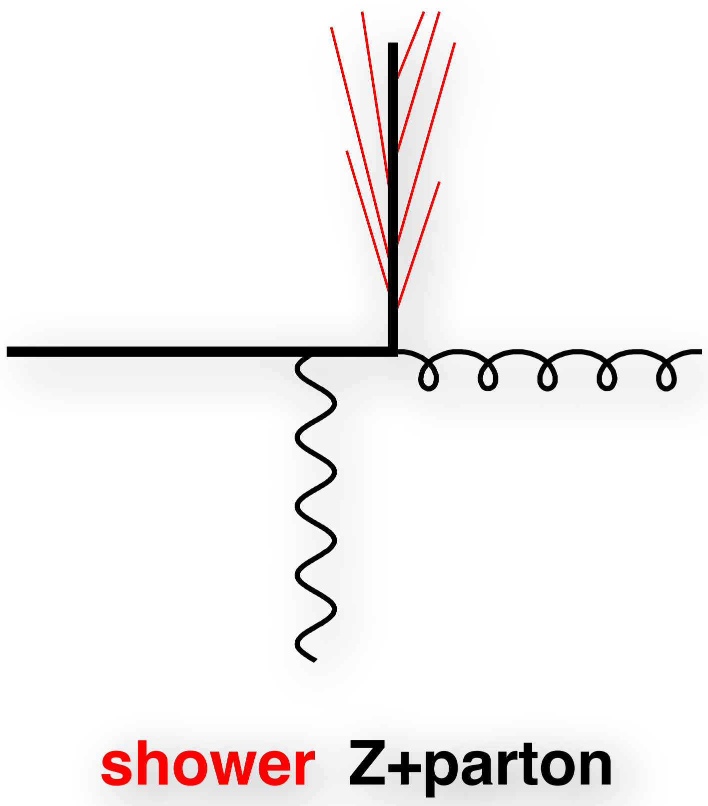
(there are topologies that are almost inaccessible via showering)

MLM matching

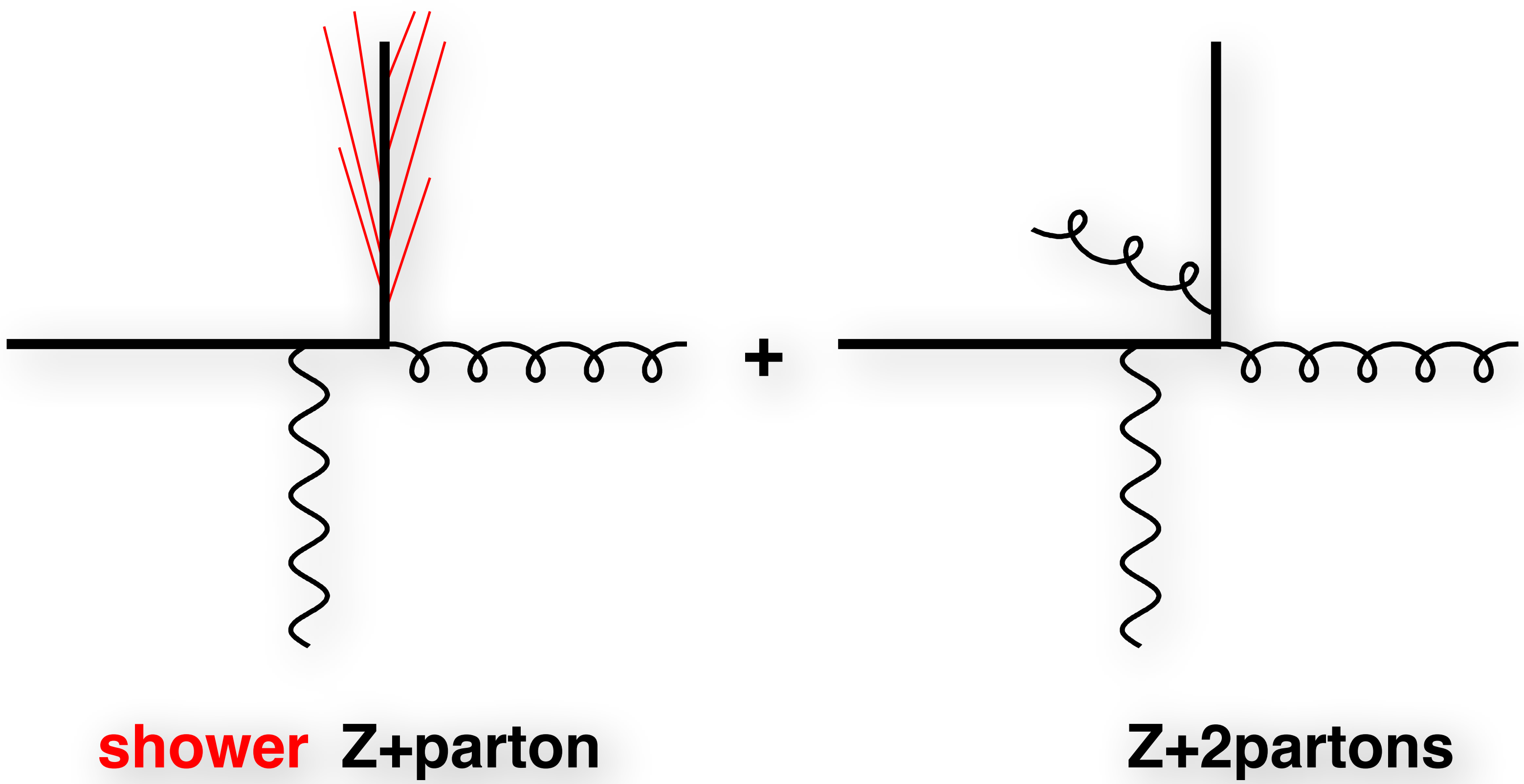


Z+parton

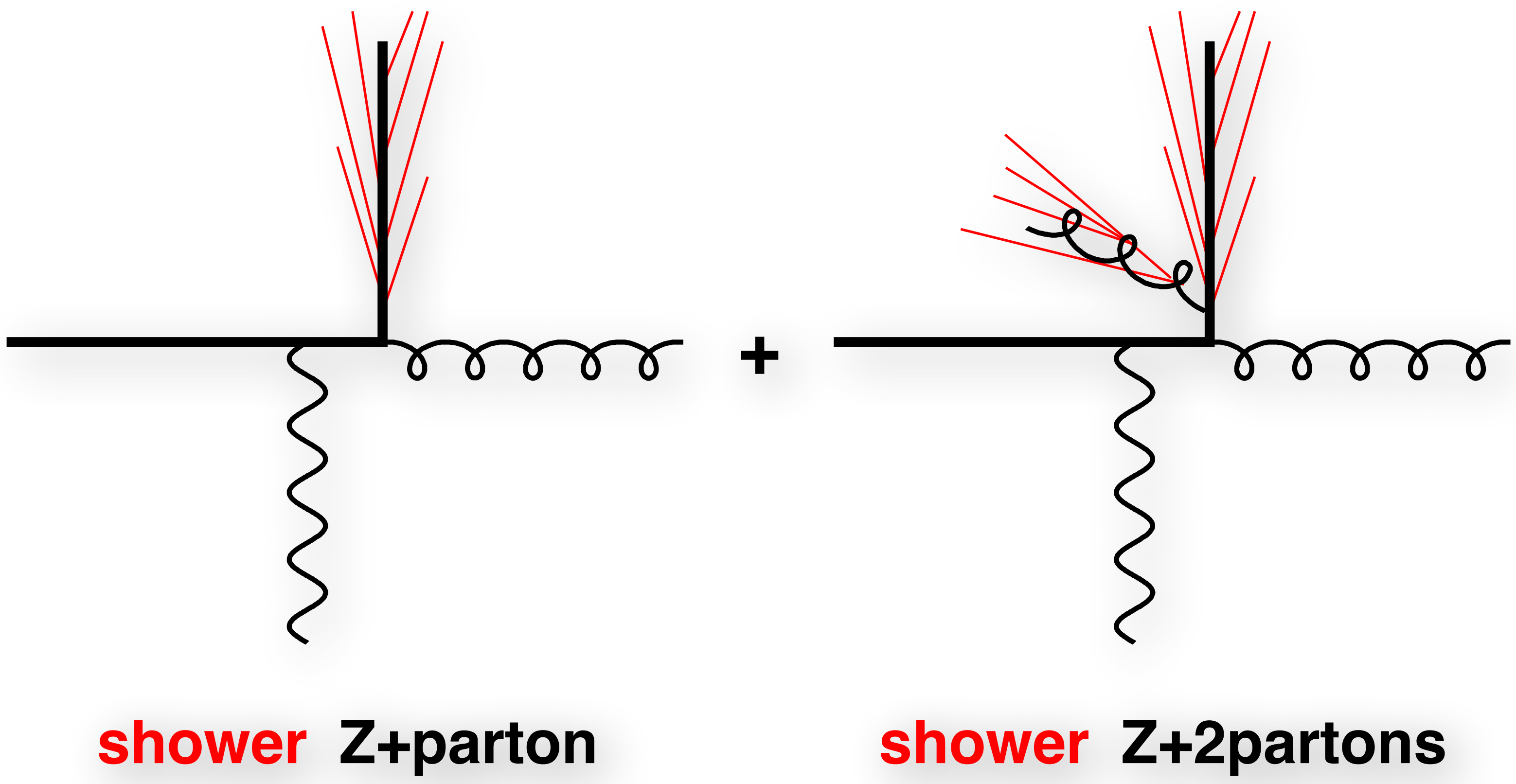
MLM matching



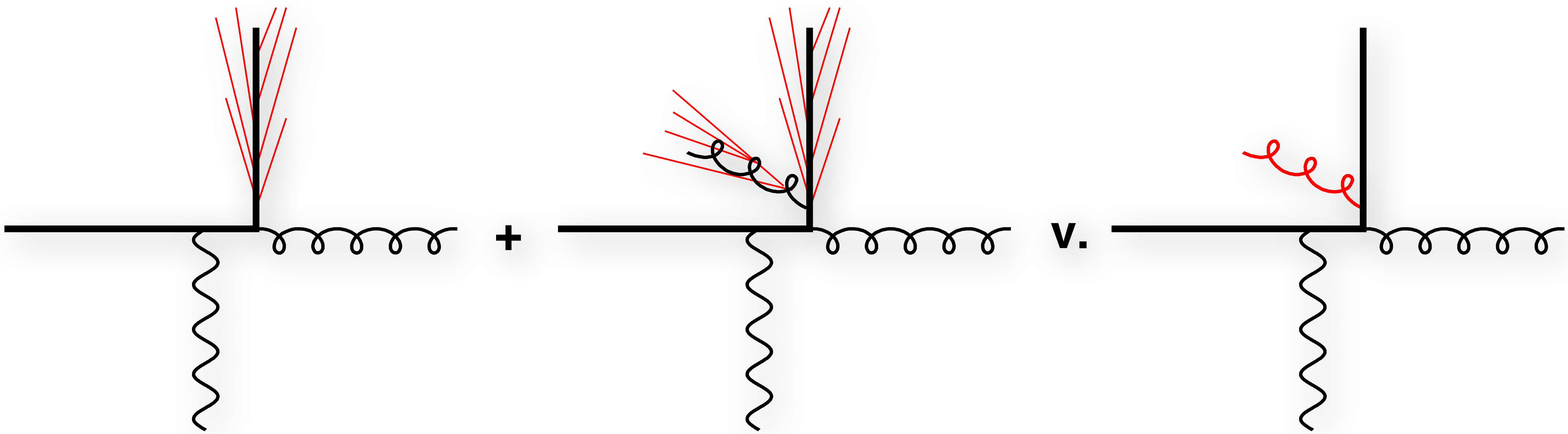
MLM matching



MLM matching



MLM matching

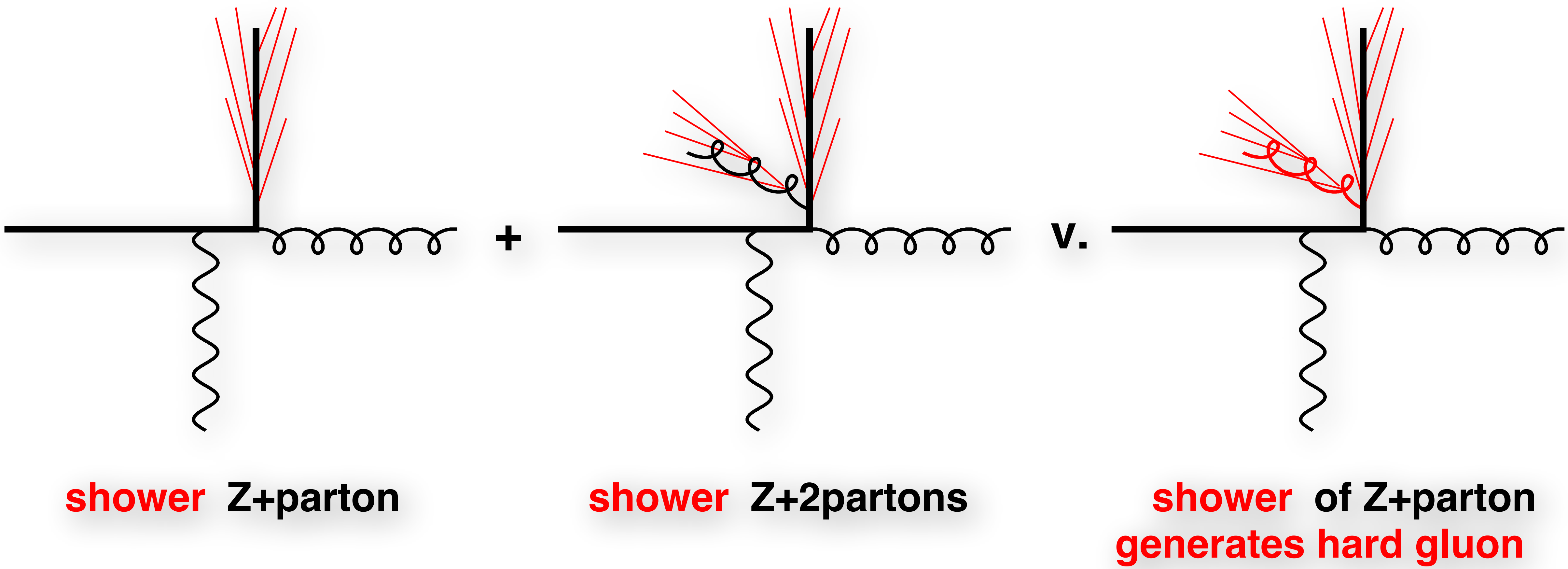


shower Z+parton

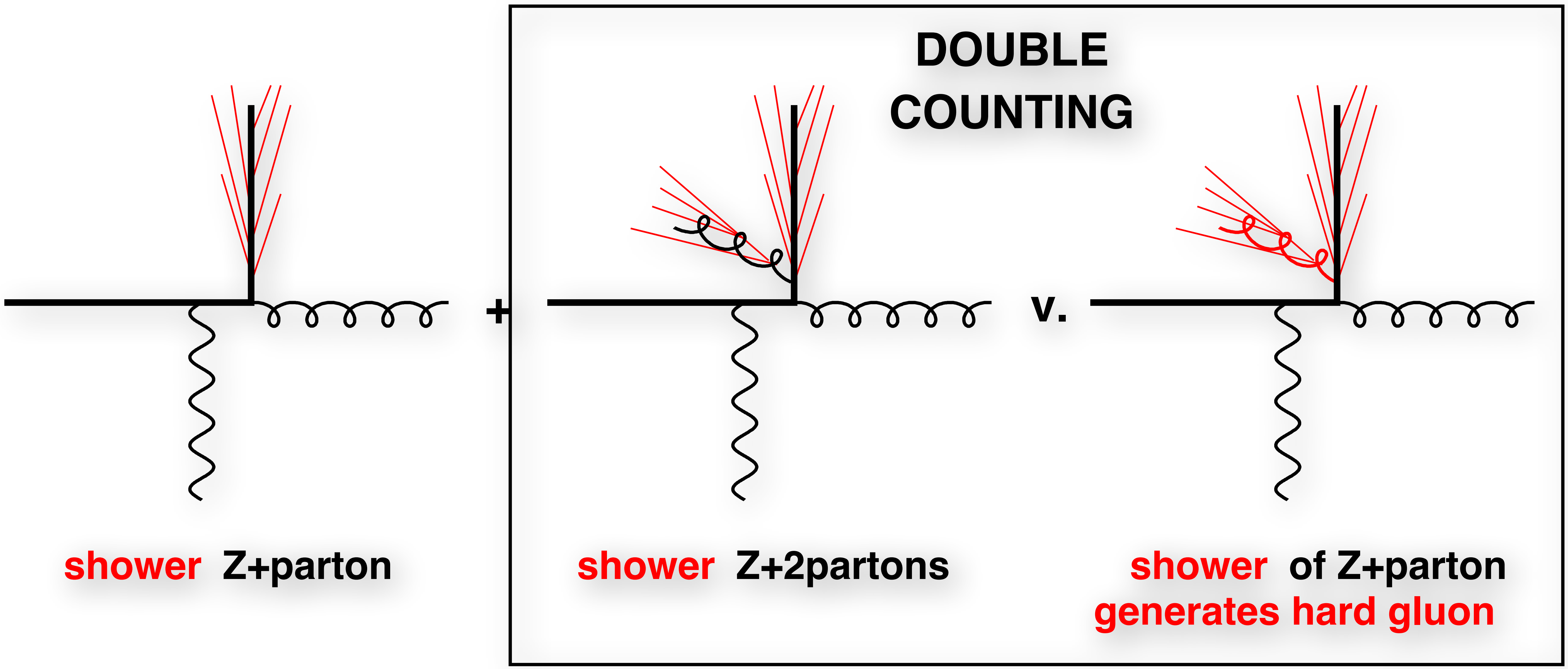
shower Z+2partons

shower of Z+parton
generates hard gluon

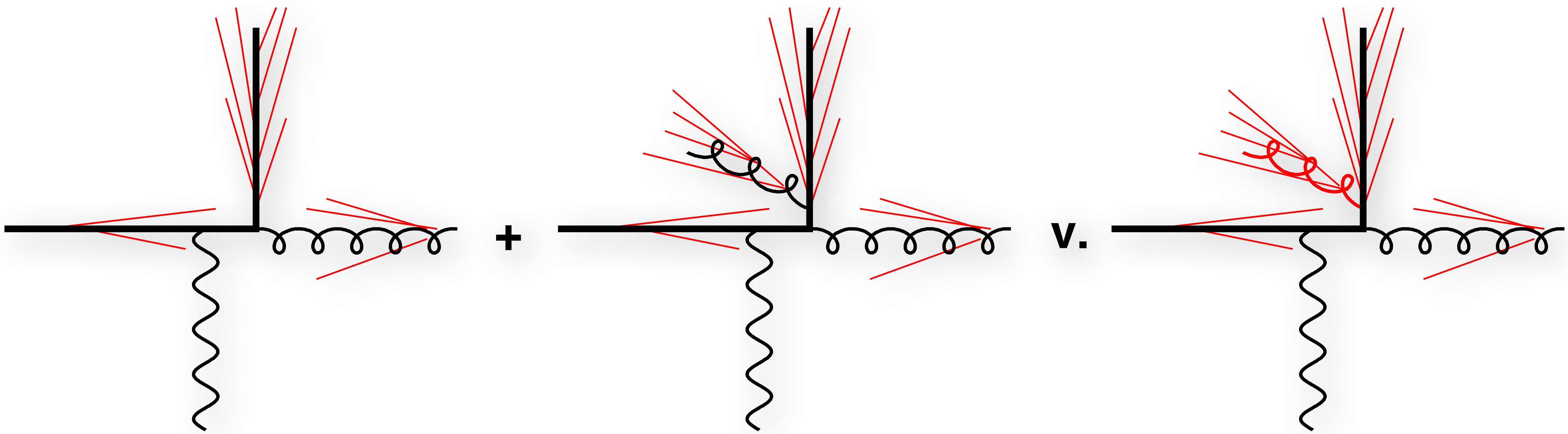
MLM matching



MLM matching



MLM matching

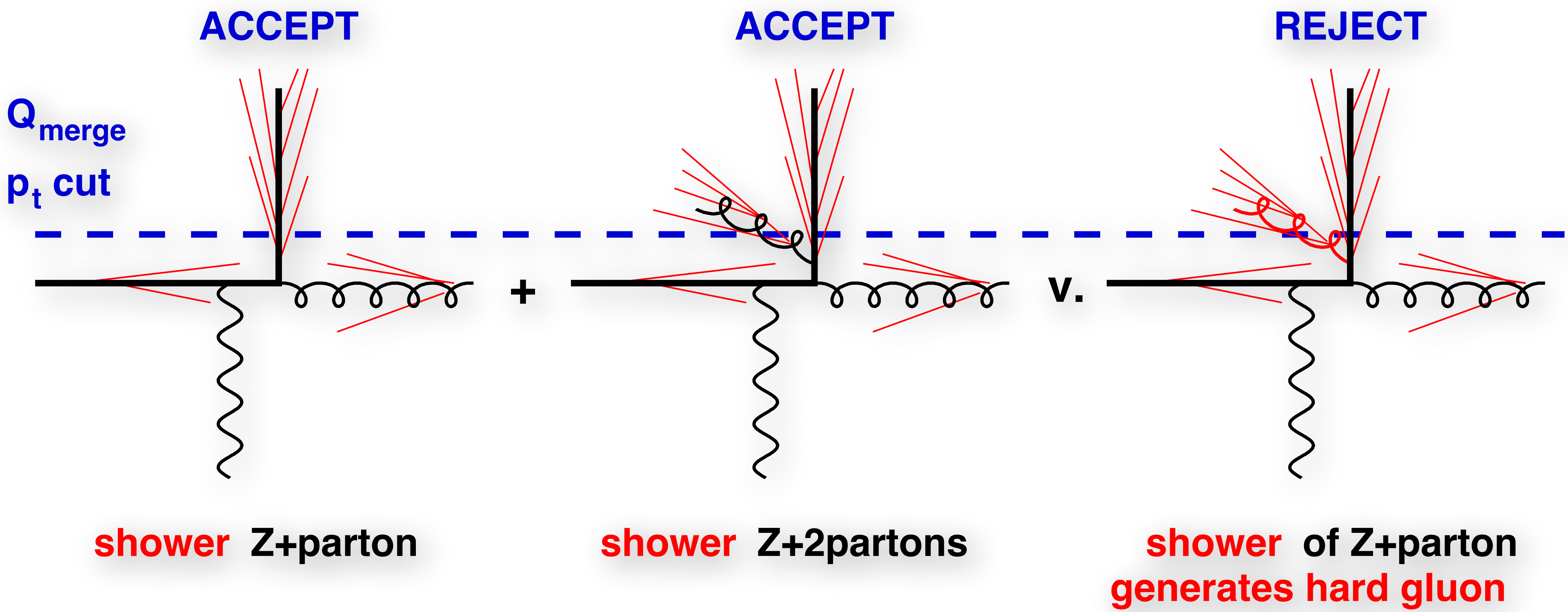


shower Z+parton

shower Z+2partons

shower of Z+parton
generates hard gluon

MLM matching



- ➤ Hard jets above scale Qmerge have distributions given by tree-level ME
- ➤ Rejection procedure eliminates "double-counted" jets from parton shower
- ➤ Rejection generates Sudakov form factors between individual jet scales

An alternative approach is called ***CKKW*** *(similar in spirit, Sudakov put in manually)*

Combining NLO accuracy with parton showers (1)

MC@NLO ideas

Frixione & Webber '02

- Expand your Monte Carlo branching to first order in α_s

  Rather non-trivial – requires deep understanding of MC

- Calculate differences wrt true $\mathcal{O}(\alpha_s)$

  both in real and virtual pieces

- If your Monte Carlo gives correct soft and/or collinear limits, those differences are **finite**

- Generate extra partonic configurations with phase-space distributions proportional to those differences and shower them

$$\text{MC@NLO} = \text{MC} \times \left(1 + \alpha_s(\sigma_{1V} - \sigma_{1V}^{MC}) + \alpha_s \int dE(\sigma_{1R}(E) - \sigma_{1R}^{MC}(E))\right)$$

All weights finite, but can be ± 1

almost any process can be generated automatically in MadGraph5_aMCatNLO (+ Pythia); also in Sherpa & Herwig

Combining NLO accuracy with parton showers (2)

POWHEG ideas

Aims to work around MC@NLO limitations — Nason '04

- the (small fraction of) negative weights
- the tight interconnection with a specific MC

Principle

- Write a simplified Monte Carlo that generates **just one emission** (the hardest one) which alone gives the correct NLO result.

  Essentially uses special Sudakov

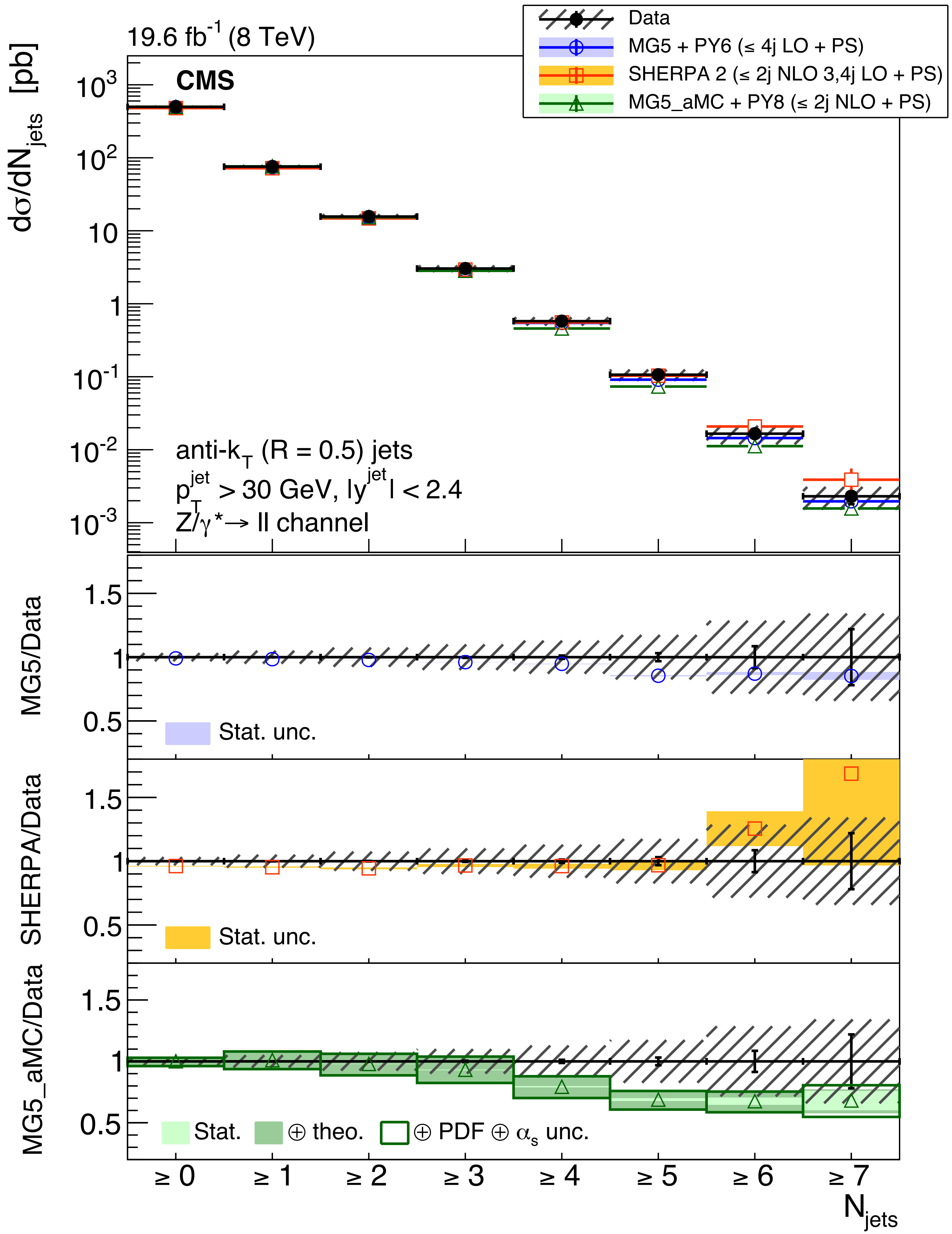
  $$\Delta(k_t) = \exp(-\int \text{exact real-radition probability above } k_t)$$

- Lets your default parton-shower do branchings below that k_t.

most processes available in the POWHEGBox (+Pythia or Herwig; or natively in Herwig)

Recent developments and research directions

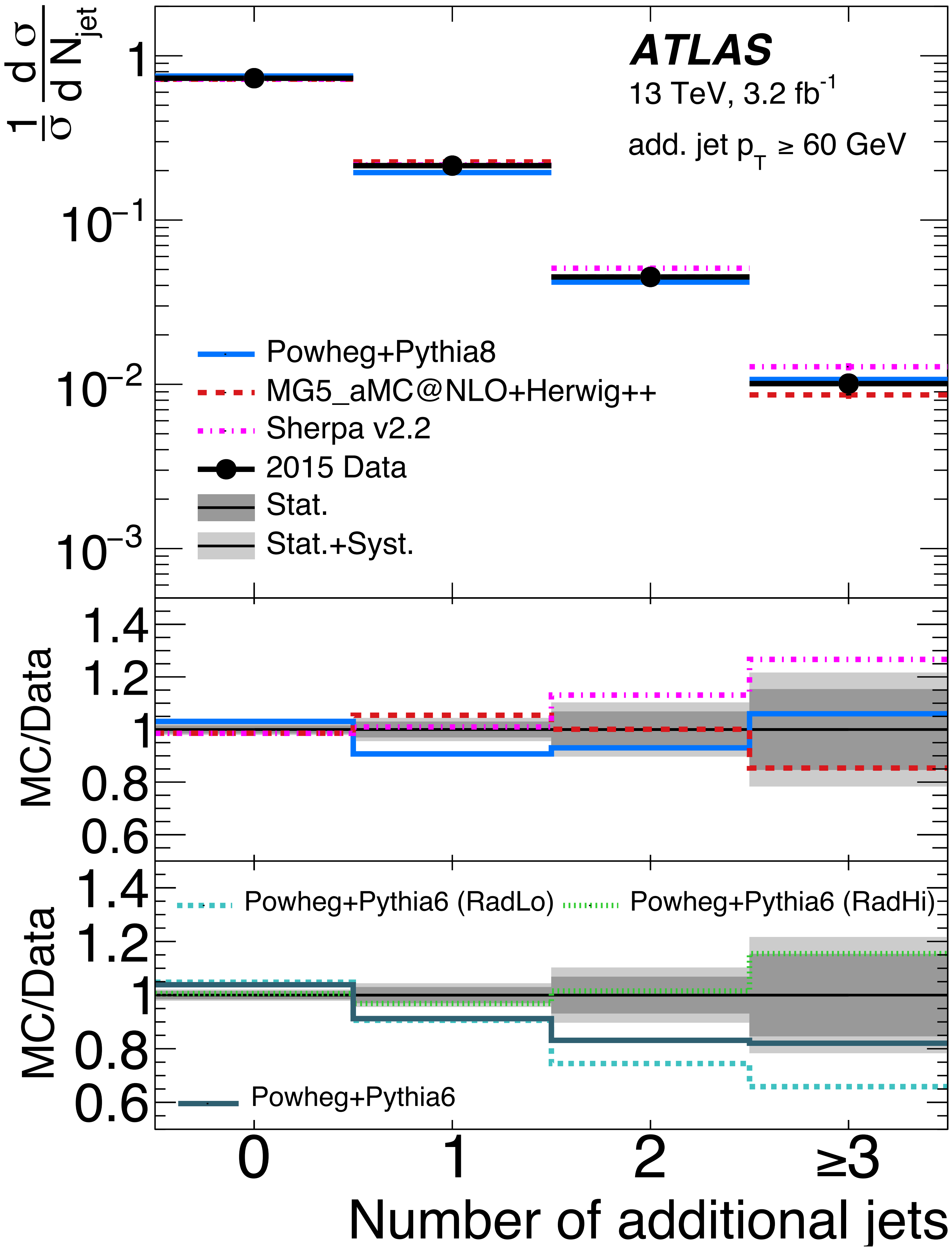
- (Much) more efficient ways of combining tree-level and showers: Vincia
- Getting shower samples that are simultaneously NLO accurate at different multiplicities (FxFx, Sherpa NLO matching)
- Showers that are NNLO accurate: MiNLO', Geneva, UNNLOPS
- Understanding the underlying resummation accuracy of showers (going beyond LL, leading colour) — and its interplay with merging / matching



- Modern tools give good predictions for multijet rates **with vector bosons**
- (up to ~ 4 jets, sometimes beyond)

not in handout

arXiv:1610.09978



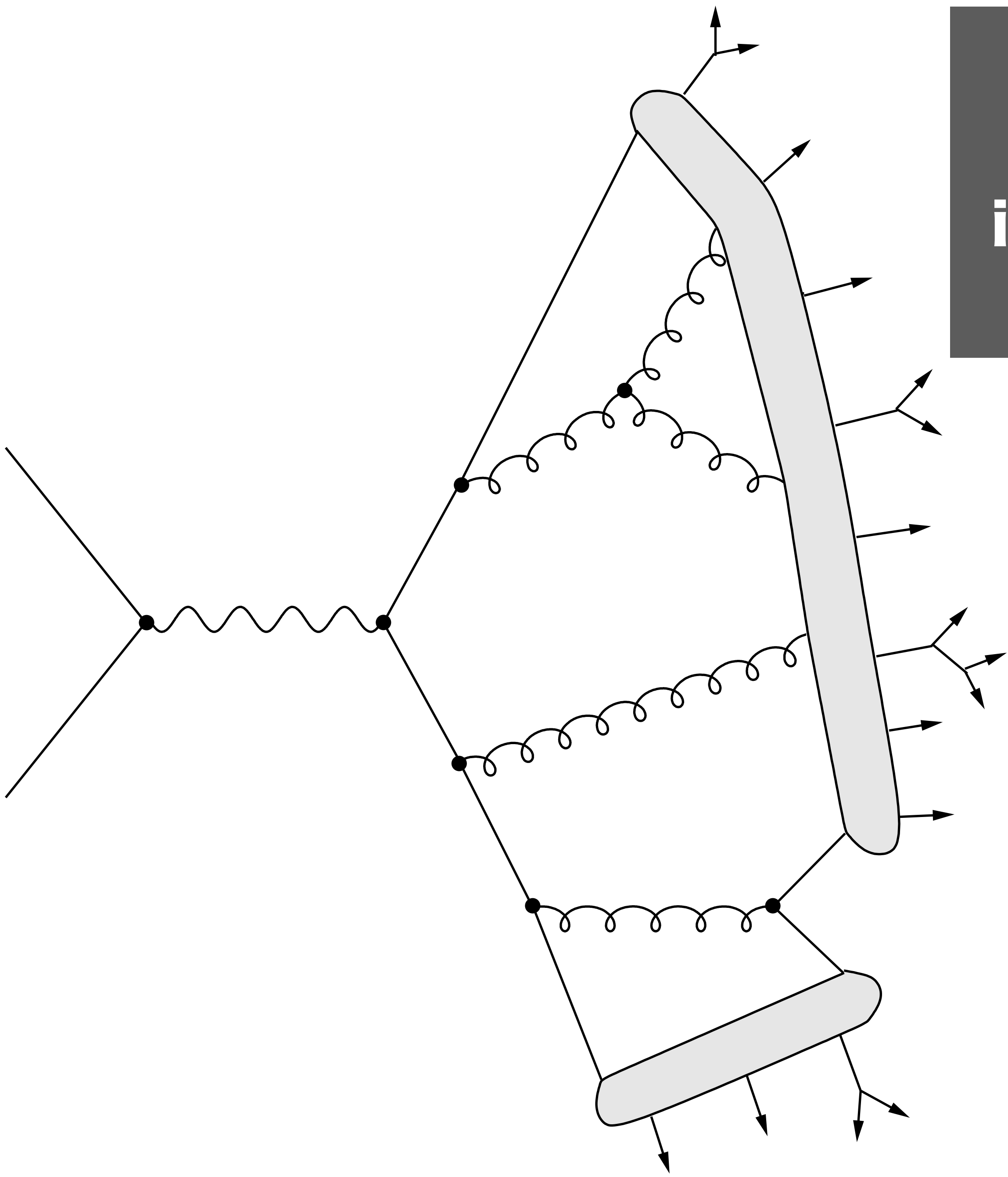
- Modern tools give good predictions for multijet rates **with top quarks**

hadronisation & MPI

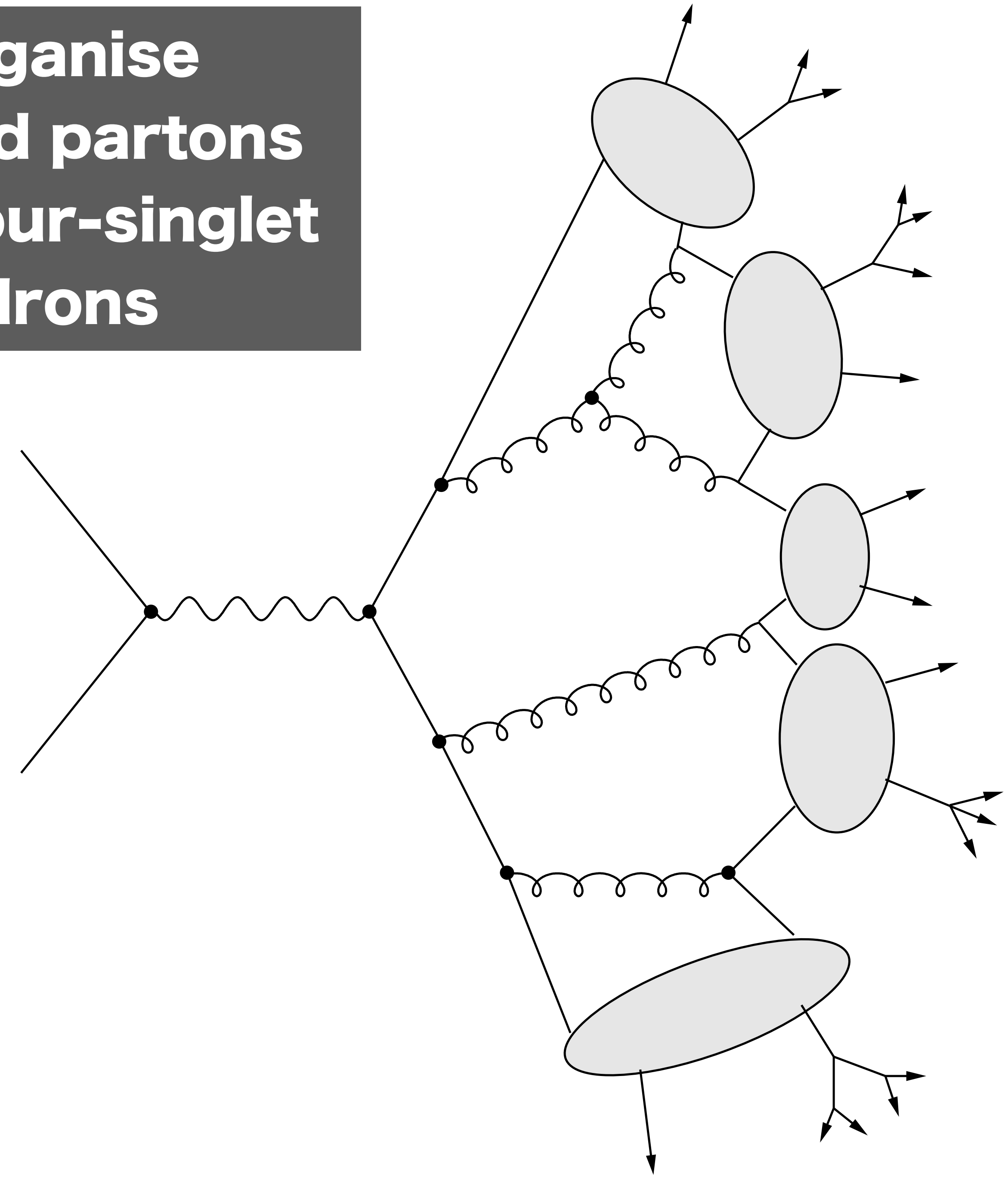
essential models for realistic events
i.e. events with hadrons

two main models for the parton–hadron transition ("hadronisation")

reorganise coloured partons into colour-singlet hadrons



String Fragmentation
(Pythia and friends)

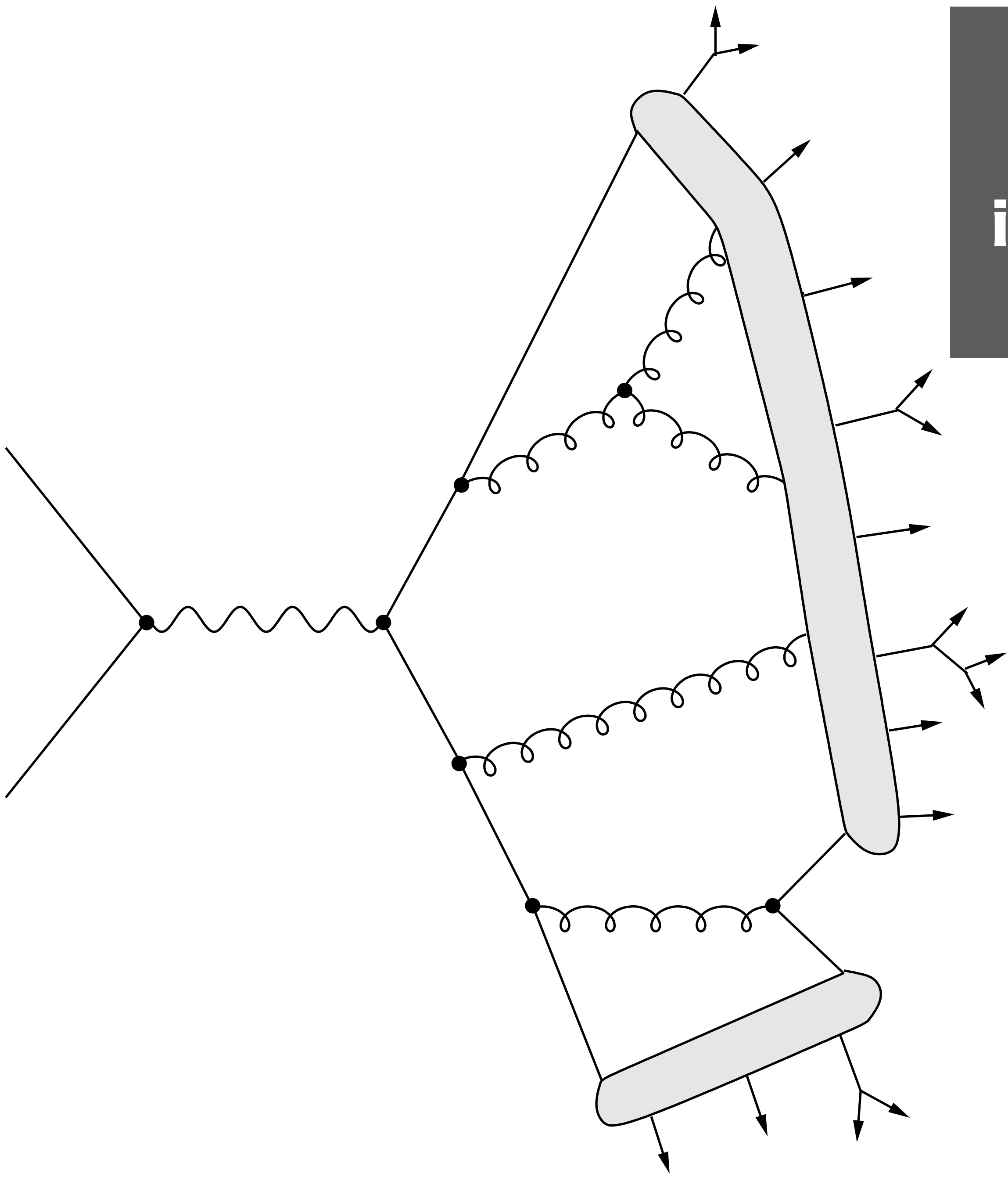


Cluster Fragmentation
(Herwig) (& Sherpa)

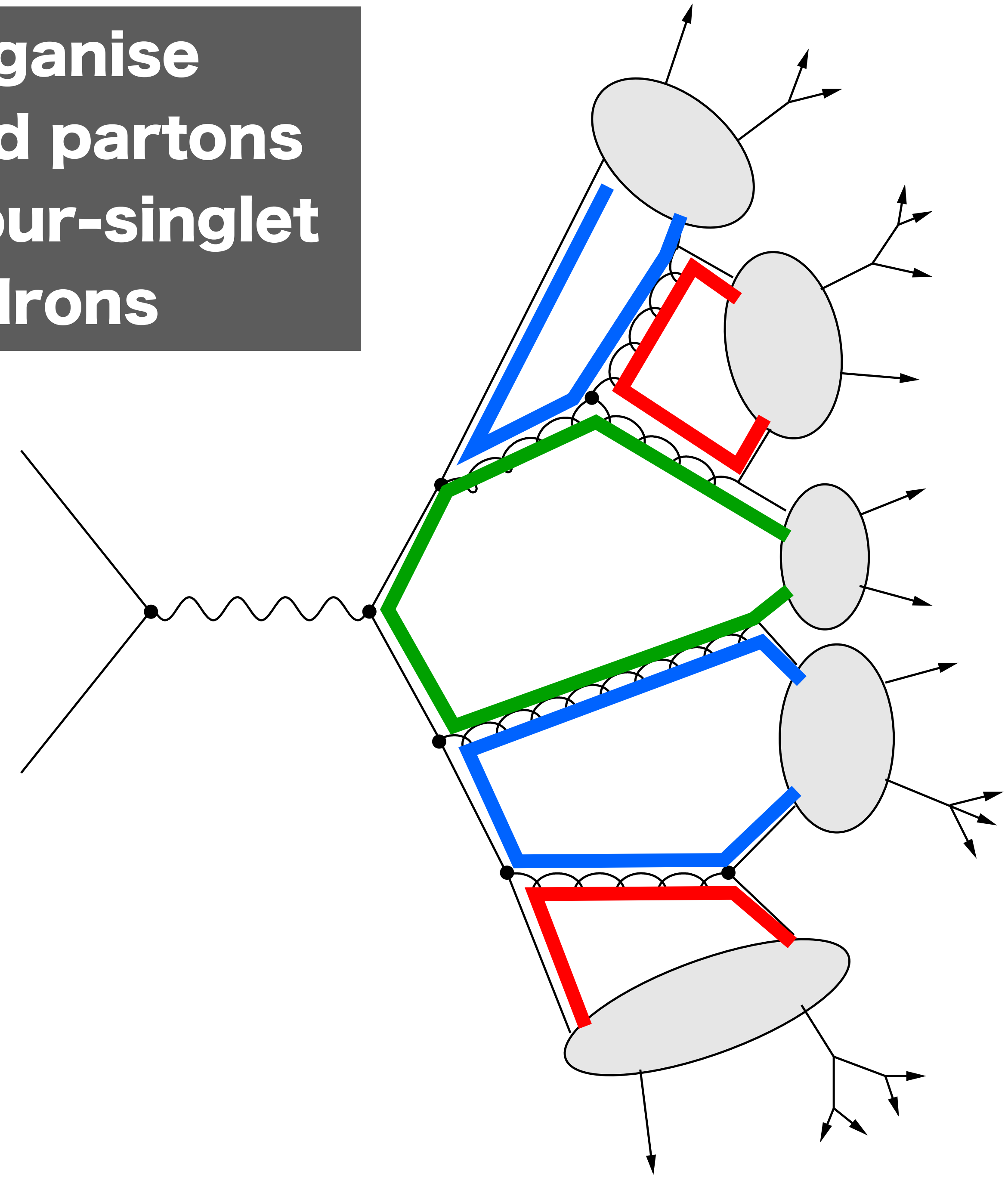
Pictures from ESW book

two main models for the parton—hadron transition ("hadronisation")

reorganise coloured partons into colour-singlet hadrons



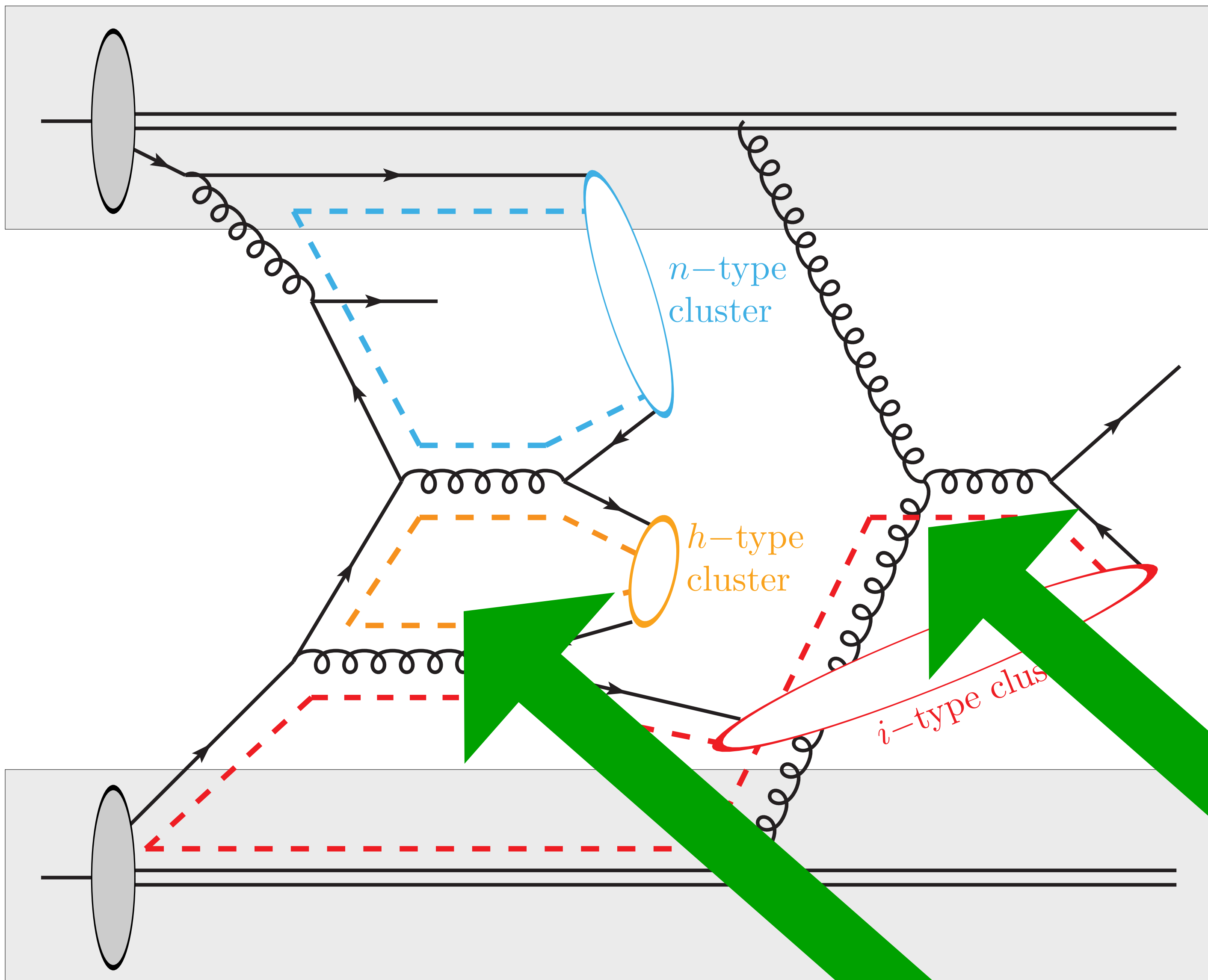
String Fragmentation
(Pythia and friends)



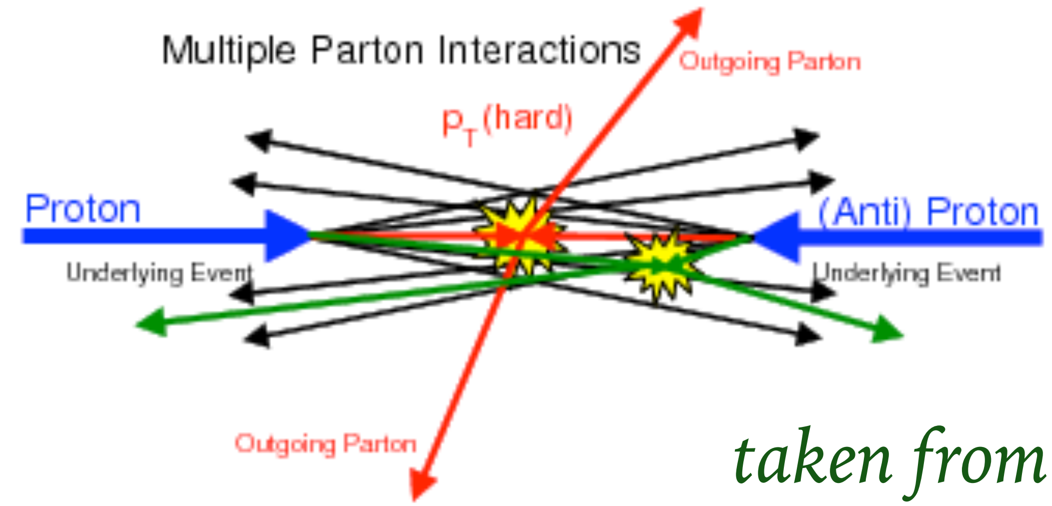
Cluster Fragmentation
(Herwig) (& Sherpa)

Pictures from ESW book

multi-parton interactions (MPI, a.k.a. underlying event)



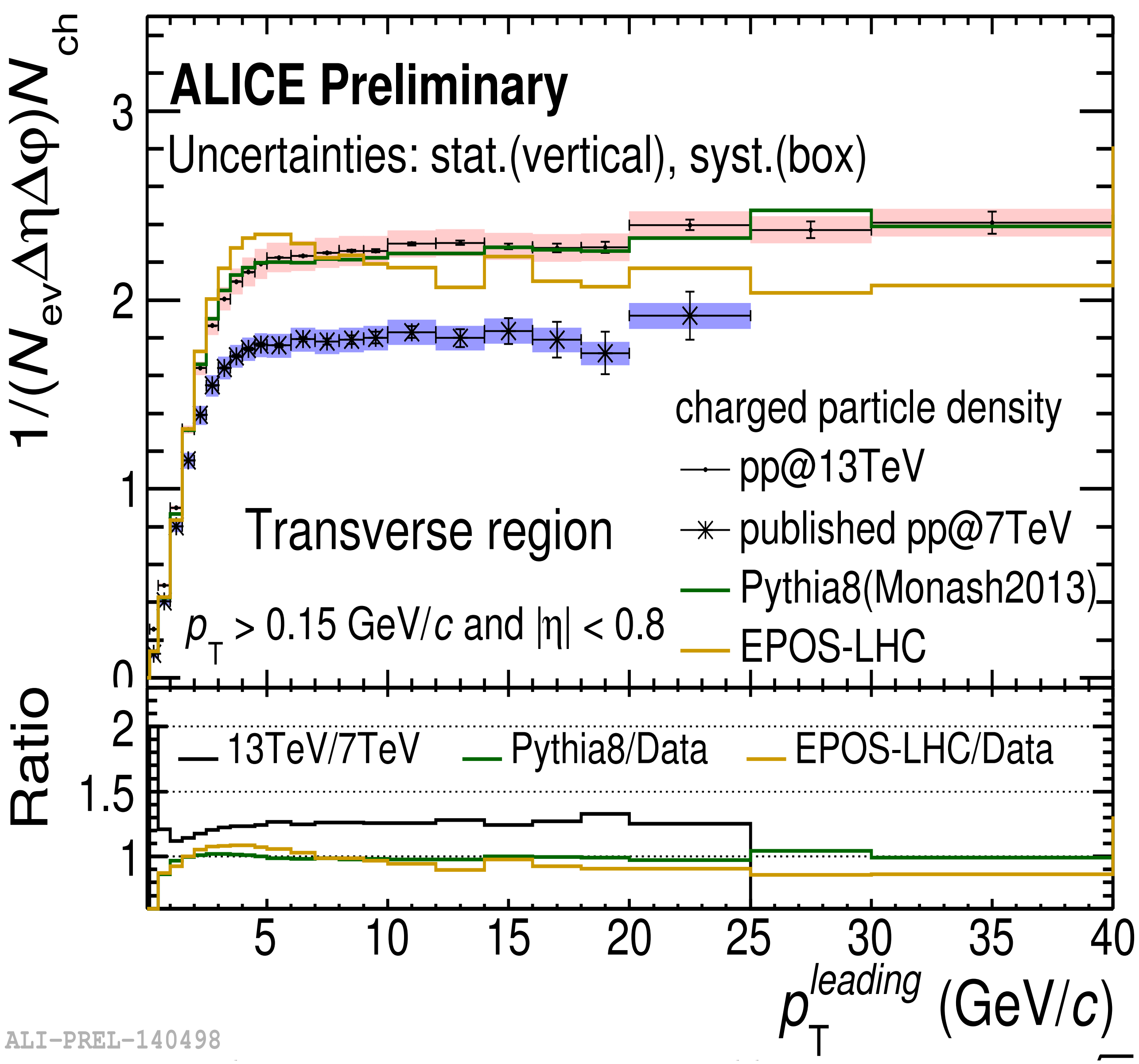
taken from
1206.2205

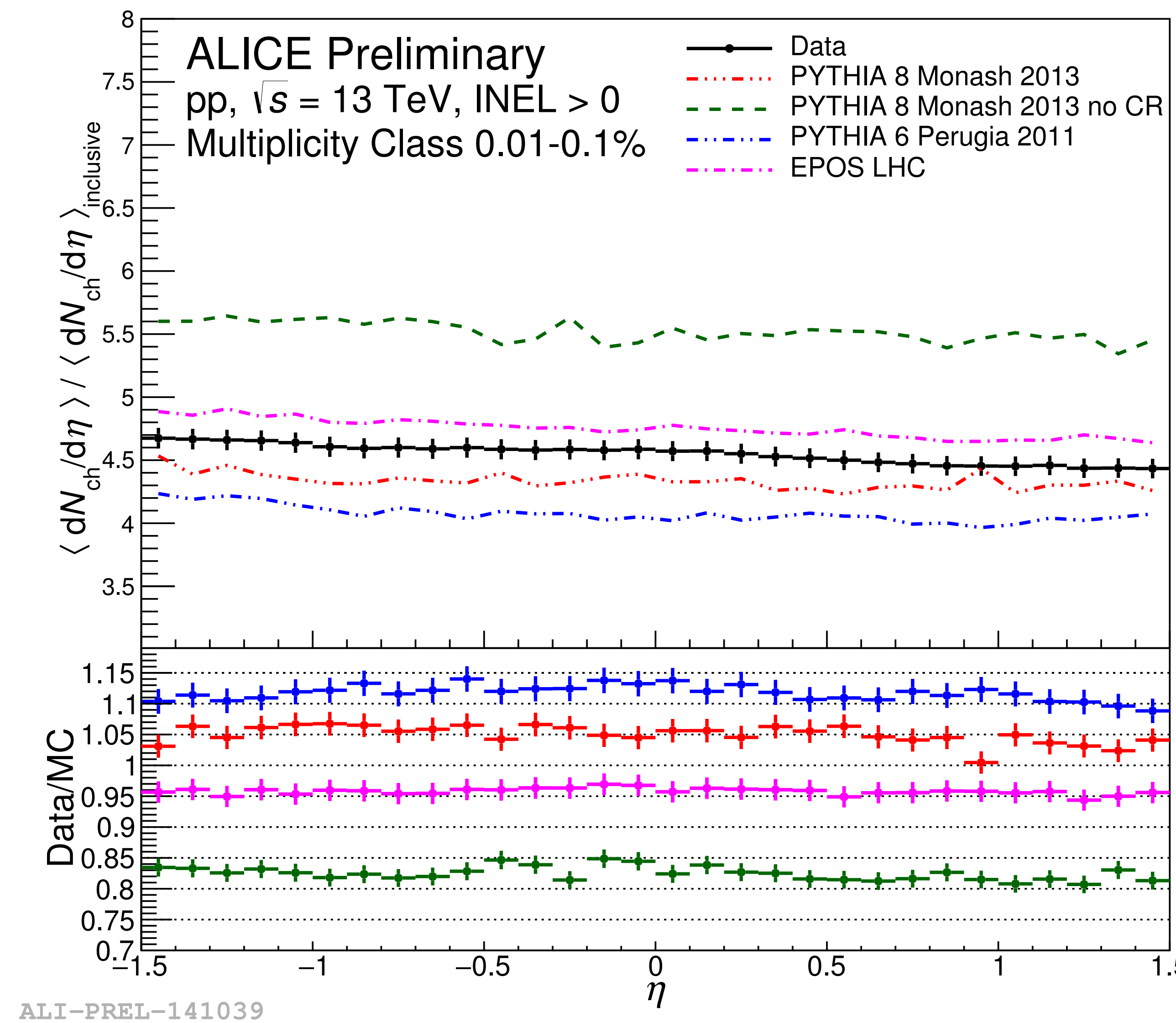


taken from
R. Field

Allow 2→2 scatterings of multiple other partons in the incoming protons

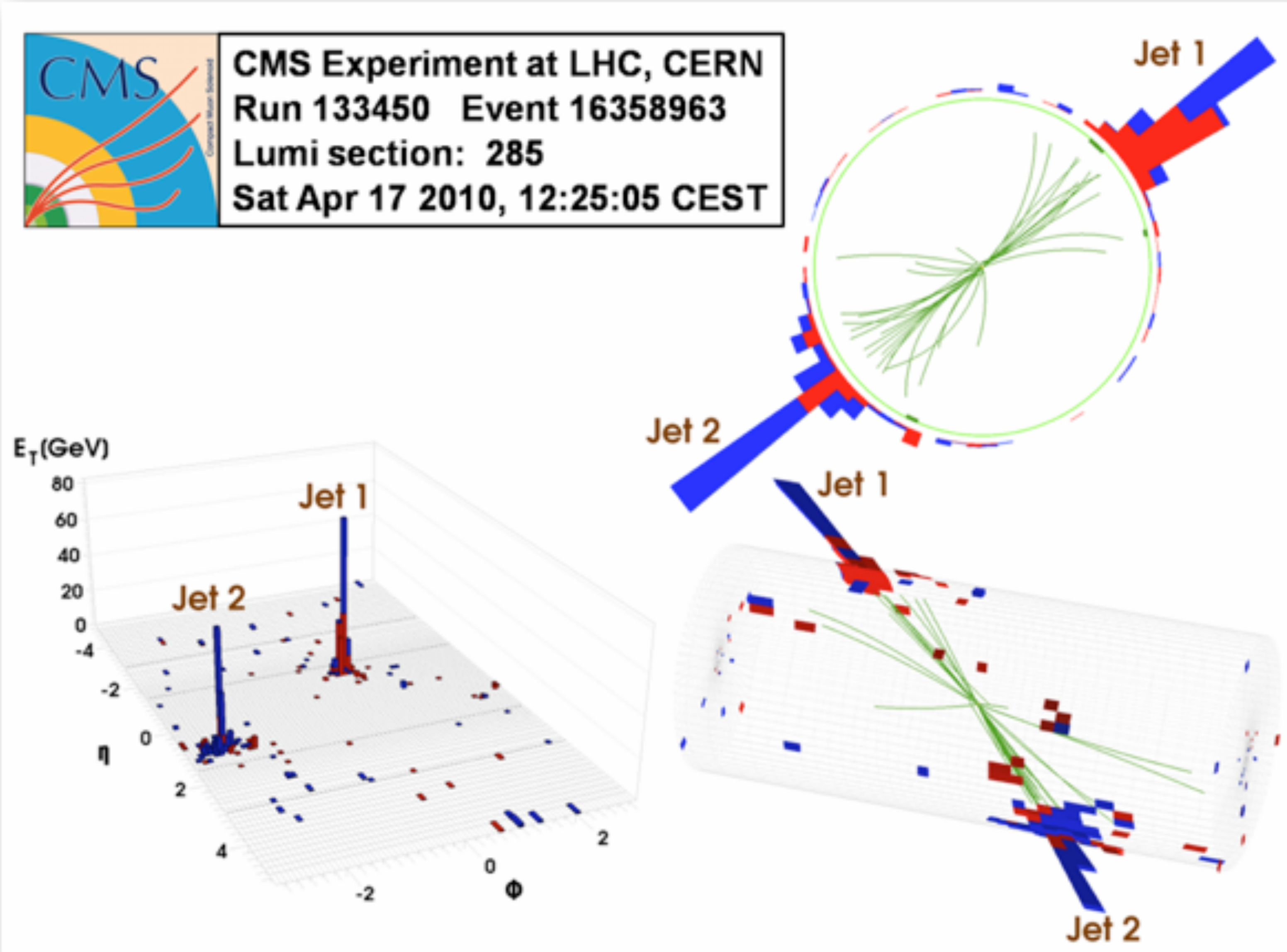
Underlying event properties v. MCs

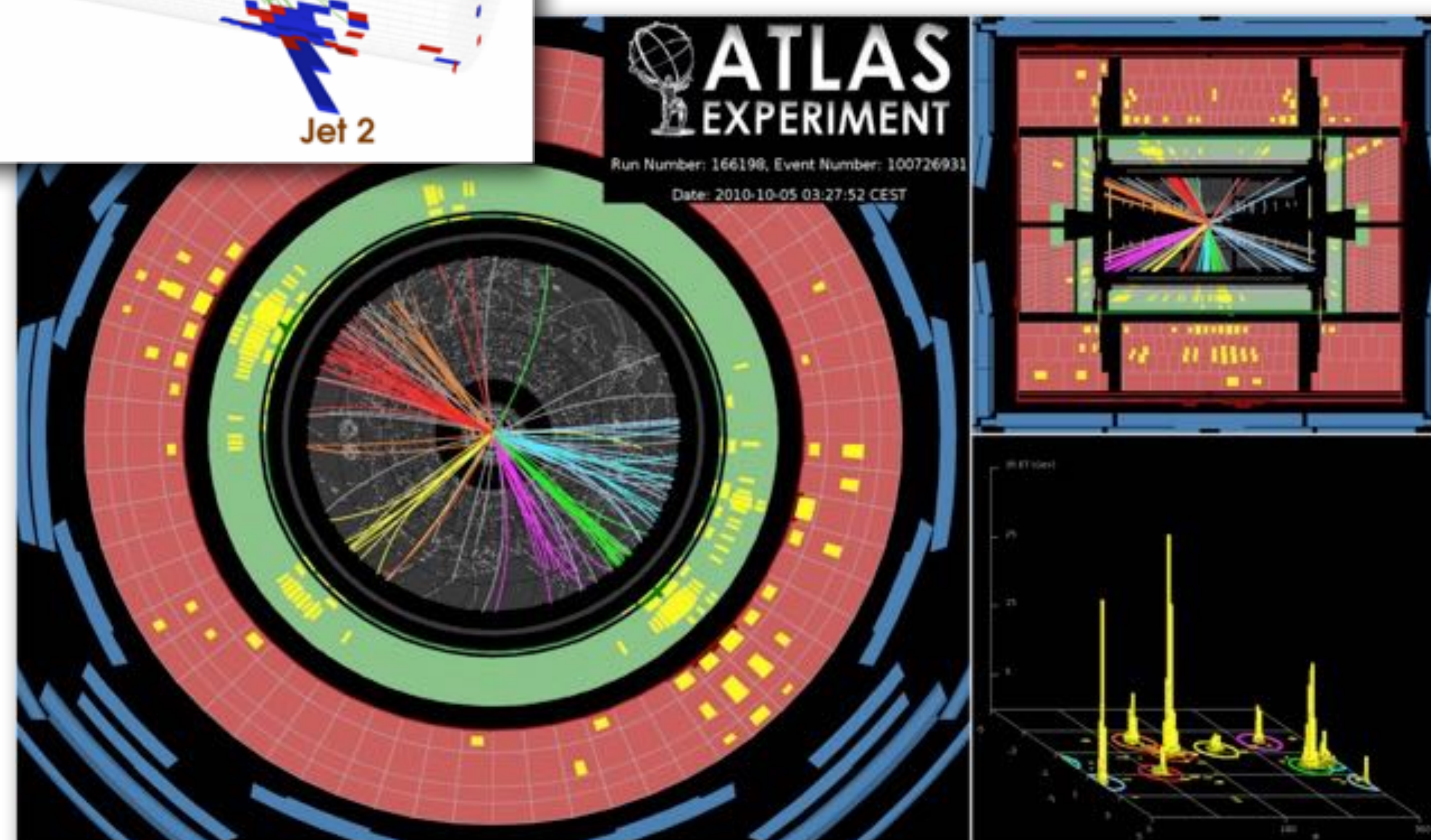




jets

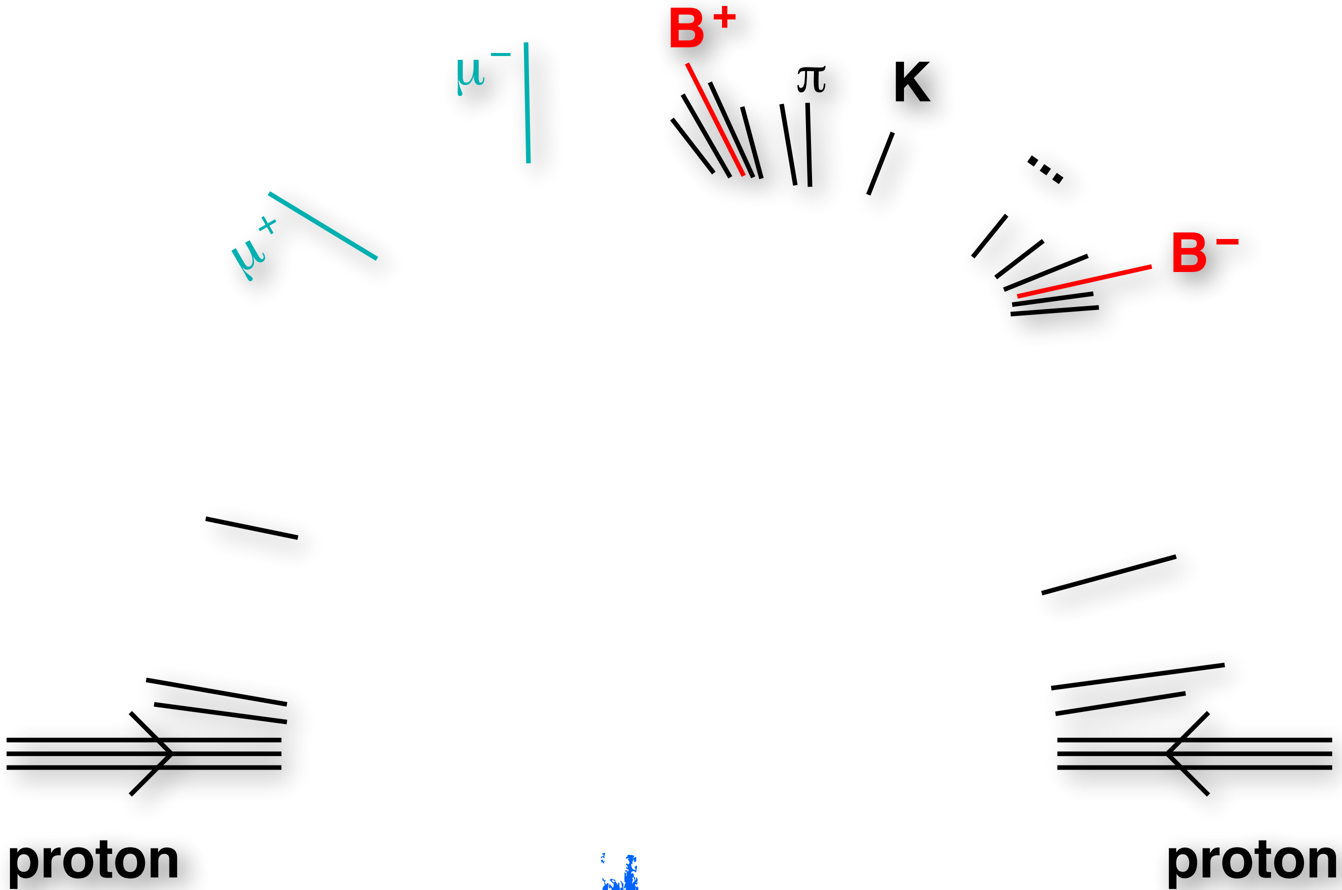
i.e. how we make sense of the hadronic part of events

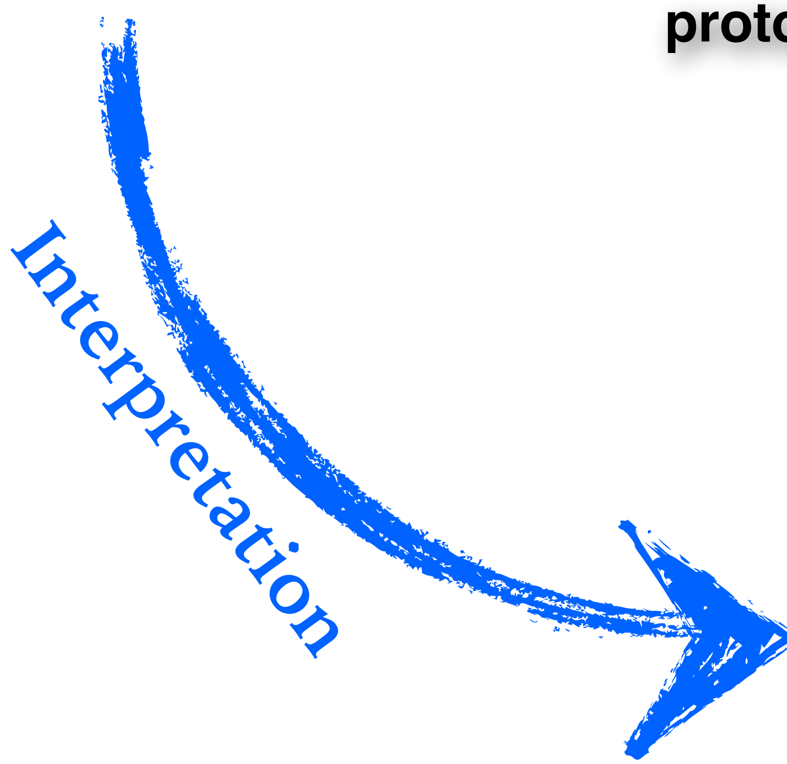


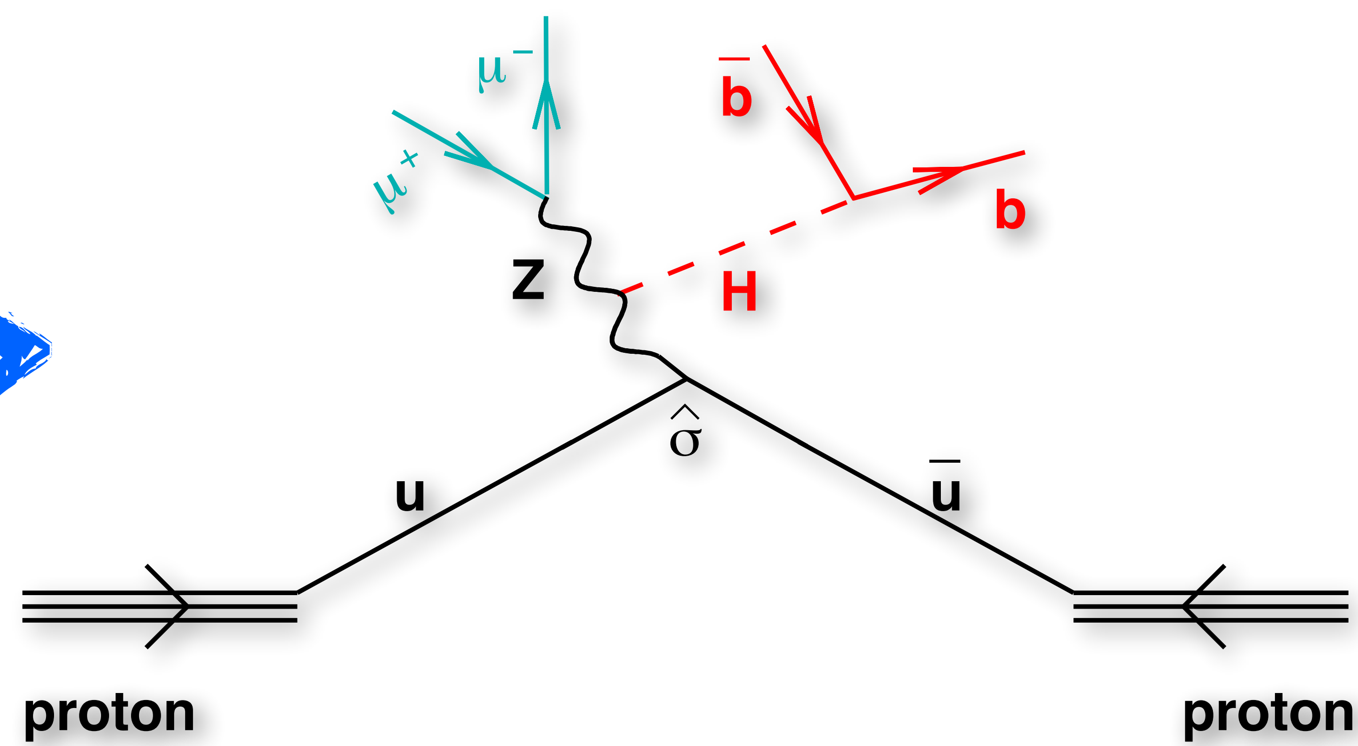


jets

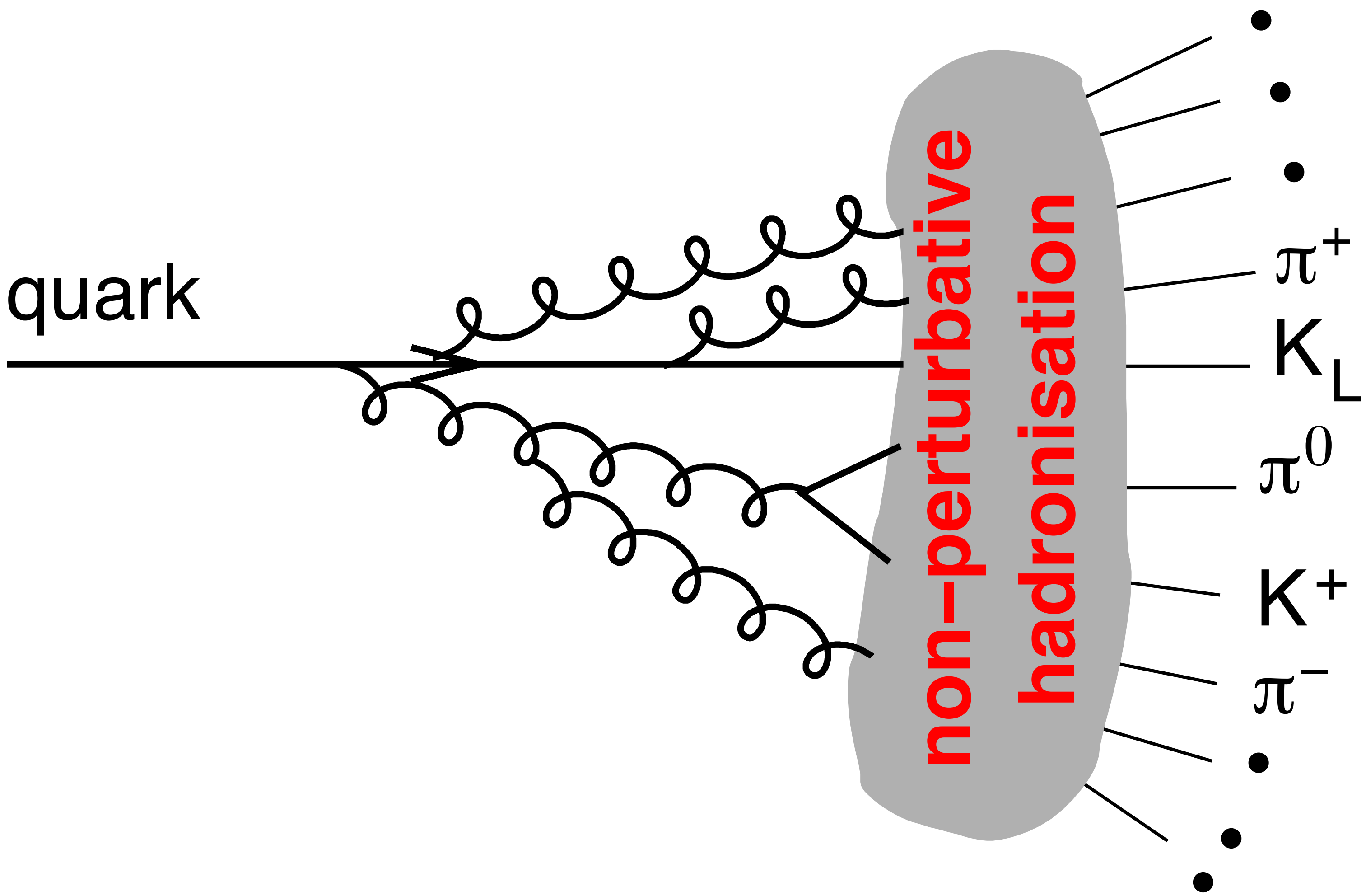
i.e. how we make sense of the hadronic part of events







Why do we see jets?



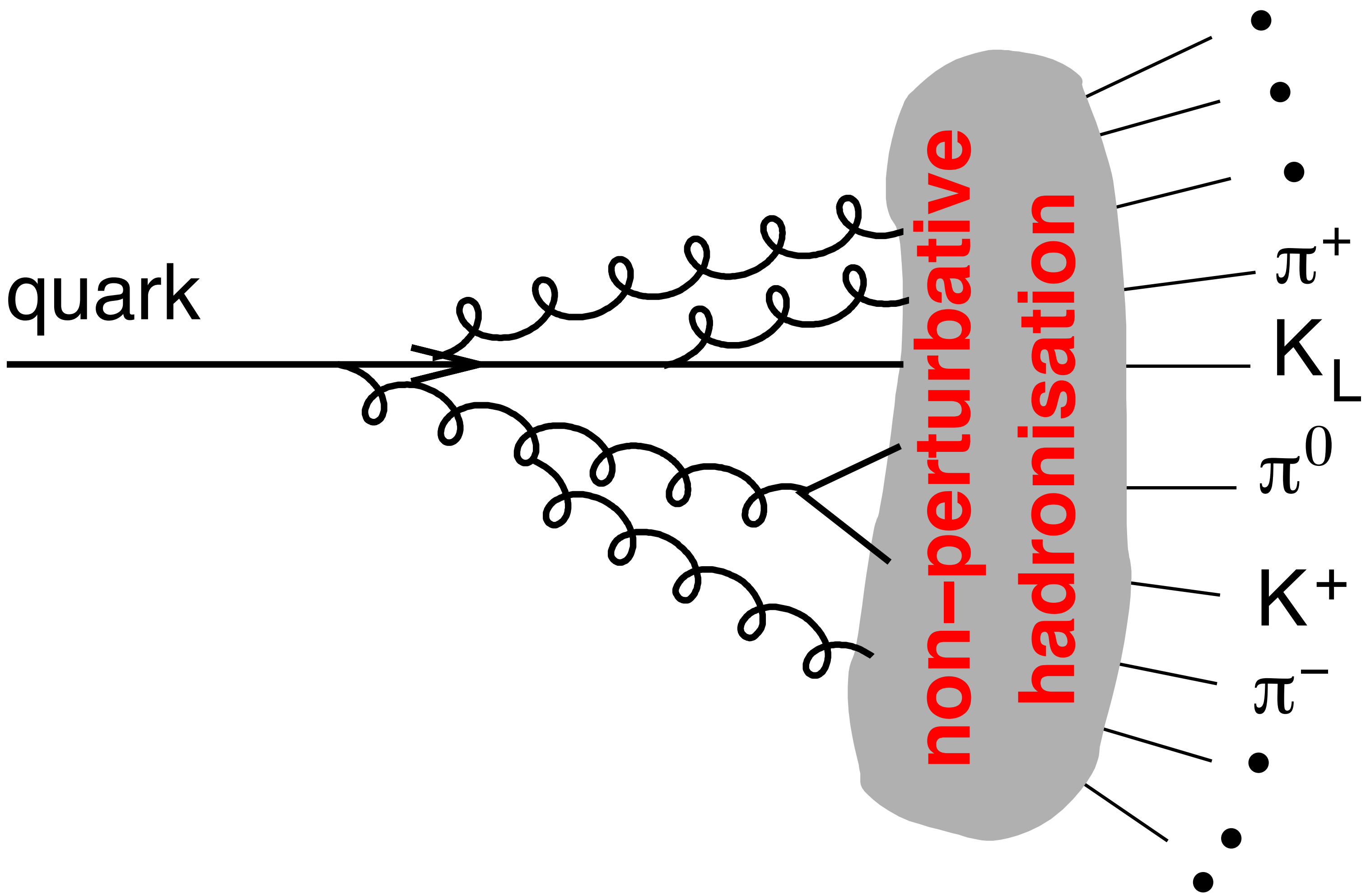
Gluon emission

$$\int \alpha_s \frac{dE}{E} \frac{d\theta}{\theta} \gg 1$$

Non-perturbative physics

$$\alpha_s \sim 1$$

Why do we see jets?



Gluon emission

$$\int \alpha_s \frac{dE}{E}\frac{d\theta}{\theta} \gg 1$$

Non-perturbative physics

$$\alpha_s \sim 1$$

While you can see jets with your eyes, **to do quantitative physics**, you need an algorithmic procedure that **defines what exactly a jet is**

make a choice, specify a Jet Definition

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

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

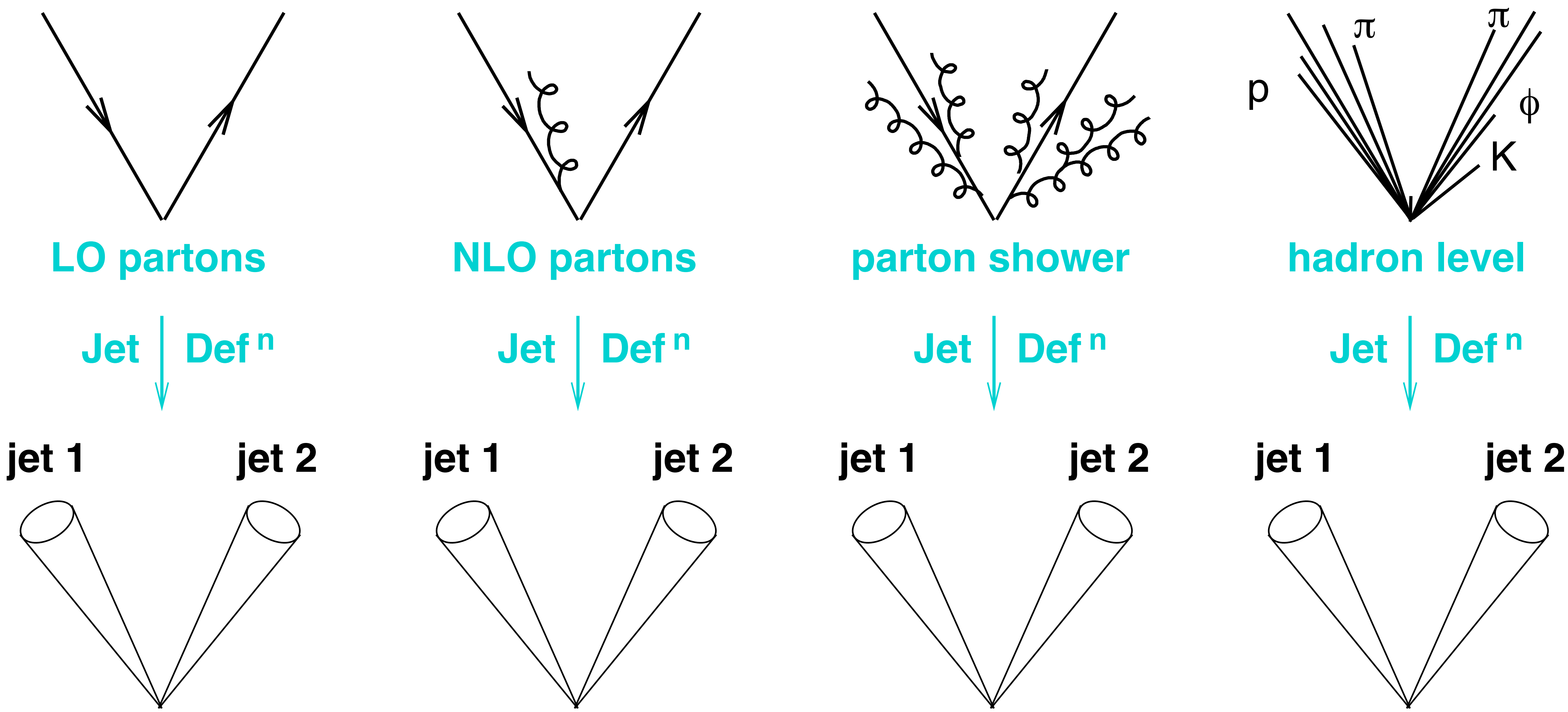
jets

- Which particles do you put together into a same jet?
- How do you recombine their momenta
(4-momentum sum is the obvious choice, right?)

"Jet [definitions] are legal contracts between theorists and experimentalists"
-- MJ Tannenbaum

They're also a way of organising the information in an event
1000's of particles per events, up to 40.000,000 events per second

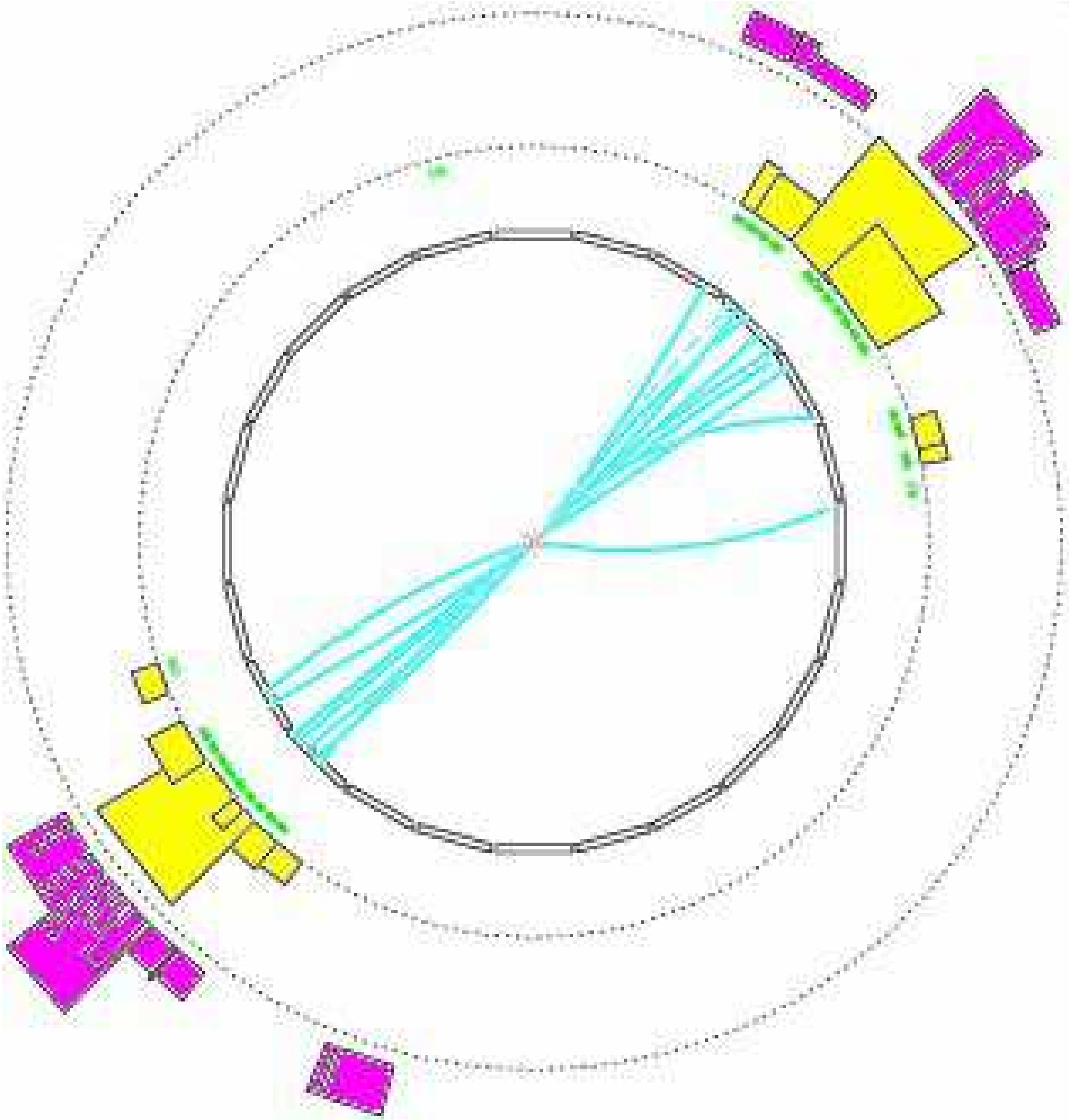
what should a jet definition achieve?

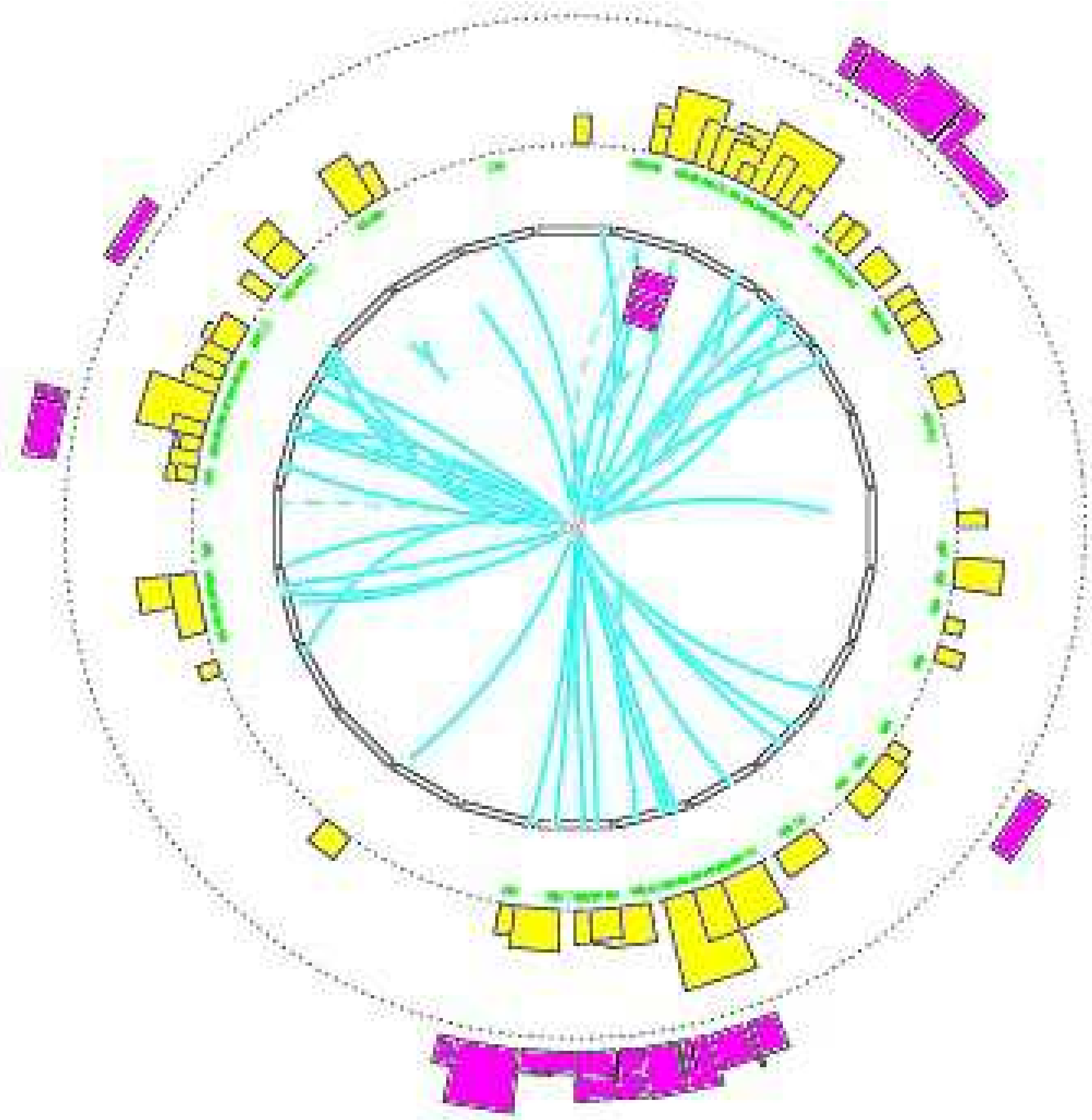


projection to jets should be resilient to QCD effects

Reconstructing jets is an ambiguous task

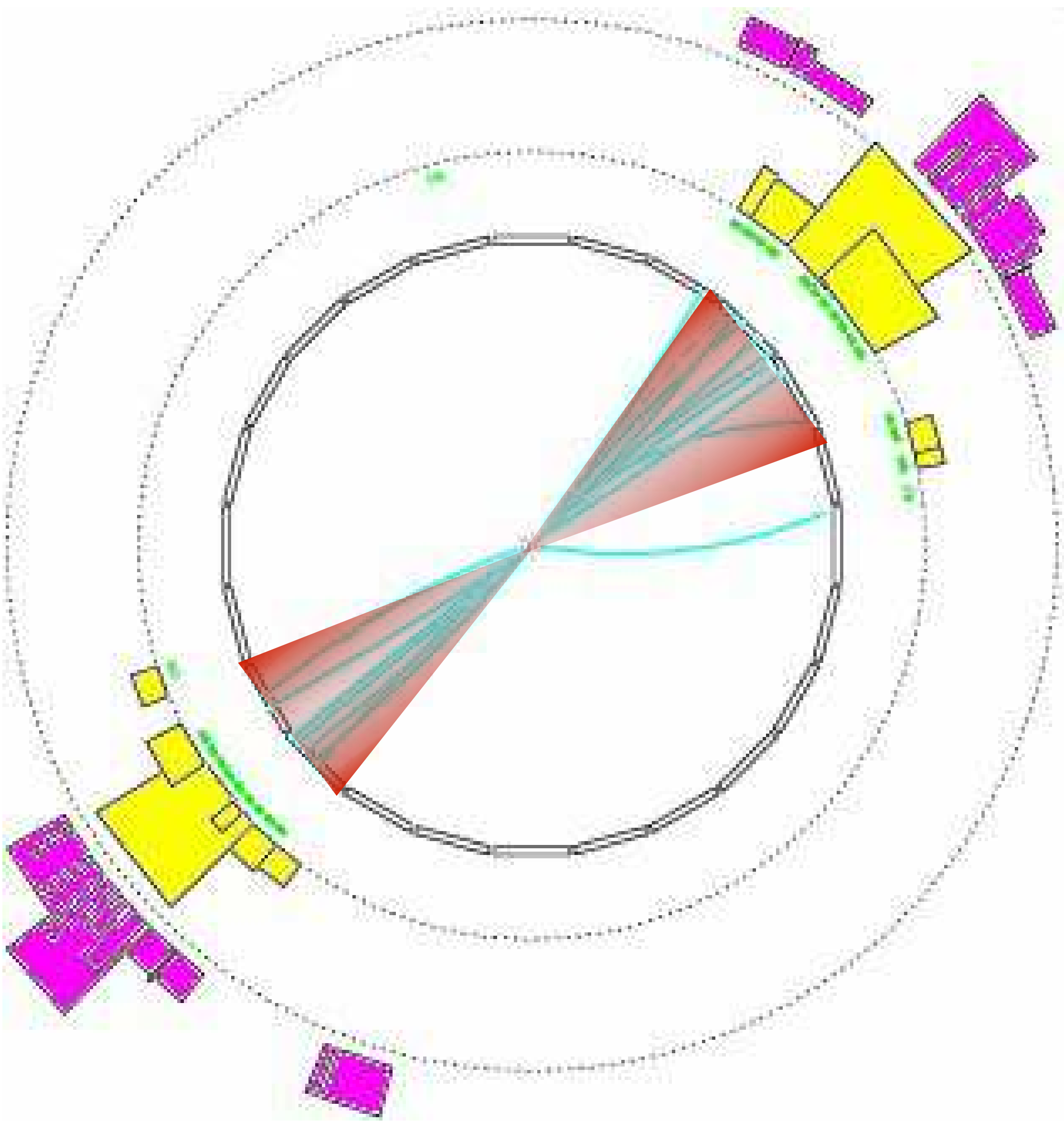
not in handout

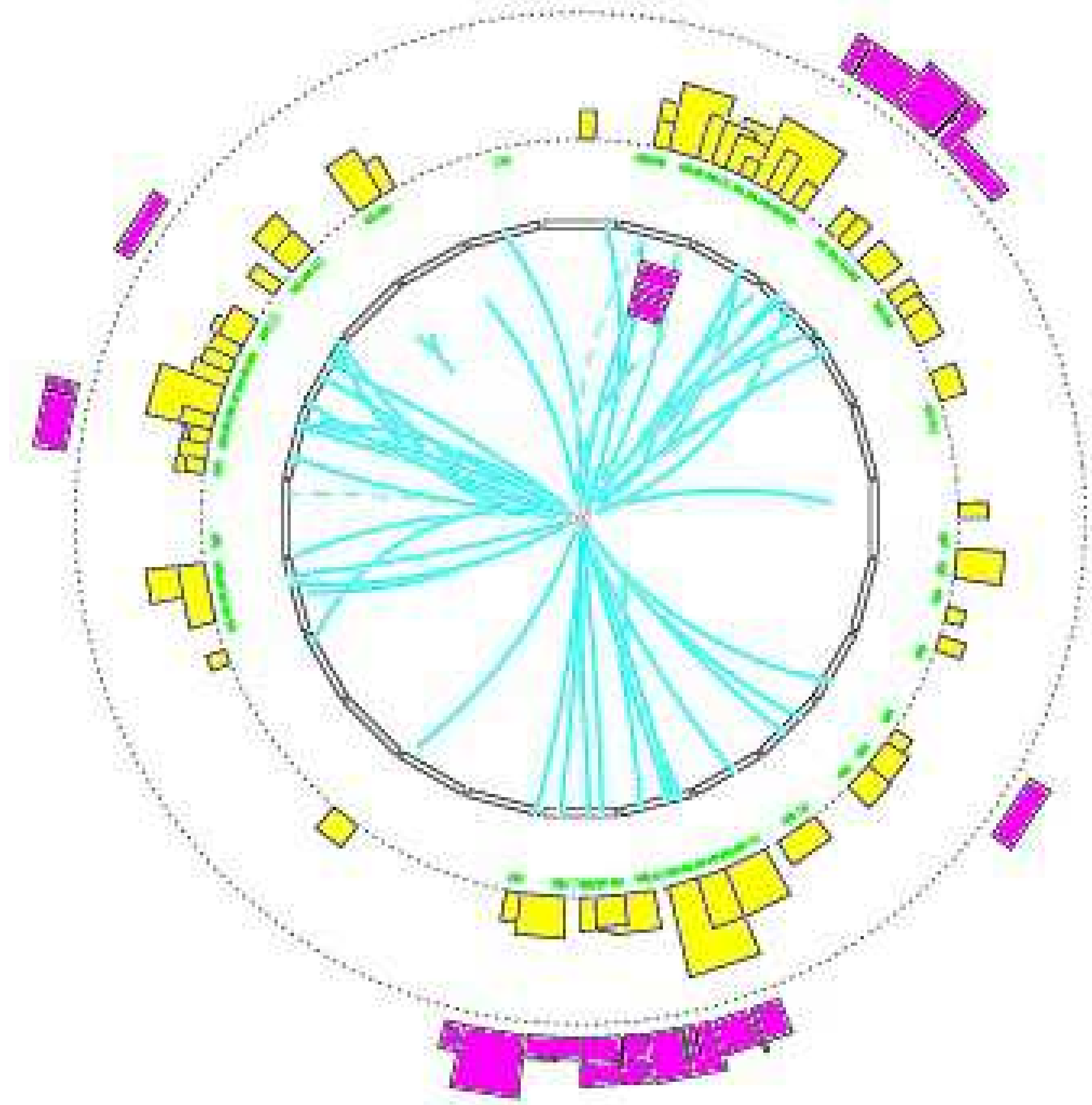




Reconstructing jets is an ambiguous task

not in handout

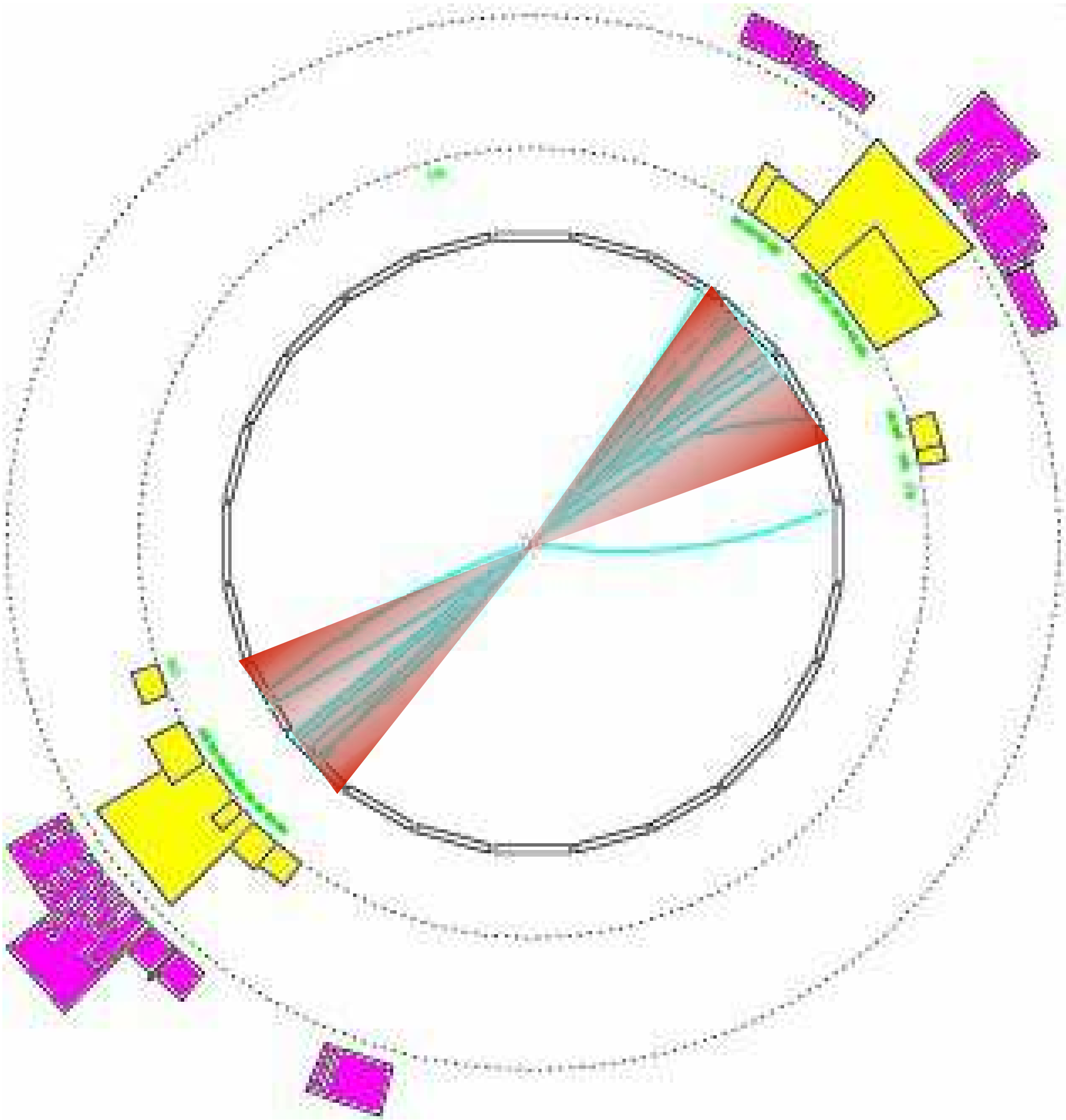




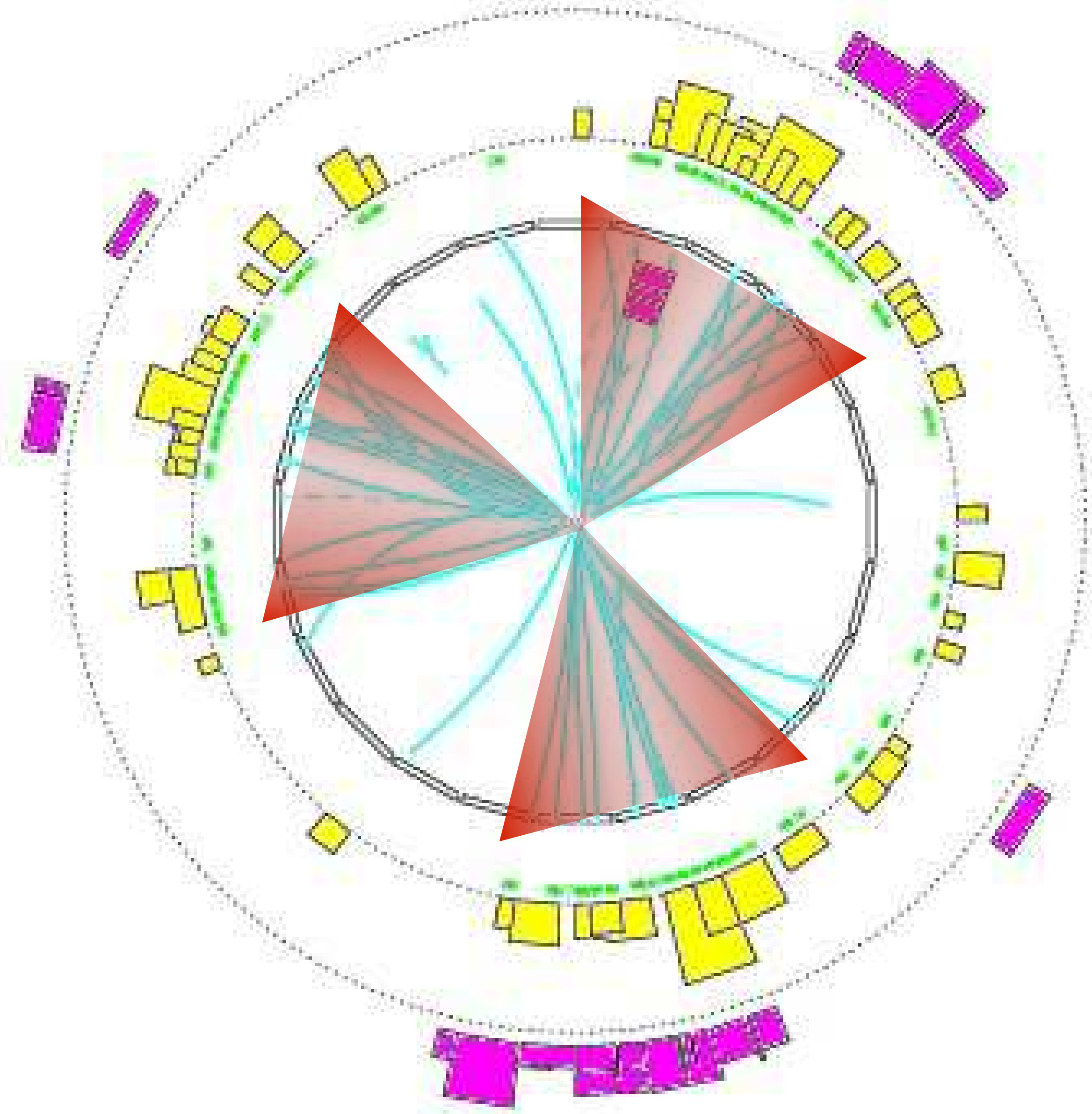
2 clear jets

Reconstructing jets is an ambiguous task

not in handout



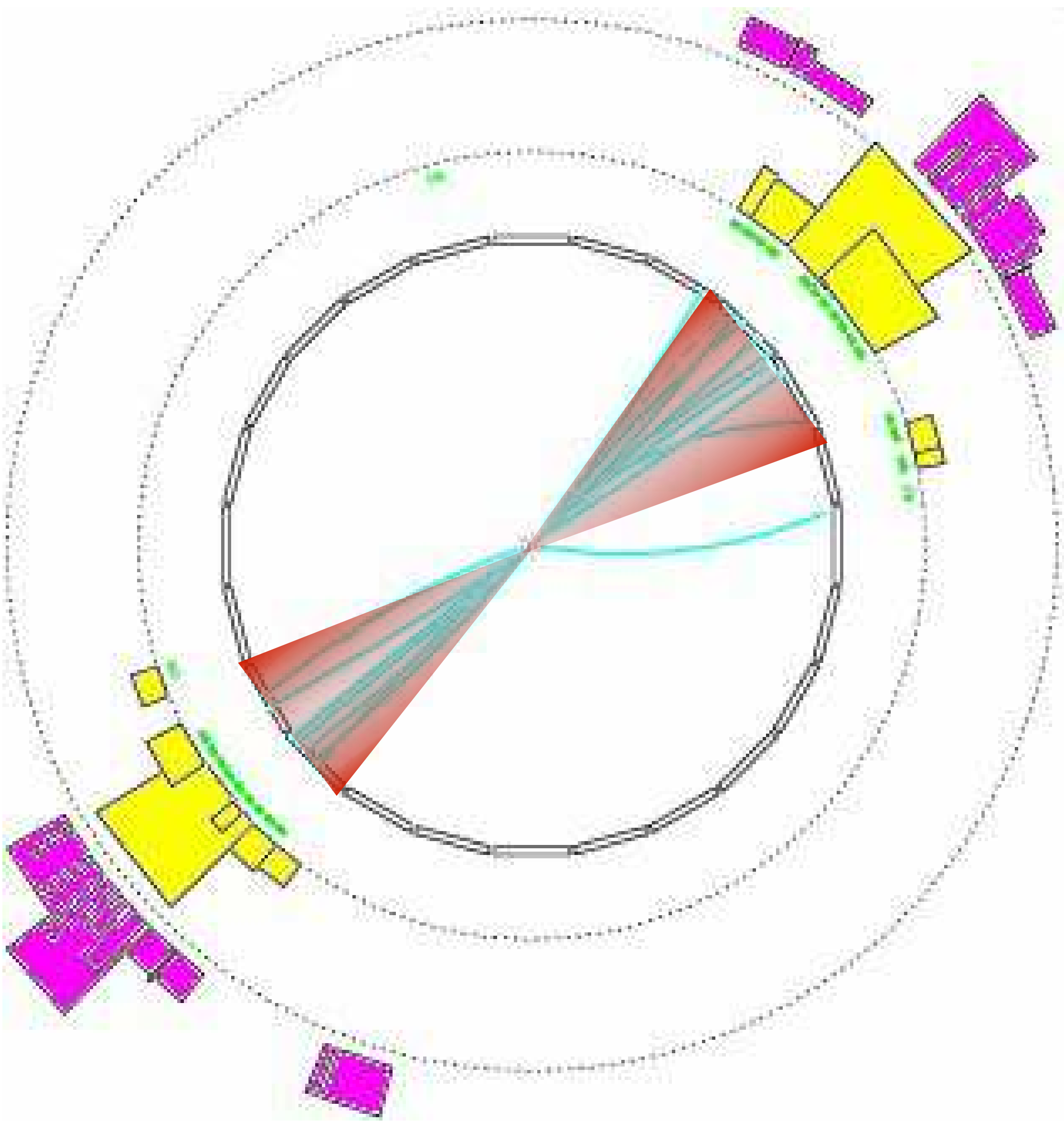
2 clear jets



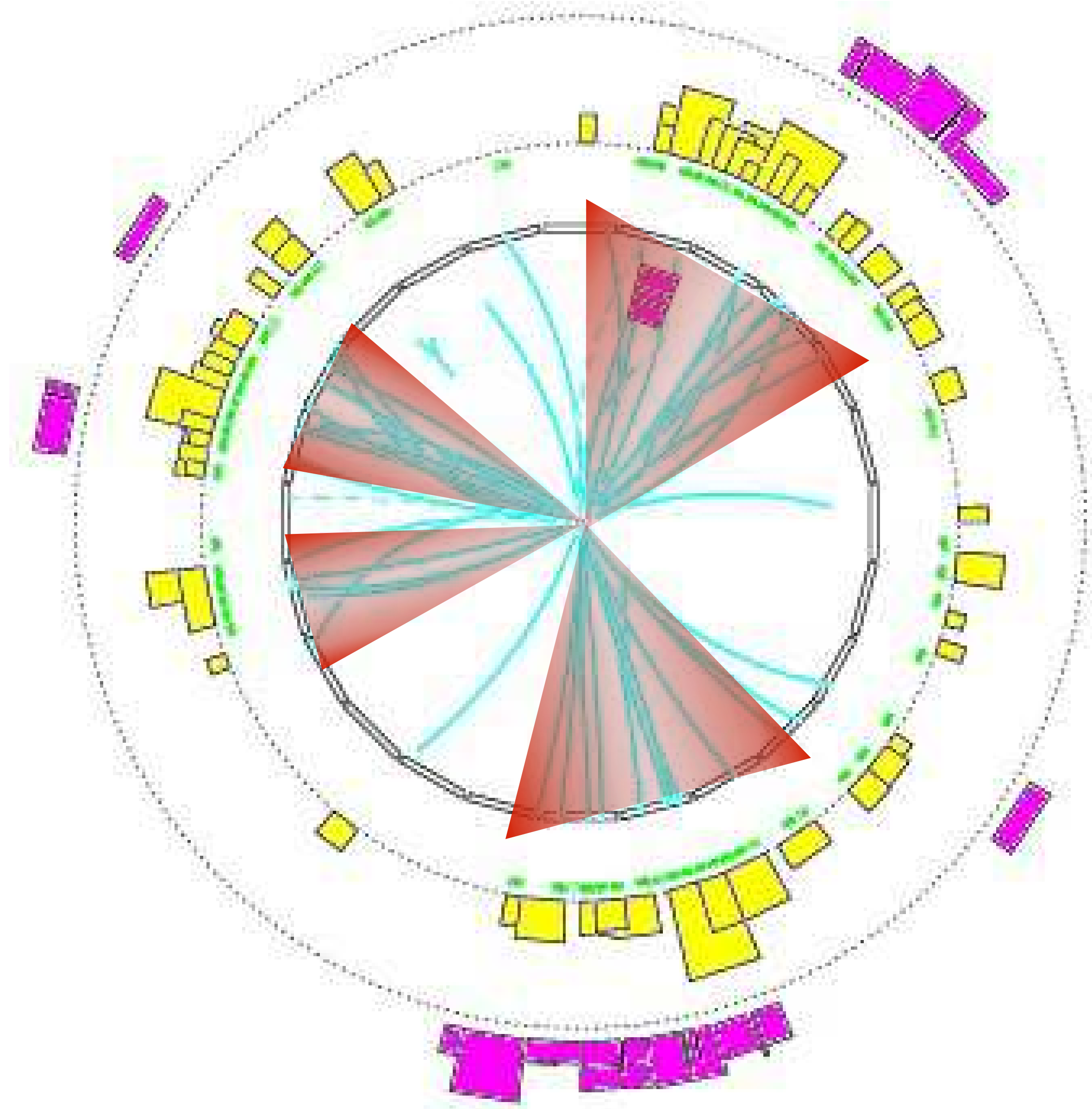
3 jets?

not in handout

Reconstructing jets is an ambiguous task



2 clear jets



3 jets?
or 4 jets?

Jet definition ingredients

not in handout

Jet algorithm

A set of rules that you apply to combine particles into jets

Jet algorithm parameters

Thresholds that help specify when two particles belong to the same jet or not.

Most hadron collider jet algorithms have two threshold parameters:

- **Jet angular radius parameter R**:
  particles closer in angle than R get recombined
  (NB: usually implemented as a condition on the distance parameter on the standard hadron collider rapidity-azimuth [y,φ] cylinder)

- **Transverse momentum threshold**:
  jets should have $p_T > p_{T,min}$

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2$$

$$y = \frac{1}{2}\ln\frac{E + p_z}{E - p_z}$$

A Sequential recombination algorithm

Involves calculating "clustering distance" between pairs of particles

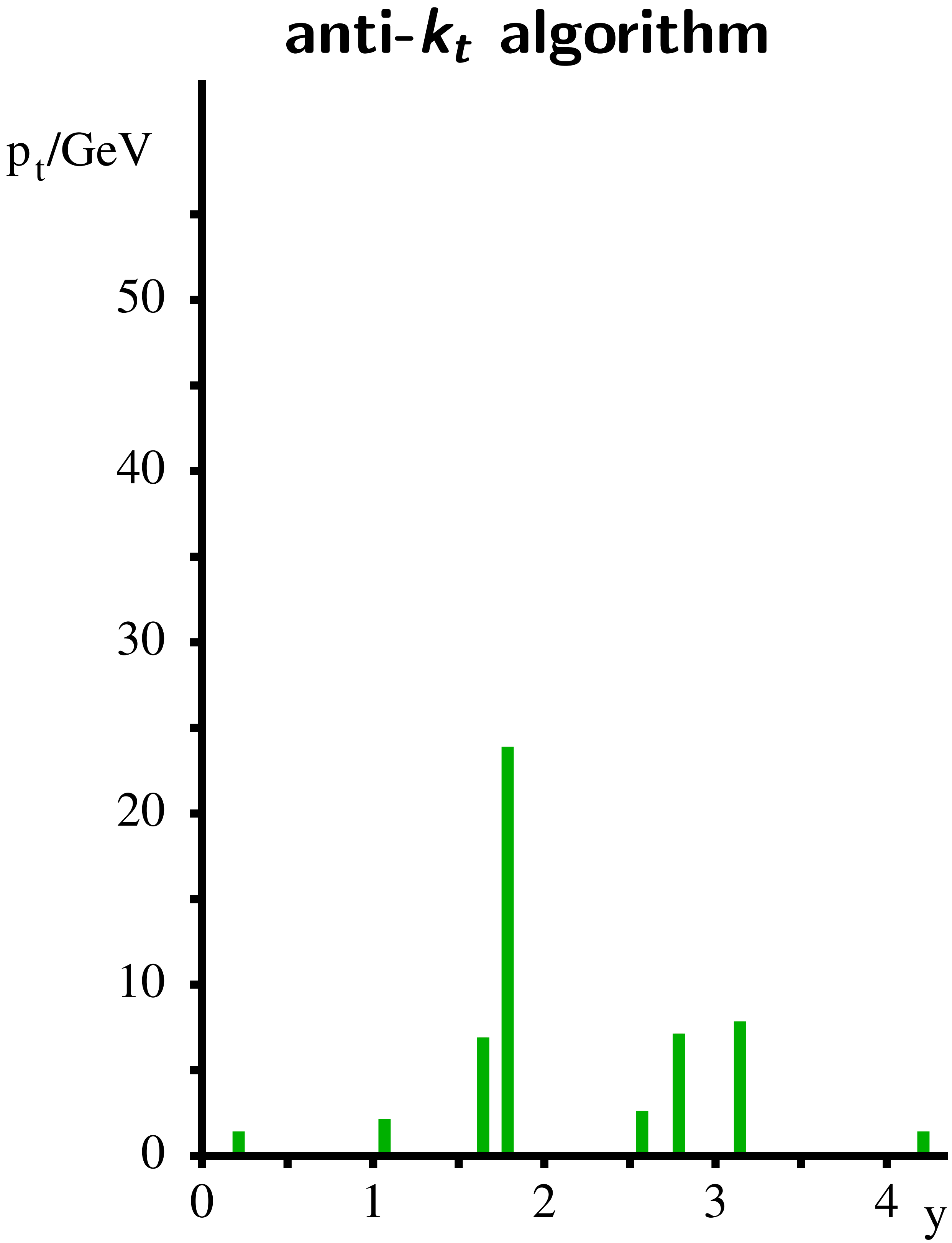
$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)}\frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$

1. Find smallest of d_{ij}, d_{iB}
2. If ij, recombine them
3. If iB, call i a jet and remove from list of particles
4. repeat from step 1 until no particles left

   Only use jets with $p_t > p_{t,min}$

anti-k_t algorithm

Cacciari, GPS & Soyez, 0802.1189



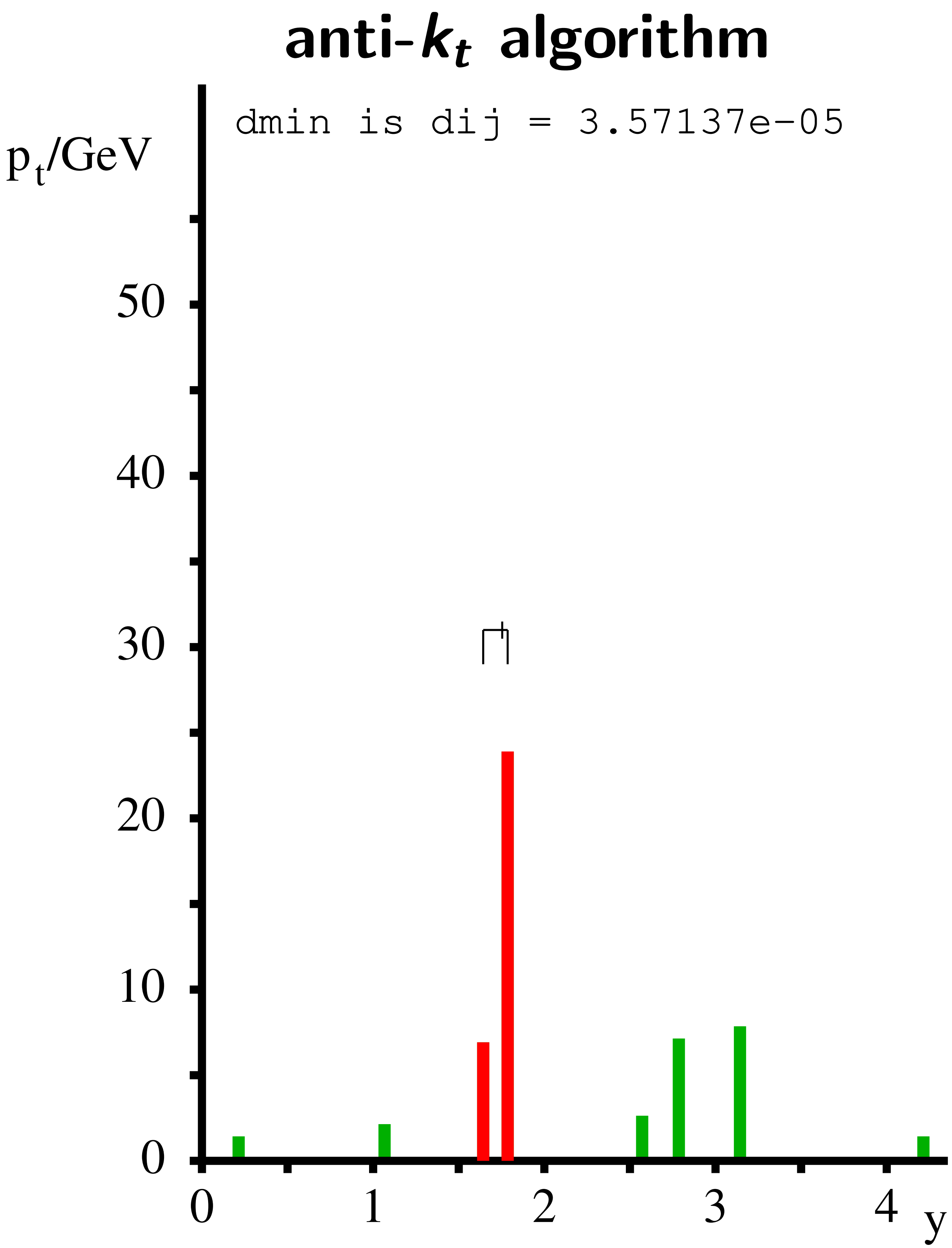
How do different sequential recombination jet algorithms build up the jet?

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R = 2.0$
$p_{T,min} = 20$ GeV
at LHC: $R = 0.4 - 1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$



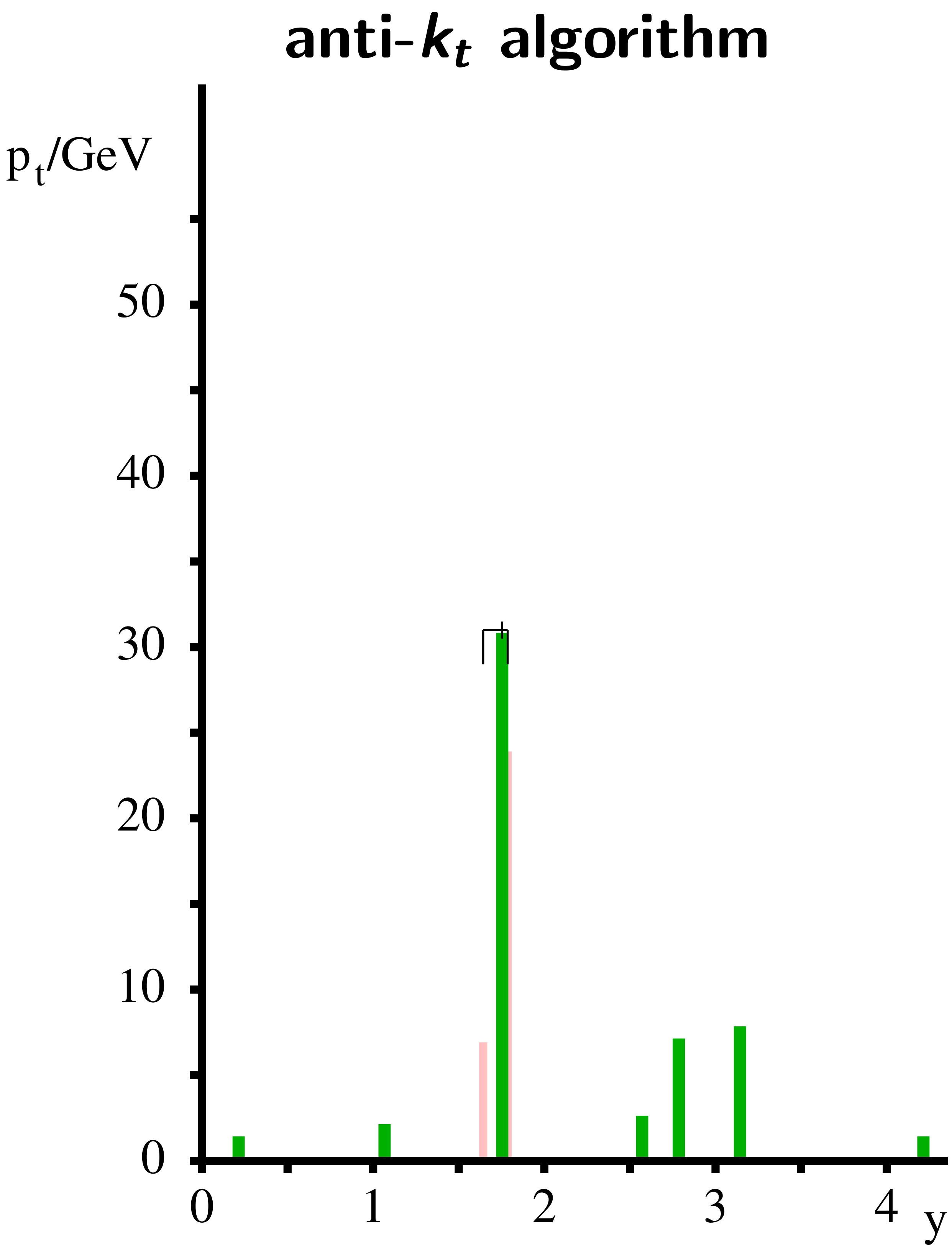
How do different sequential recombination jet algorithms build up the jet?

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R=2.0$
$p_{T,min}=20$ GeV
at LHC: $R=0.4-1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2}\ln\frac{E+p_z}{E-p_z}$$



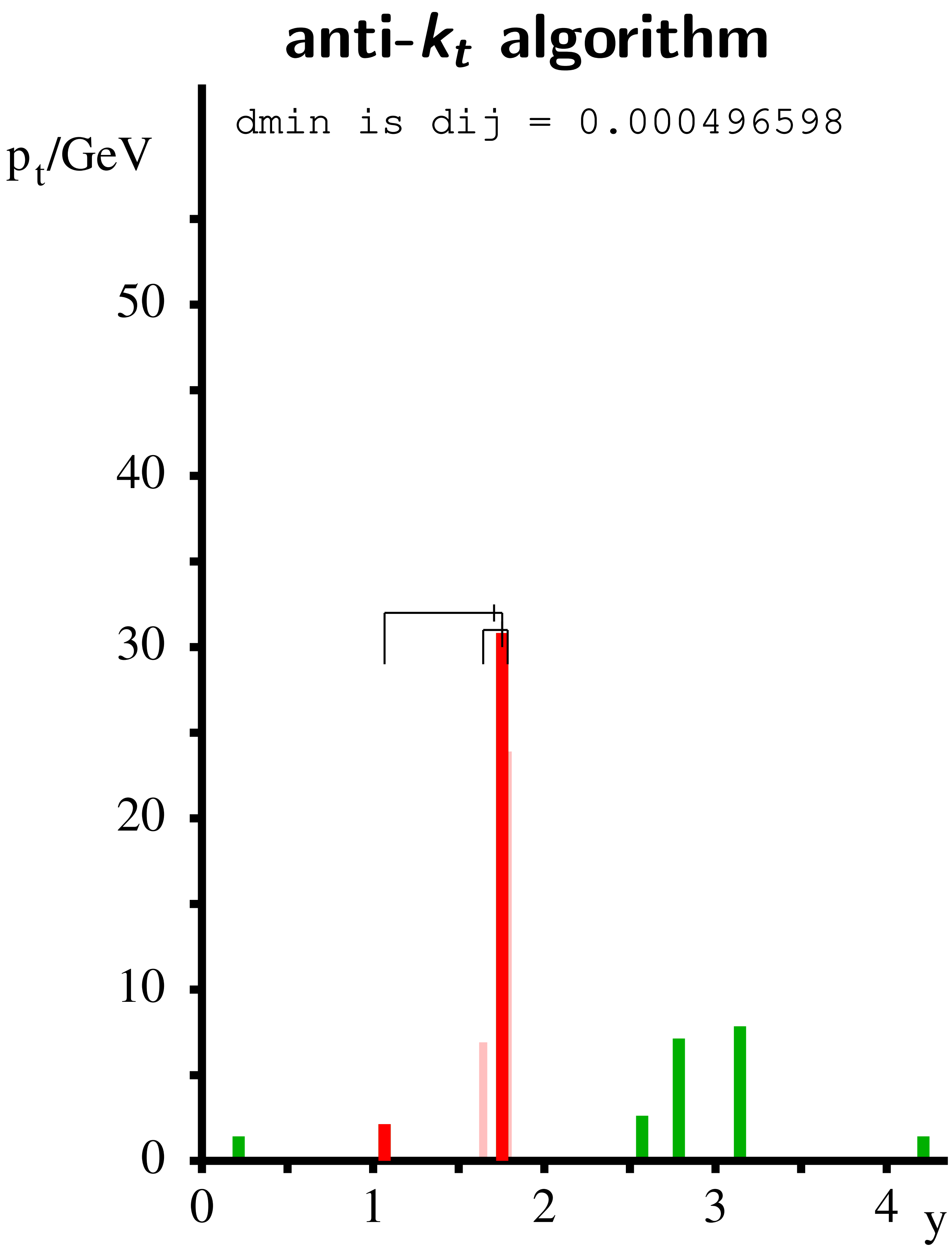
How do different sequential recombination jet algorithms build up the jet?

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R=2.0$
$p_{T,min}=20$ GeV
at LHC: $R=0.4-1.0$]

$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2$ $\quad y = \frac{1}{2}\ln\frac{E+p_z}{E-p_z}$



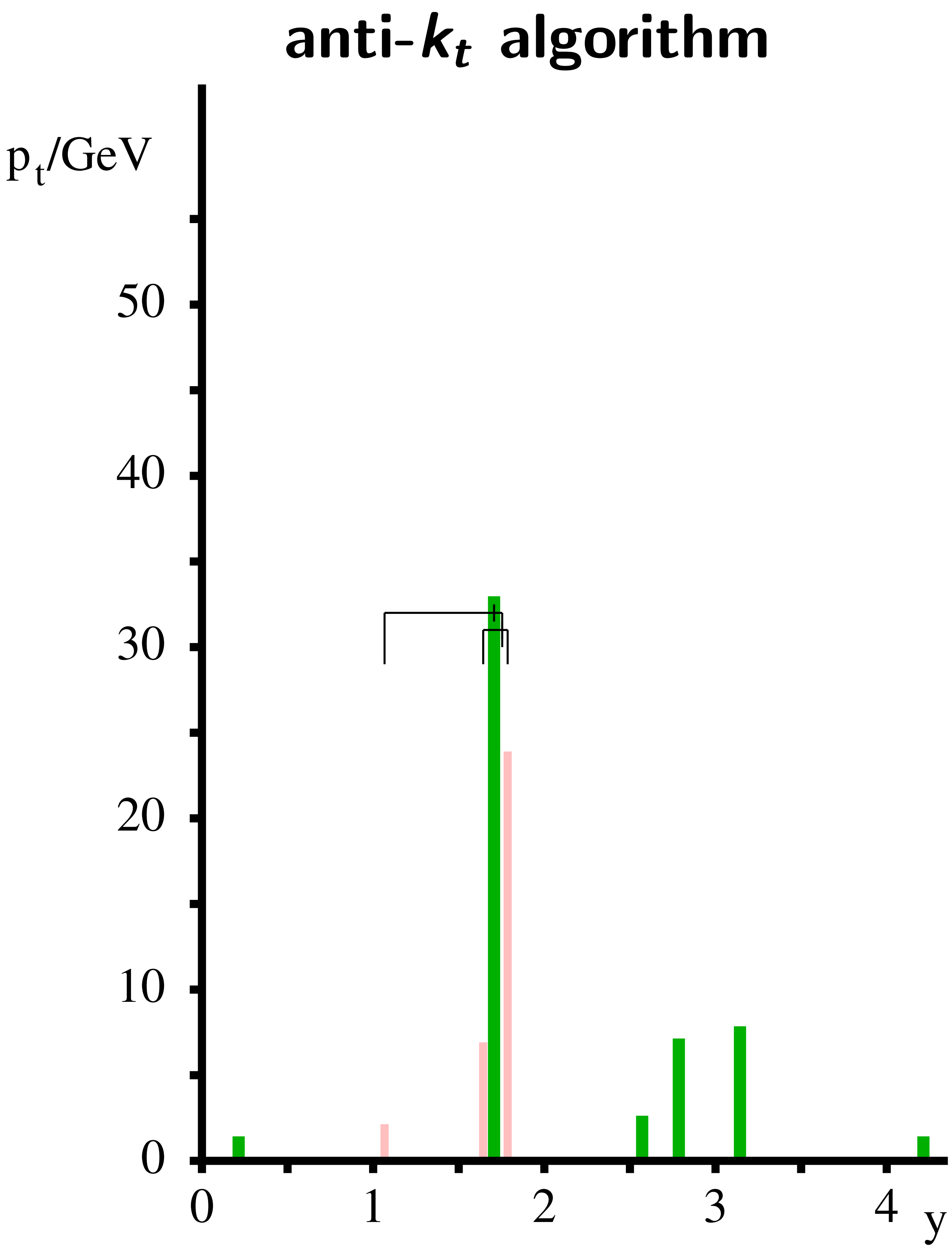
How do different sequential recombination jet algorithms build up the jet?

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R=2.0$
$p_{T,min}=20$ GeV
at LHC: $R=0.4-1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2}\ln\frac{E+p_z}{E-p_z}$$



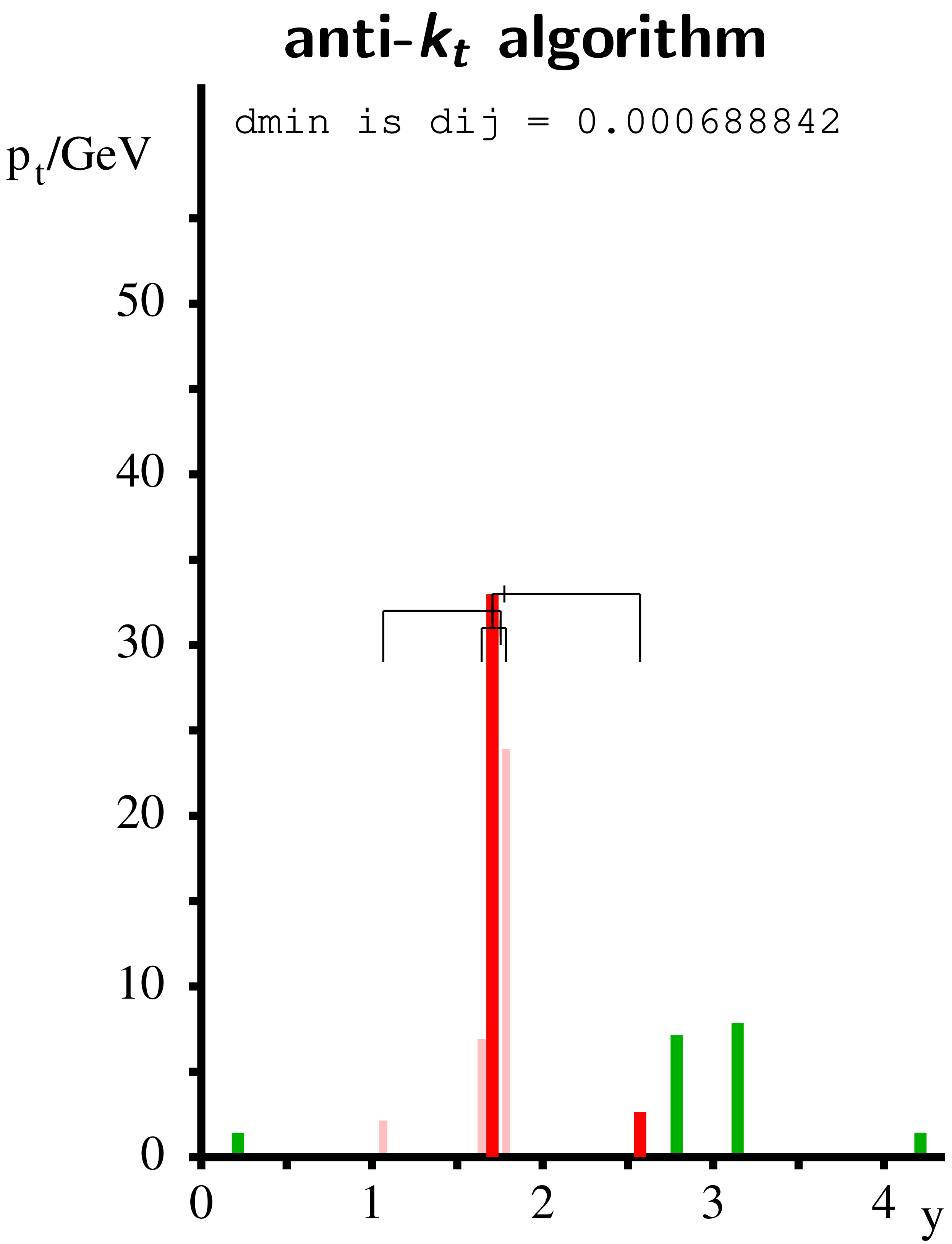
How do different sequential recombination jet algorithms build up the jet?

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R = 2.0$
$p_{T,min} = 20$ GeV
at LHC: $R = 0.4 - 1.0$]

$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2$ $\qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$



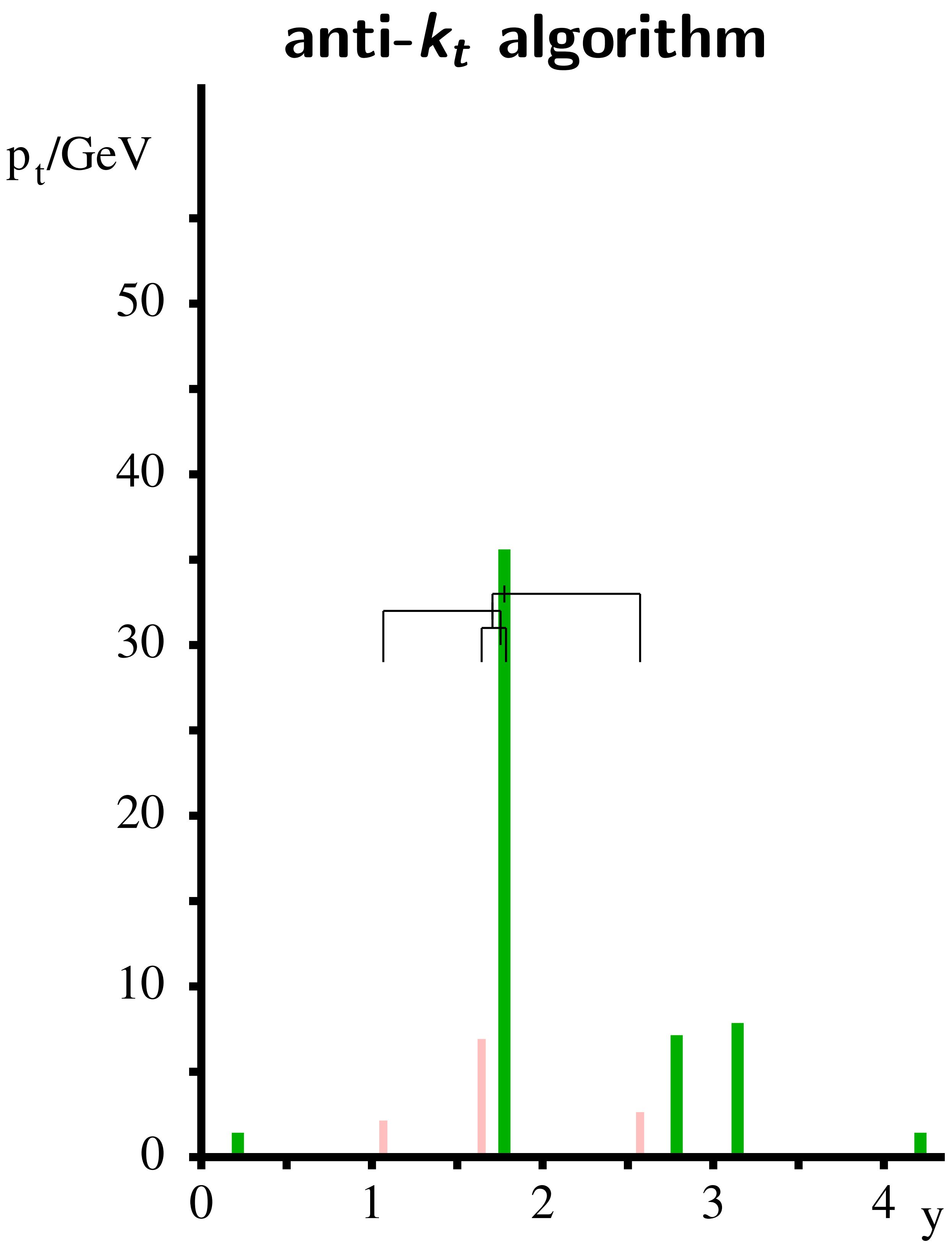
How do different sequential recombination jet algorithms build up the jet?

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here R=2.0
$p_{T,min}$=20 GeV
at LHC: R=0.4 – 1.0]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$



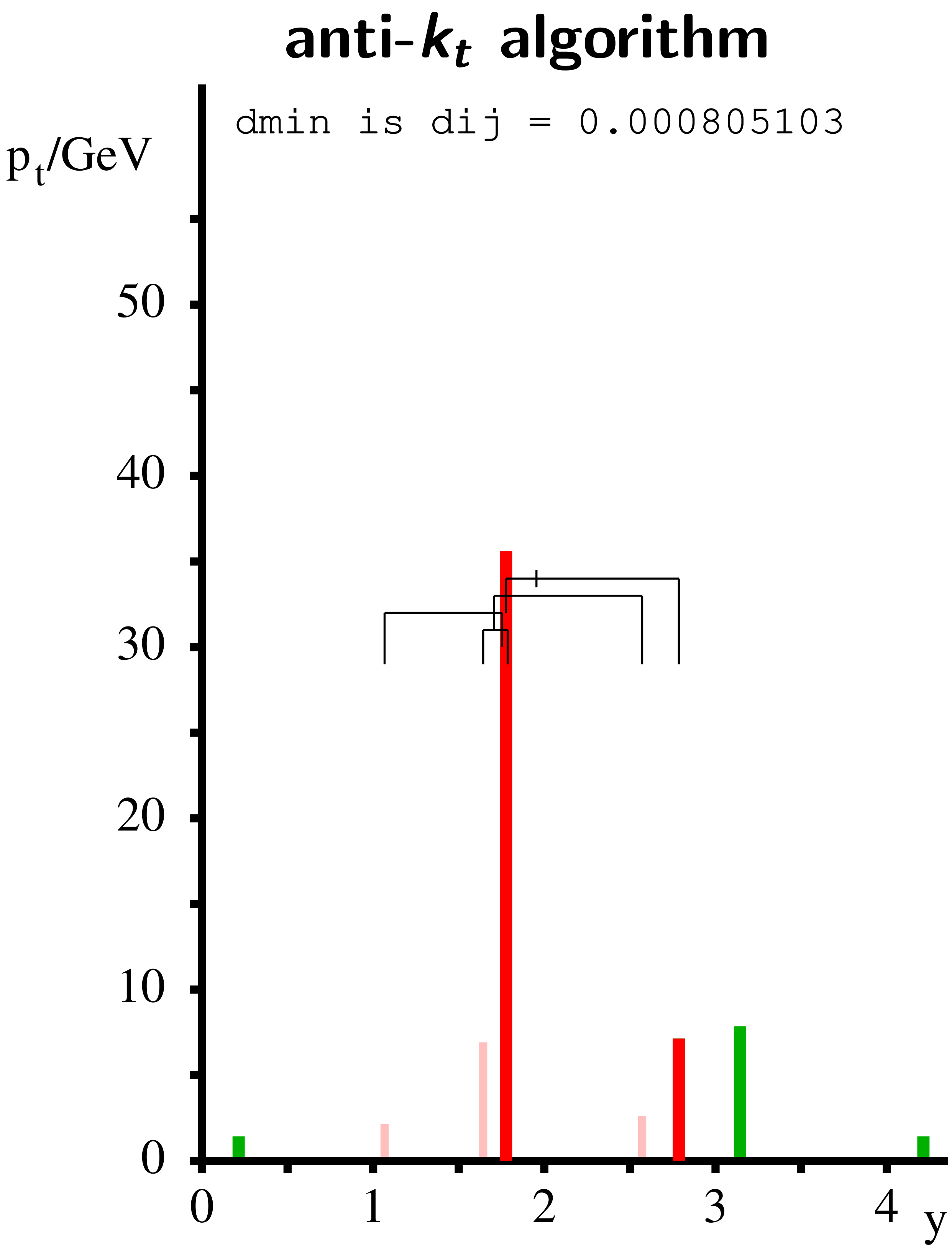
How do different sequential recombination jet algorithms build up the jet?

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here R=2.0
$p_{T,min}$=20 GeV
at LHC: R=0.4 – 1.0]

$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$



How do different sequential recombination jet algorithms build up the jet?

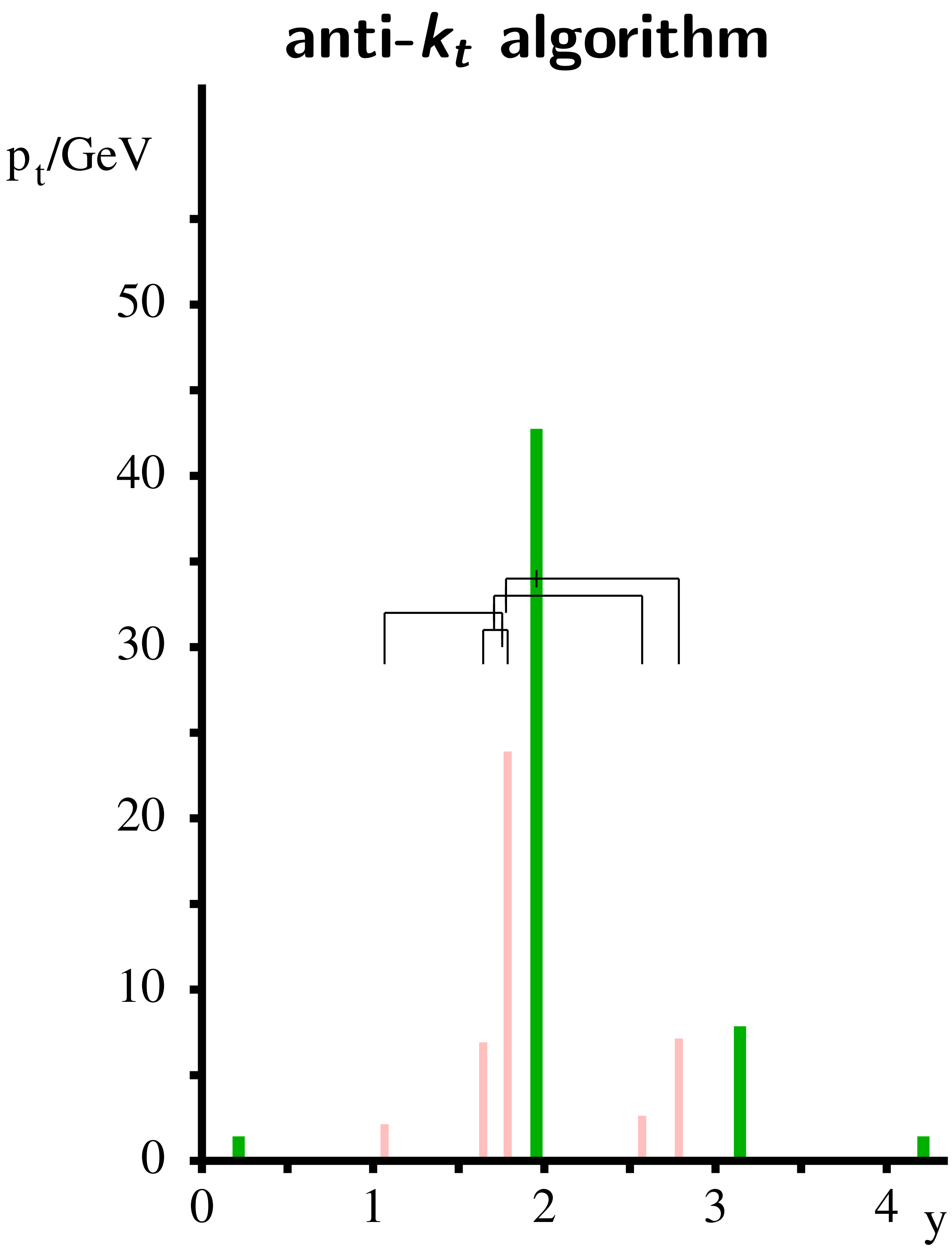
Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R=2.0$
$p_{T,min}=20$ GeV
at LHC: $R=0.4-1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2}\ln\frac{E+p_z}{E-p_z}$$



How do different sequential recombination jet algorithms build up the jet?

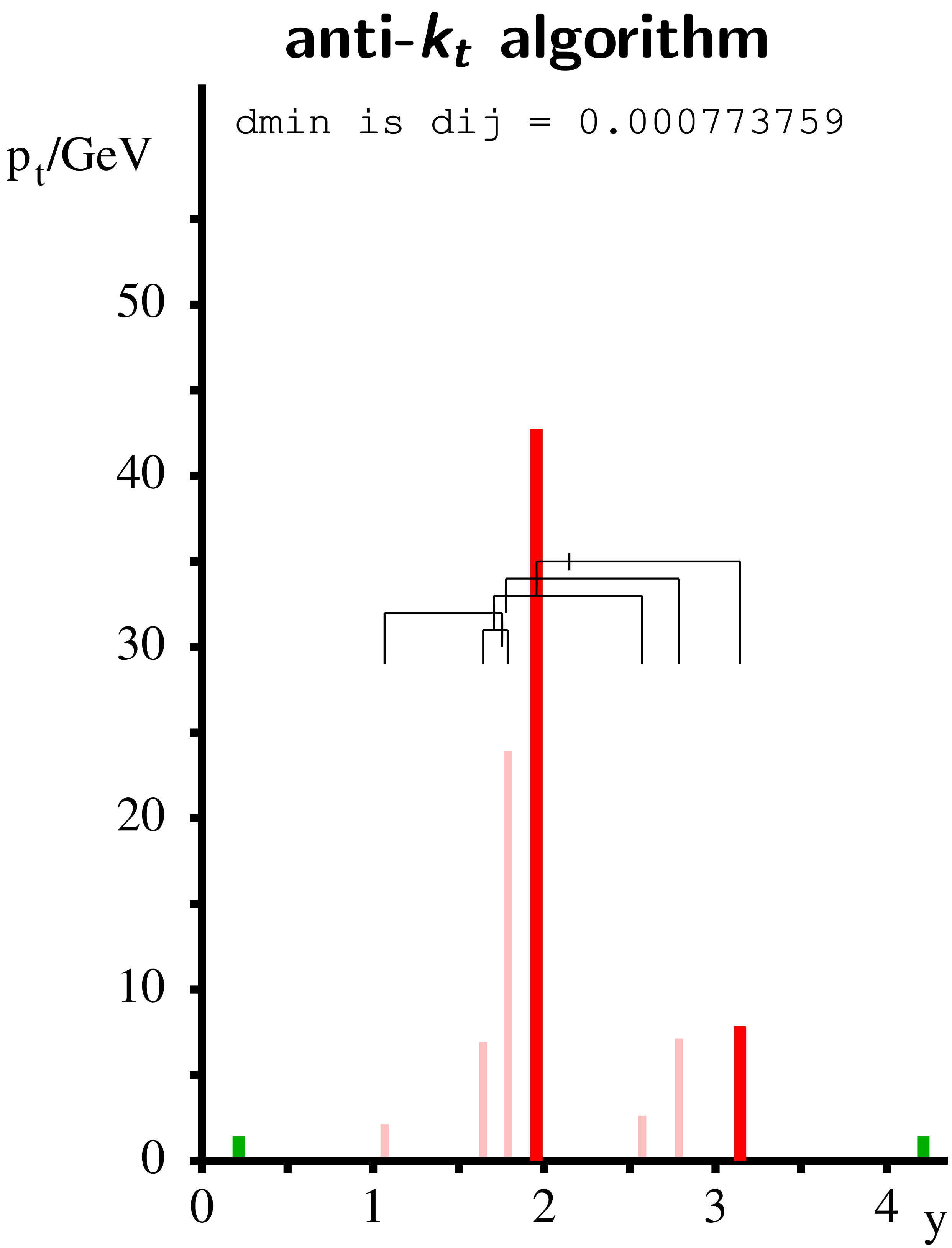
Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

*[here $R = 2.0$
$p_{T,min} = 20$ GeV
at LHC: $R = 0.4 - 1.0$]*

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$



How do different sequential recombination jet algorithms build up the jet?

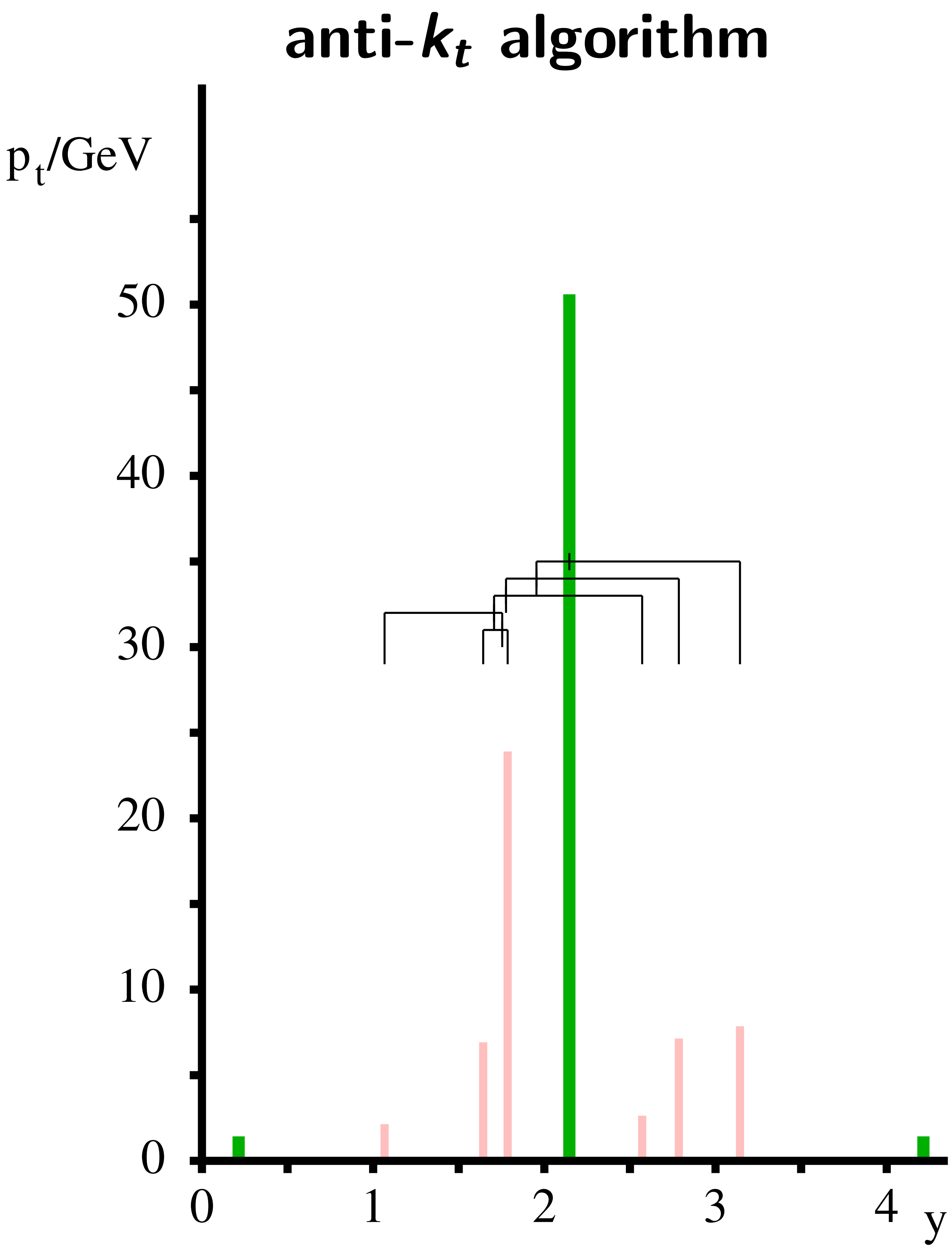
Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here R=2.0
$p_{T,min}$=20 GeV
at LHC: R=0.4 – 1.0]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$



How do different sequential recombination jet algorithms build up the jet?

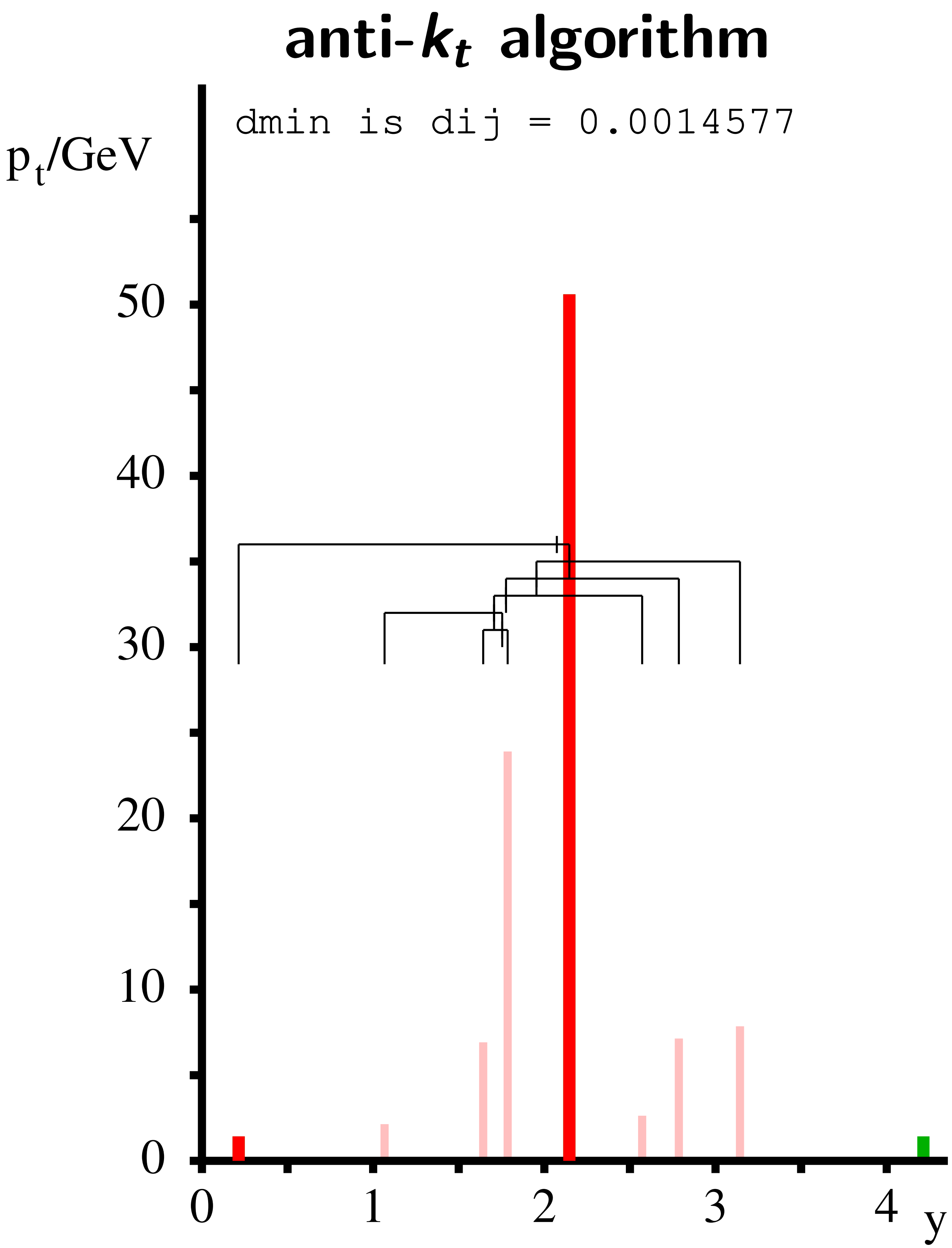
Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R = 2.0$
$p_{T,min} = 20$ GeV
at LHC: $R = 0.4 - 1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$



How do different sequential recombination jet algorithms build up the jet?

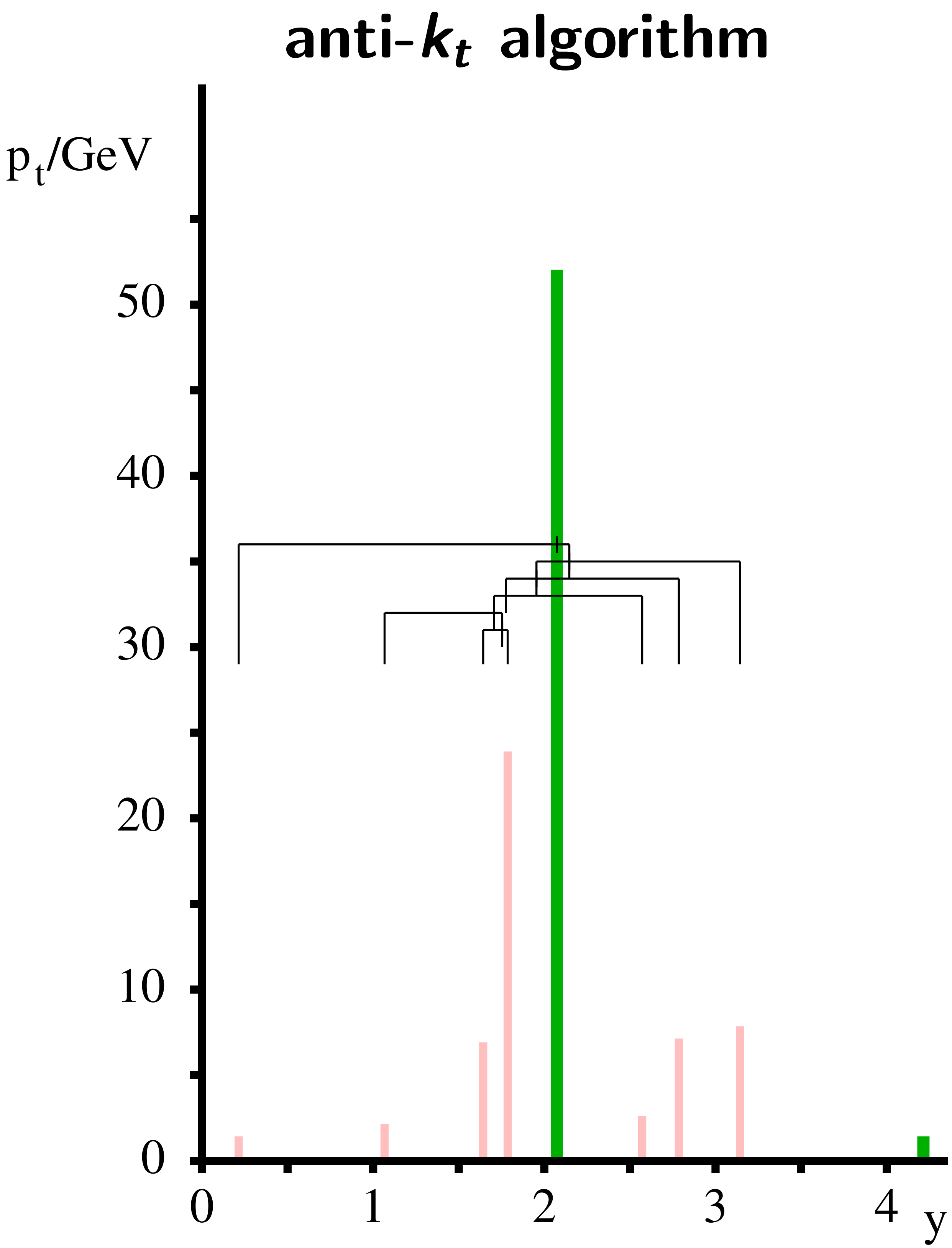
Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here R = 2.0
$p_{T,min}$ = 20 GeV
at LHC: R = 0.4 – 1.0]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$



How do different sequential recombination jet algorithms build up the jet?

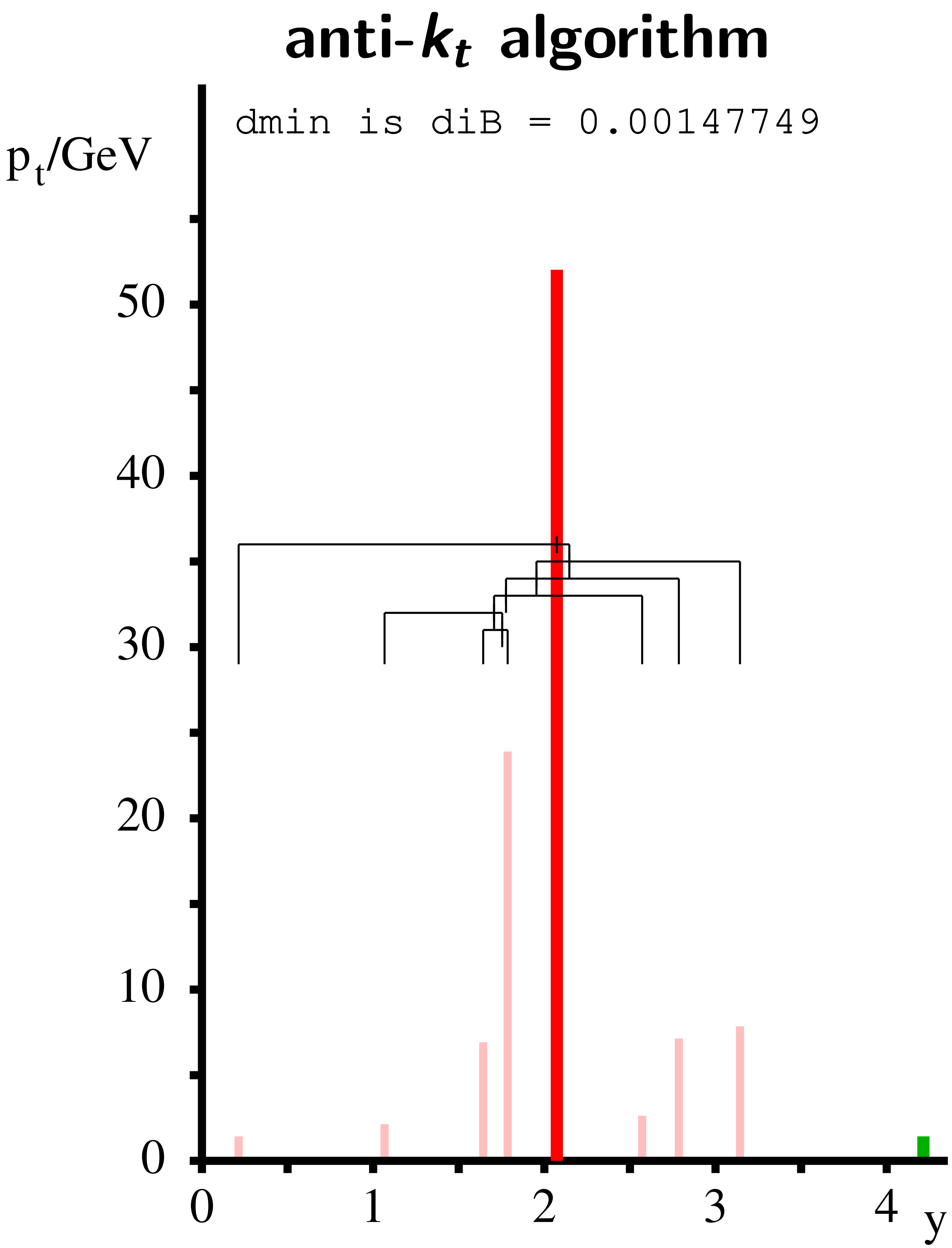
Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R=2.0$
$p_{T,min}=20$ GeV
at LHC: $R=0.4-1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2}\ln\frac{E+p_z}{E-p_z}$$



How do different sequential recombination jet algorithms build up the jet?

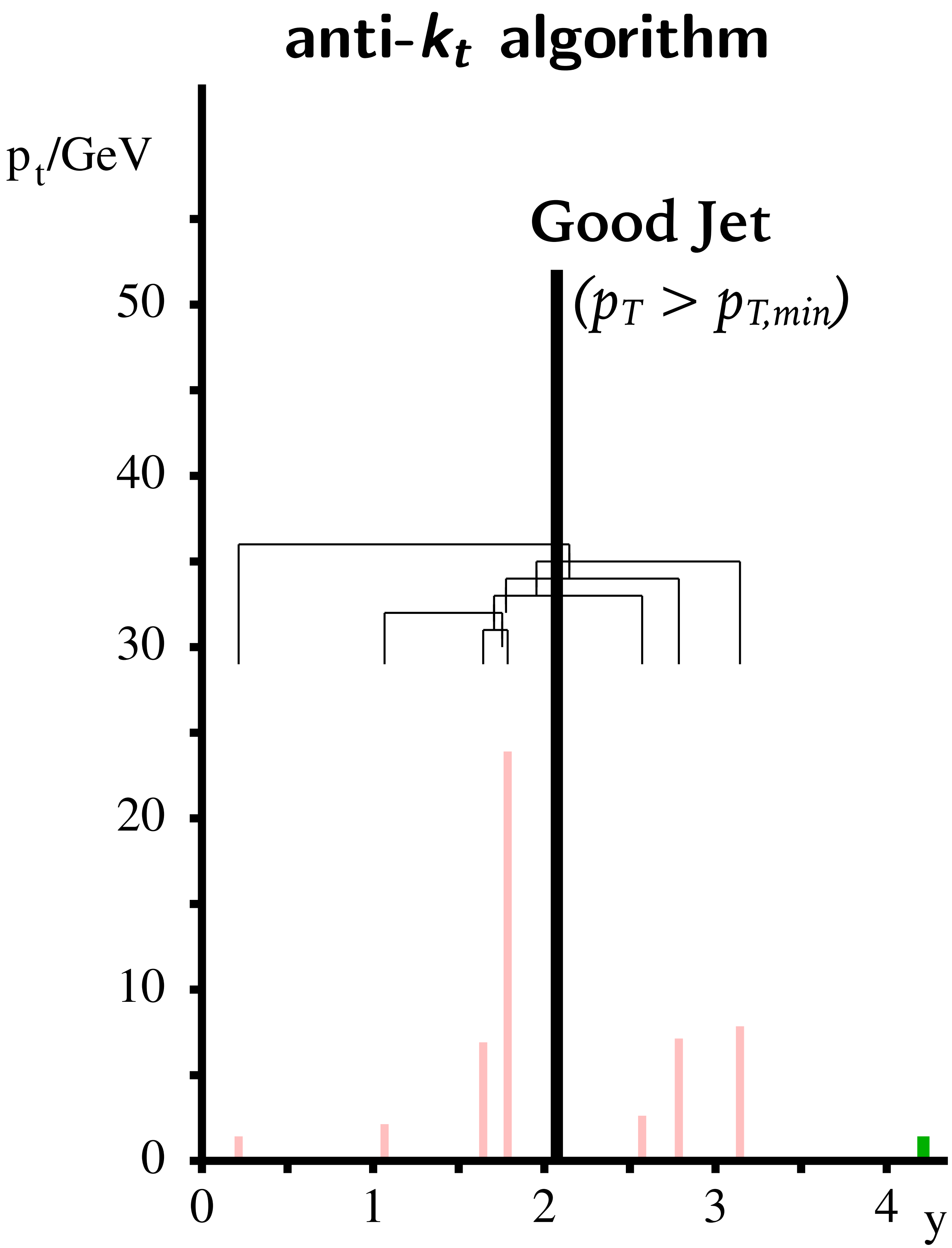
Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R=2.0$
$p_{T,min}=20$ GeV
at LHC: $R=0.4-1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2}\ln\frac{E+p_z}{E-p_z}$$



How do different sequential recombination jet algorithms build up the jet?

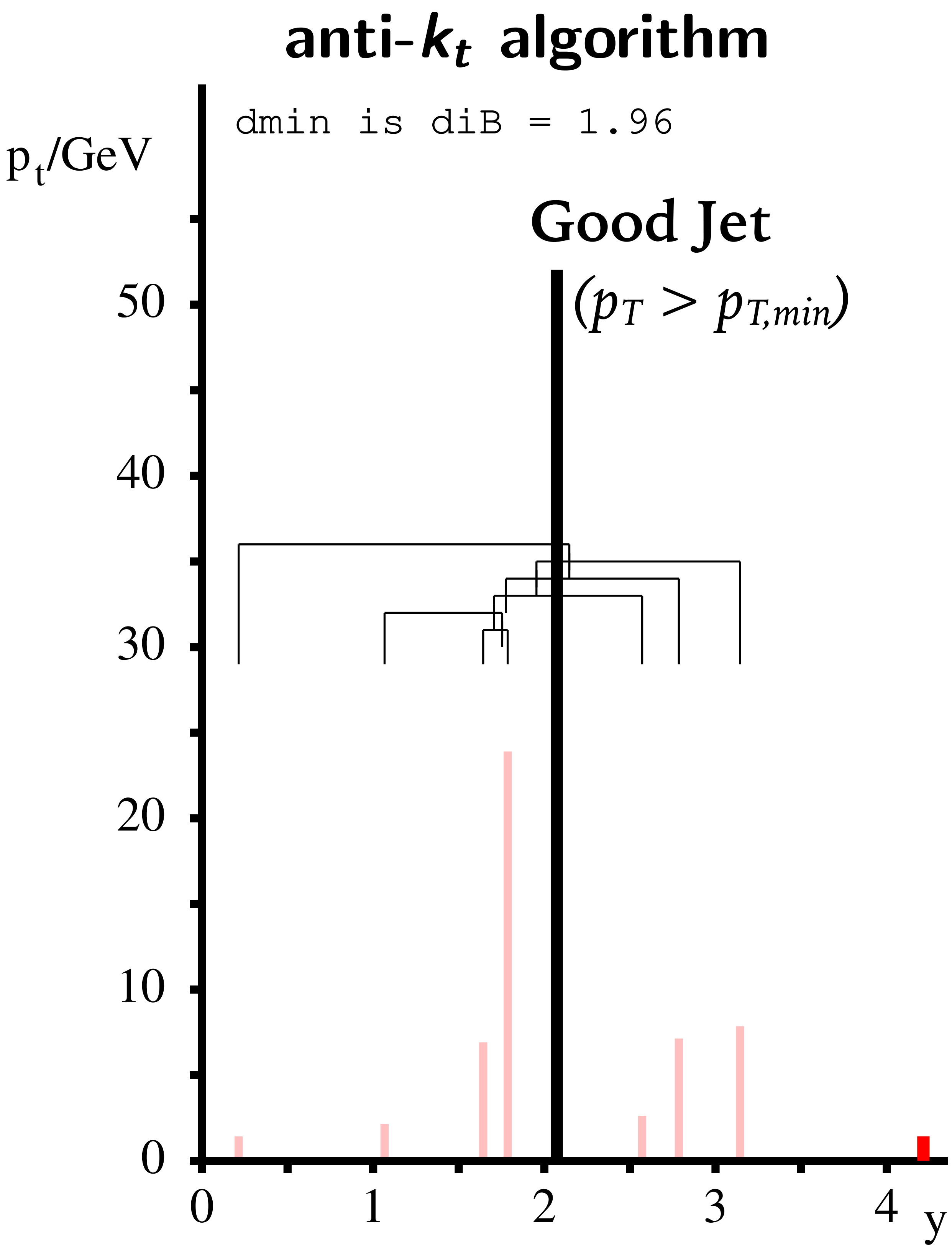
Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R = 2.0$
$p_{T,min} = 20$ GeV
at LHC: $R = 0.4 - 1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$



How do different sequential recombination jet algorithms build up the jet?

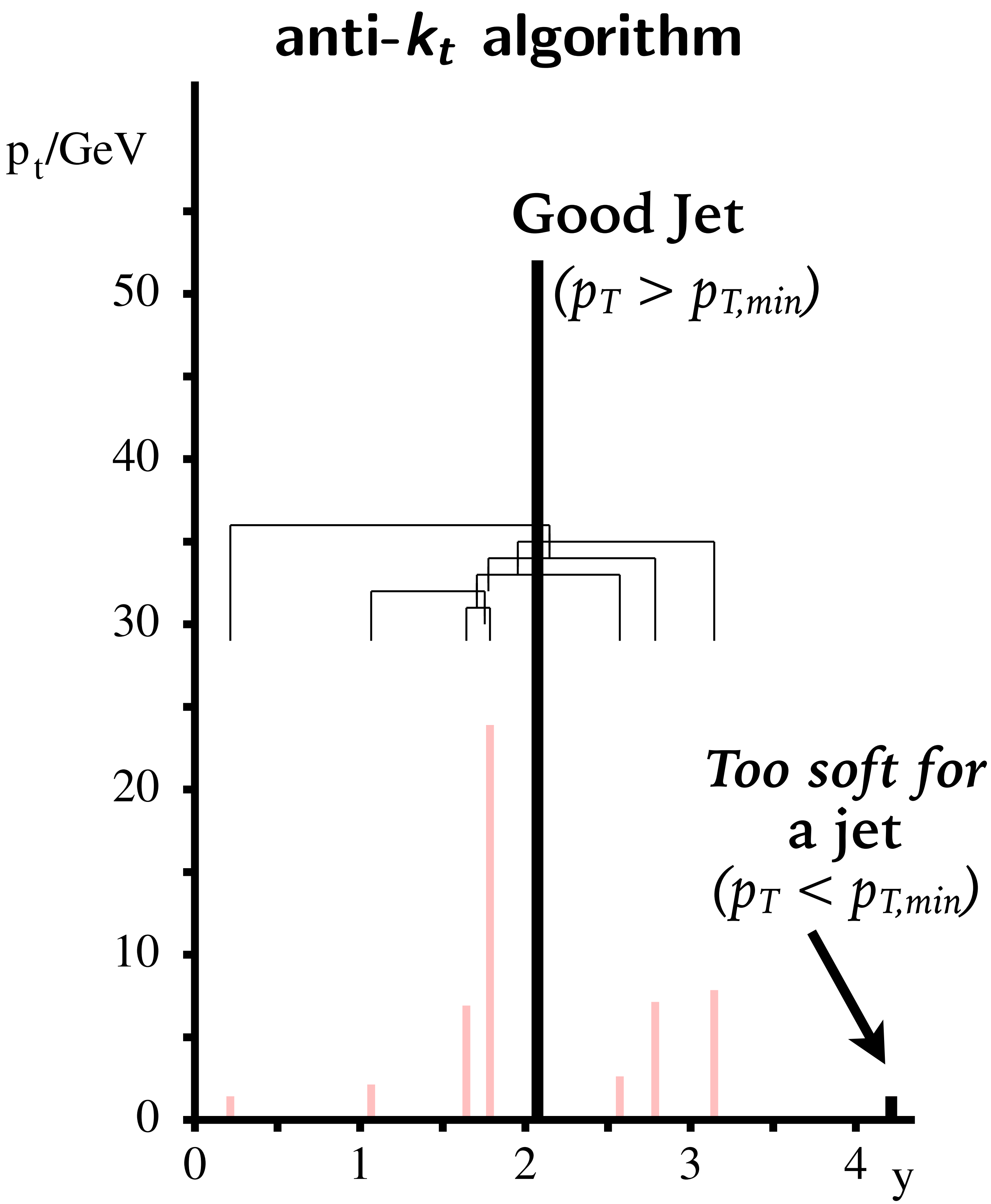
Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R = 2.0$
$p_{T,min} = 20$ GeV
at LHC: $R = 0.4 - 1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$



How do different sequential recombination jet algorithms build up the jet?

Anti-k_t gradually makes its way through the secondary blob: only hierarchy in clustering is distance from hard core.

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}$$

$$d_{iB} = \frac{1}{p_{ti}^2}$$

[here $R = 2.0$
$p_{T,min} = 20$ GeV
at LHC: $R = 0.4 - 1.0$]

$$\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2 \qquad y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$

anti-k_t in action [full simulated event]

Clustering grows around hard cores

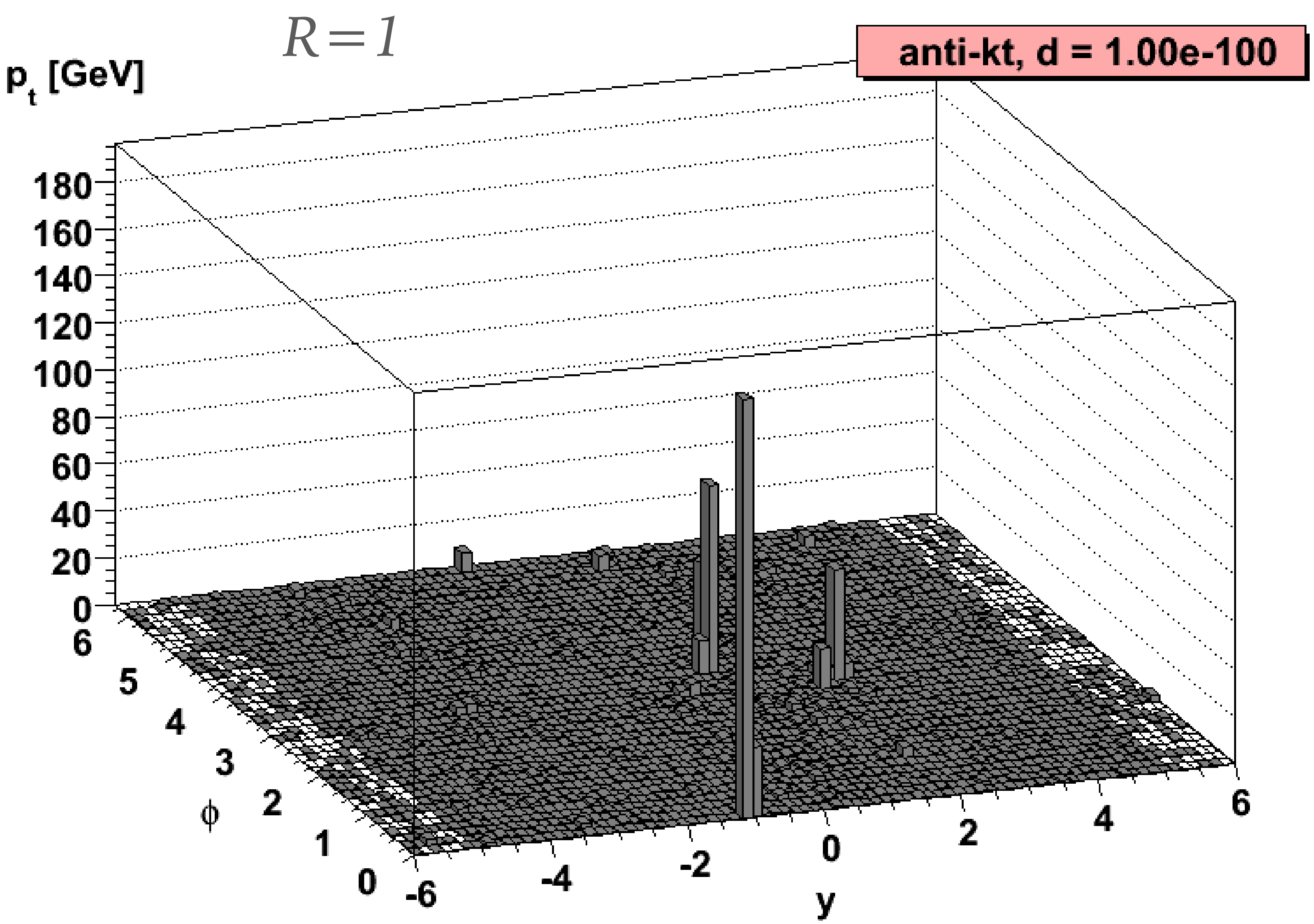
$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$

$R = 1$

anti-k_t in action [full simulated event]

Clustering grows around hard cores

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \quad d_{iB} = \frac{1}{p_{ti}^2}$$

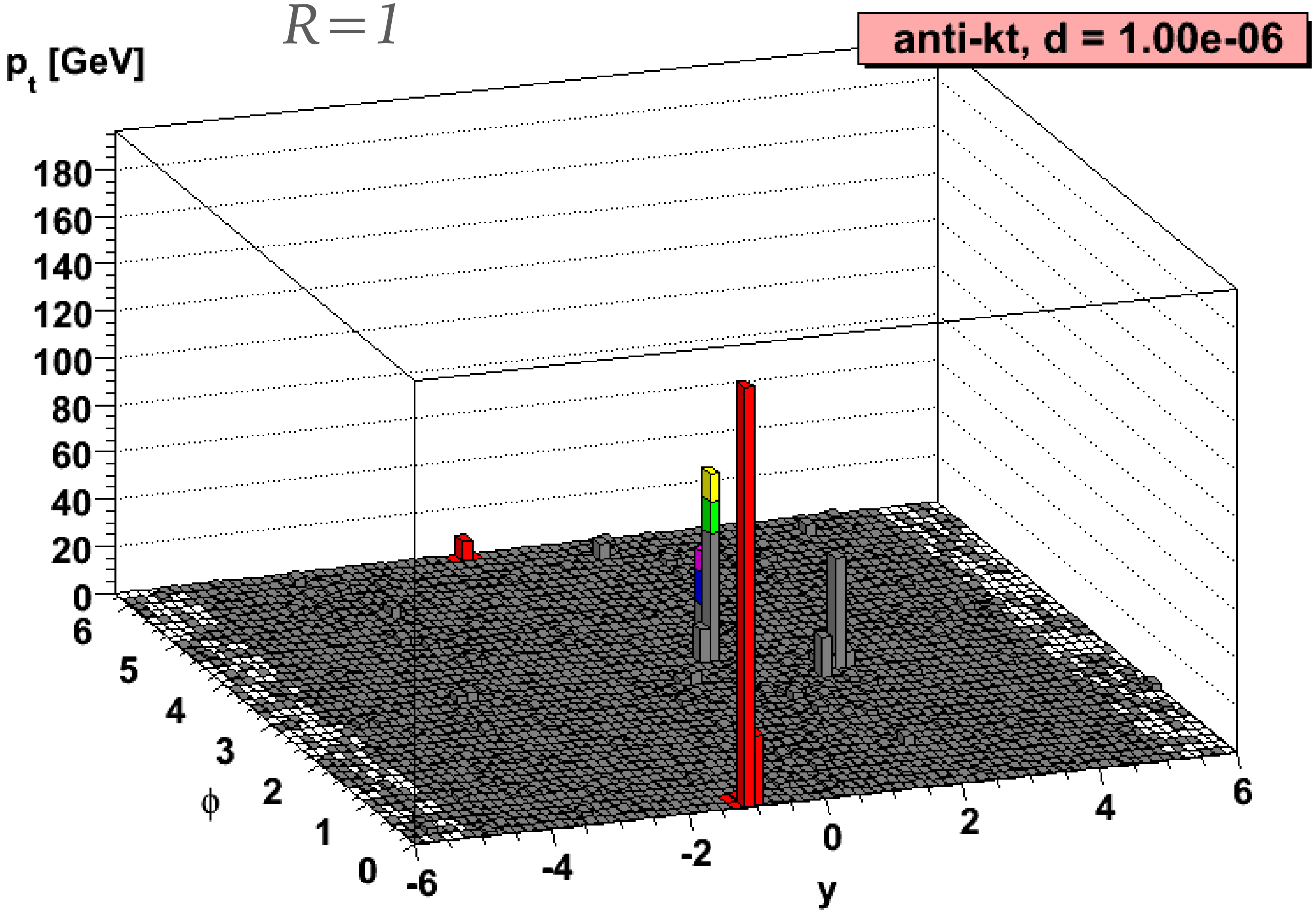


anti-k_t in action [full simulated event]

Clustering grows around hard cores

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$

$R = 1$

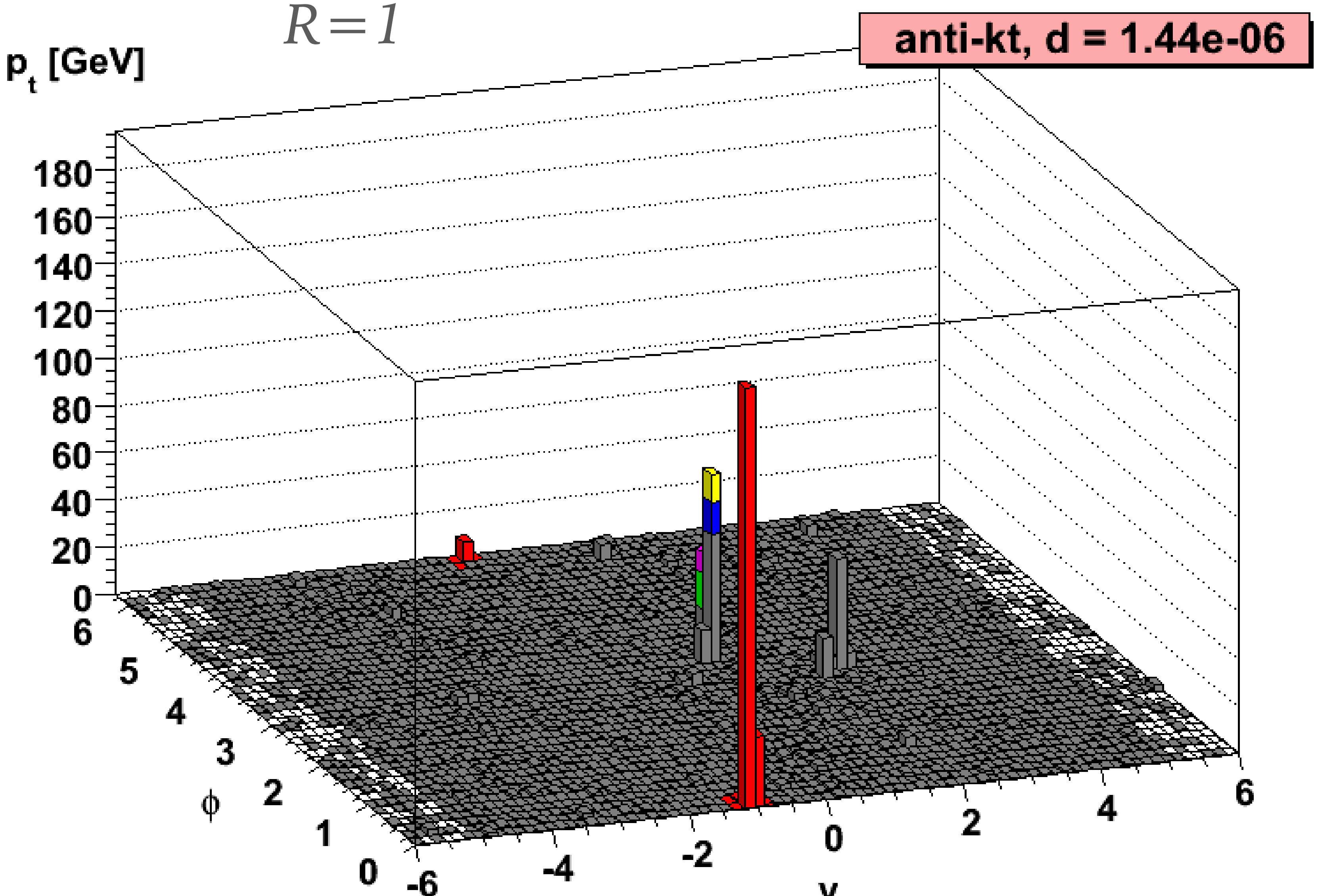


anti-k_t in action [full simulated event]

Clustering grows around hard cores

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$

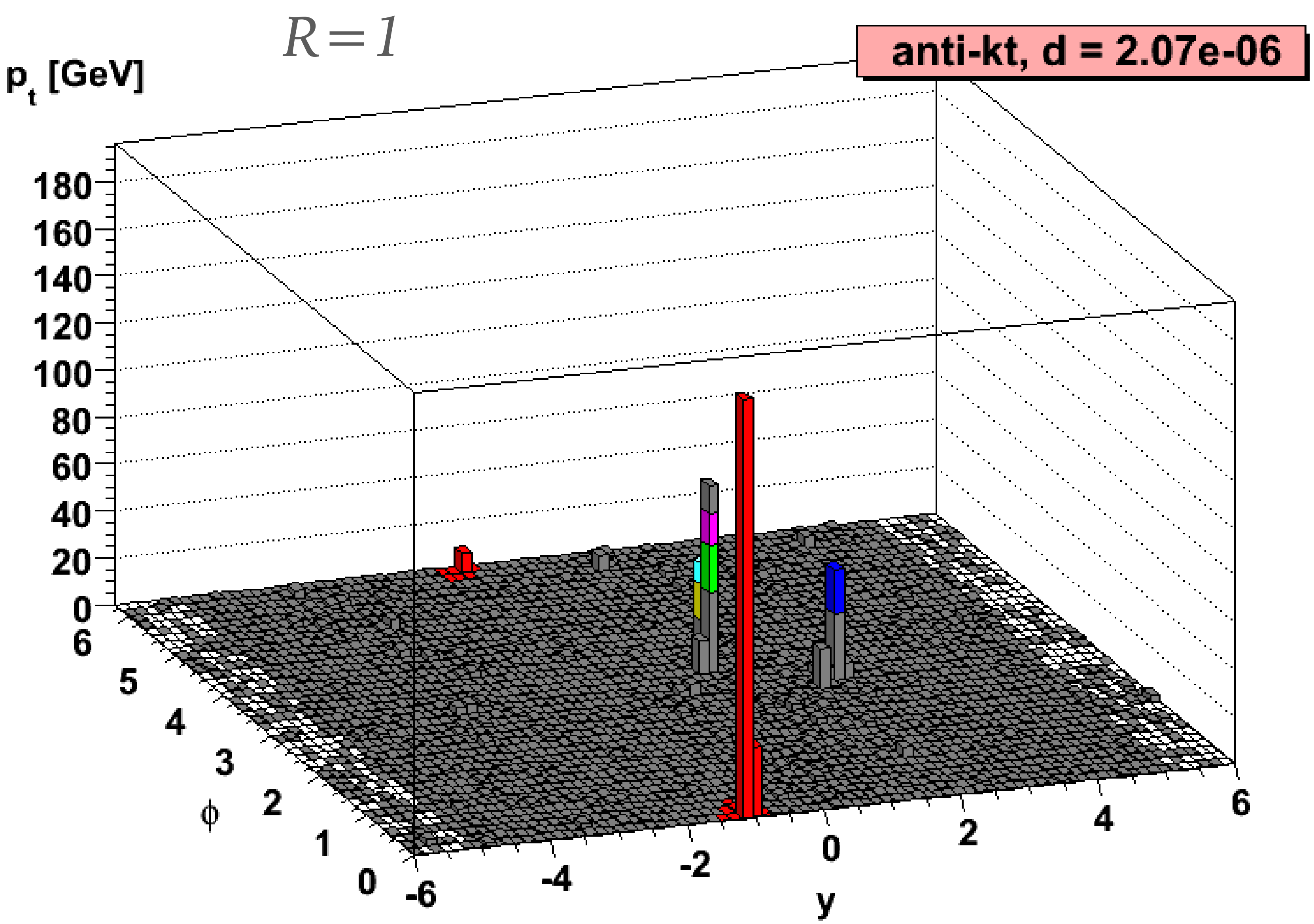
$R = 1$



anti-k_t in action [full simulated event]

Clustering grows around hard cores

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \quad d_{iB} = \frac{1}{p_{ti}^2}$$

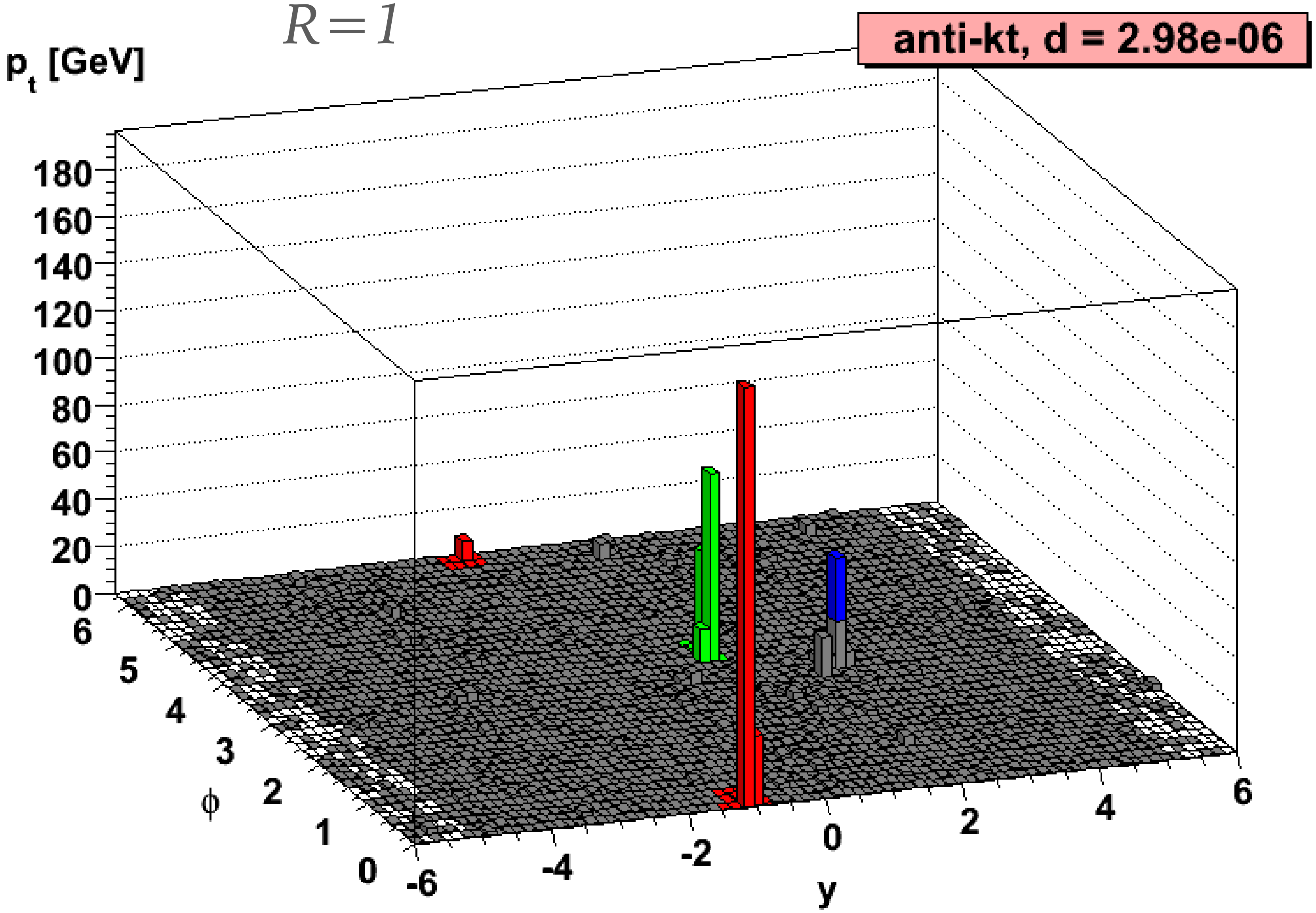


anti-k_t in action [full simulated event]

Clustering grows around hard cores

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$

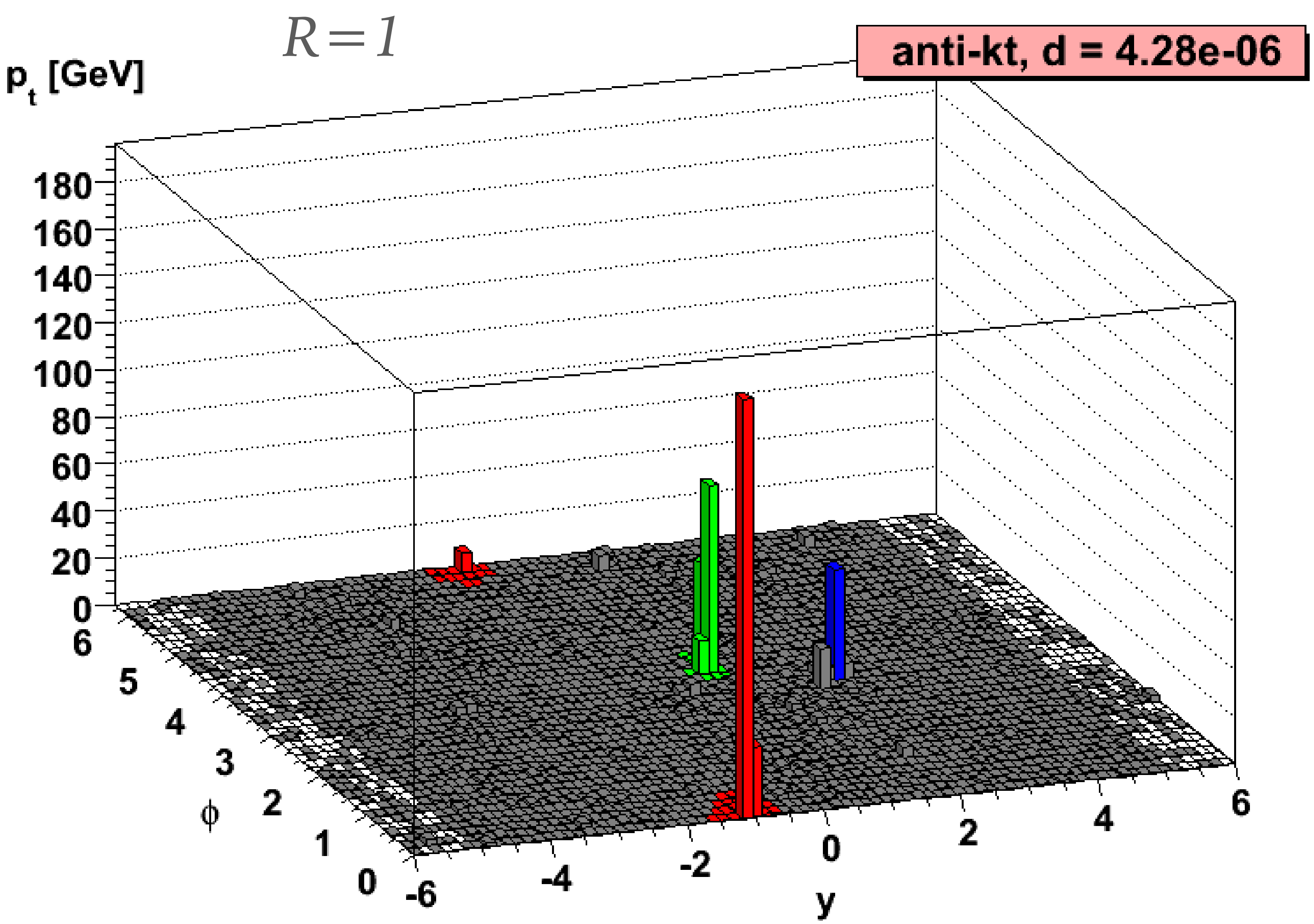
$R = 1$



anti-k_t in action [full simulated event]

Clustering grows around hard cores

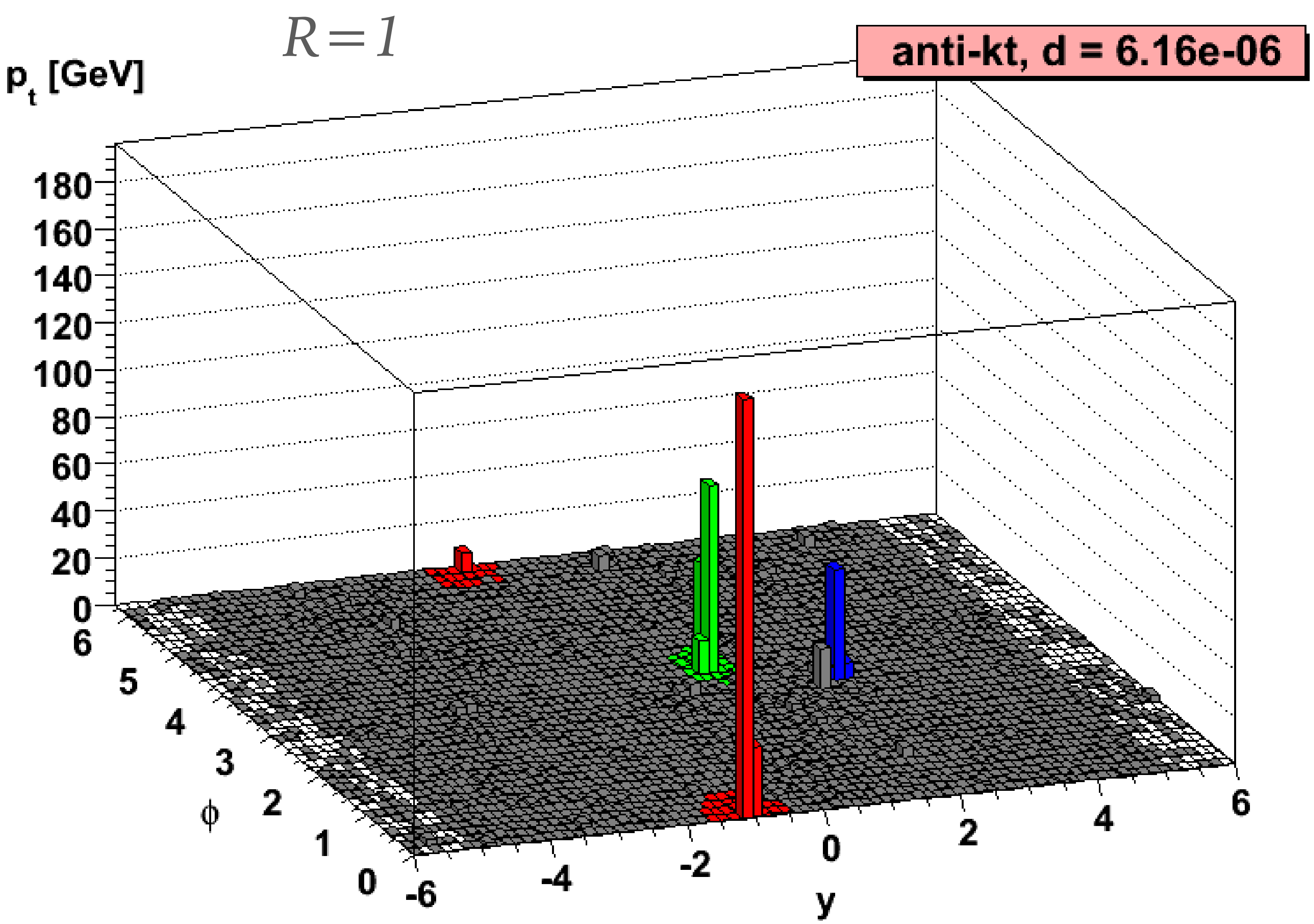
$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$



anti-k_t in action [full simulated event]

Clustering grows around hard cores

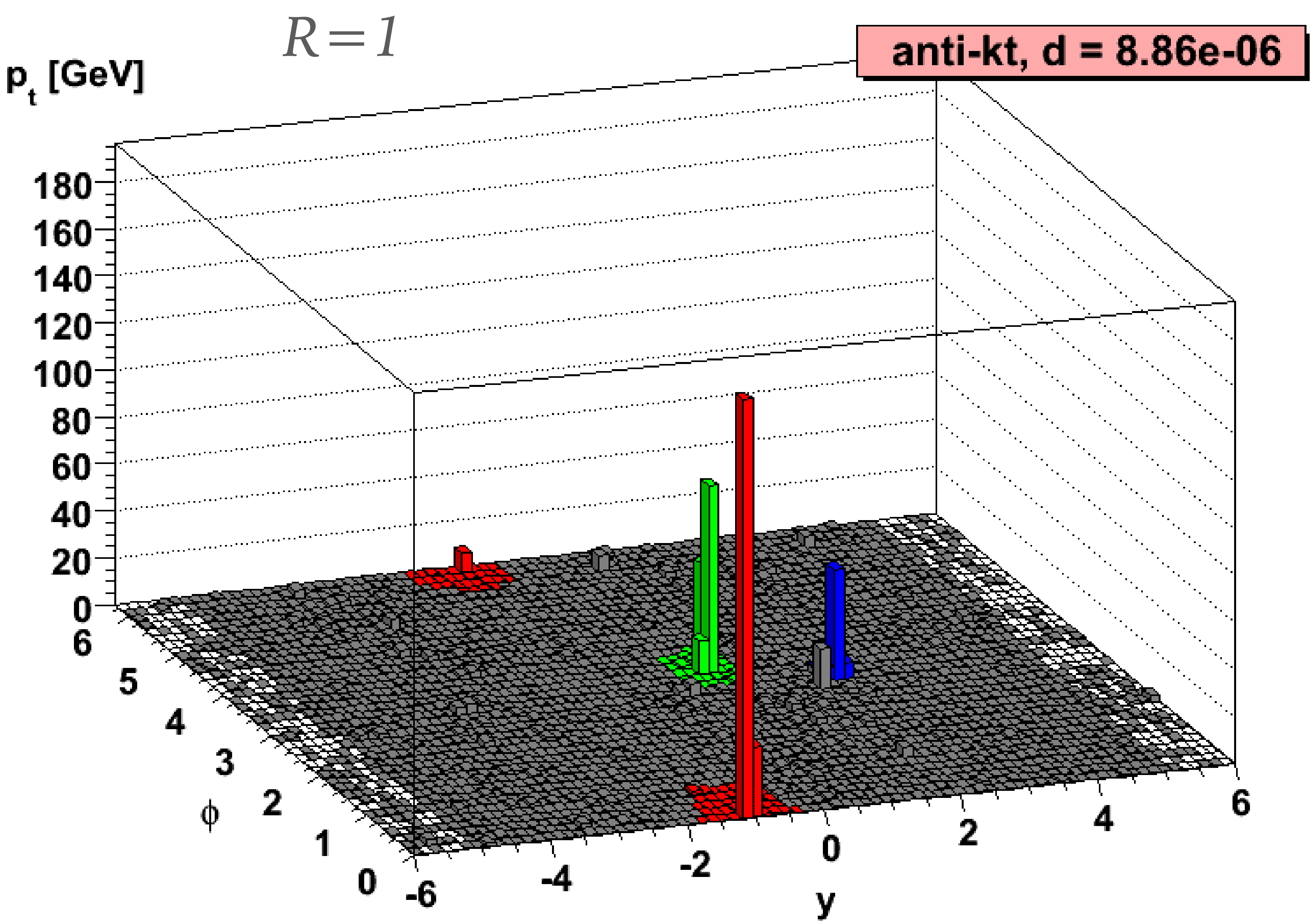
$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \quad d_{iB} = \frac{1}{p_{ti}^2}$$



anti-k_t in action [full simulated event]

Clustering grows around hard cores

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \quad d_{iB} = \frac{1}{p_{ti}^2}$$

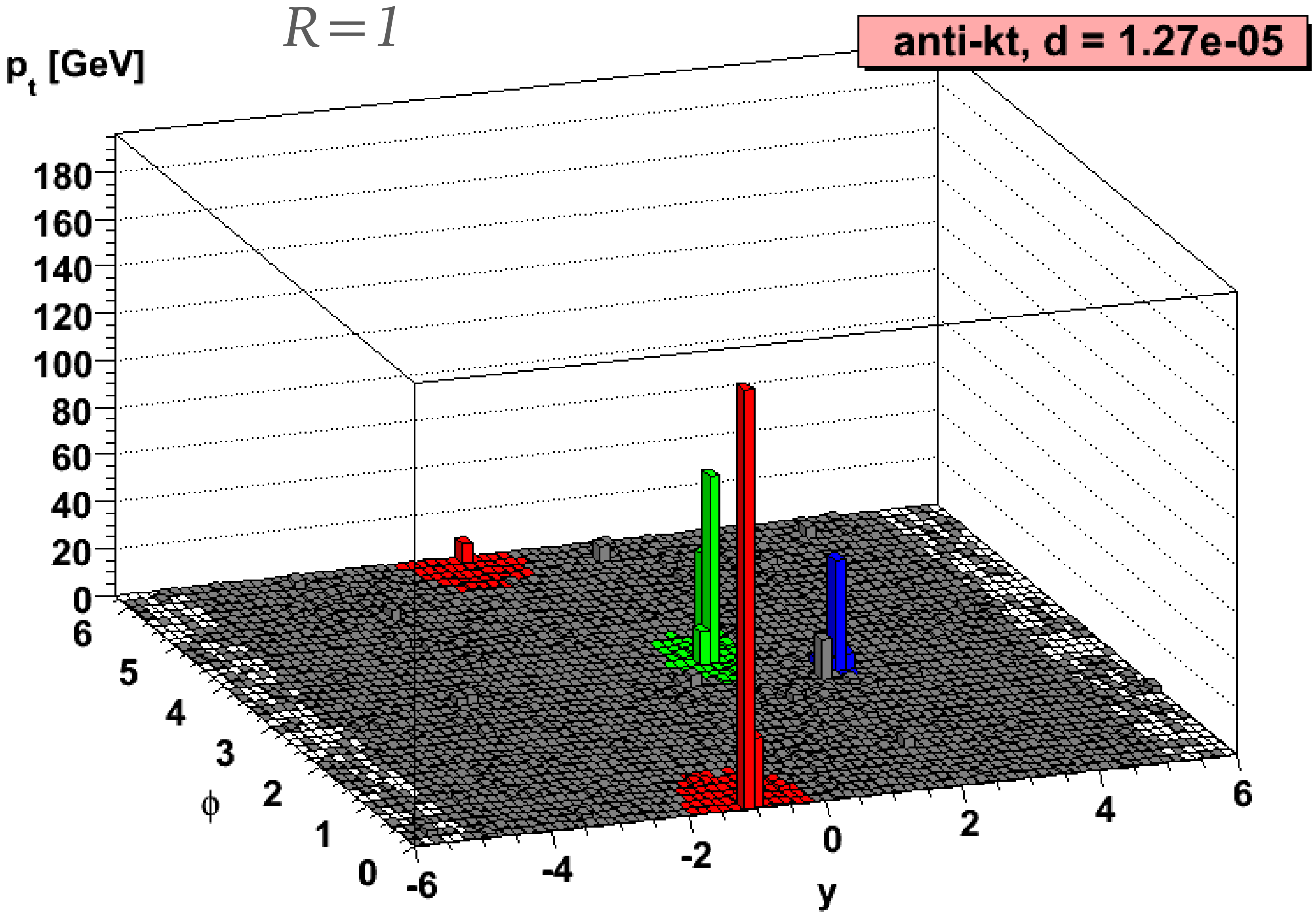


anti-k_t in action [full simulated event]

Clustering grows around hard cores

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$

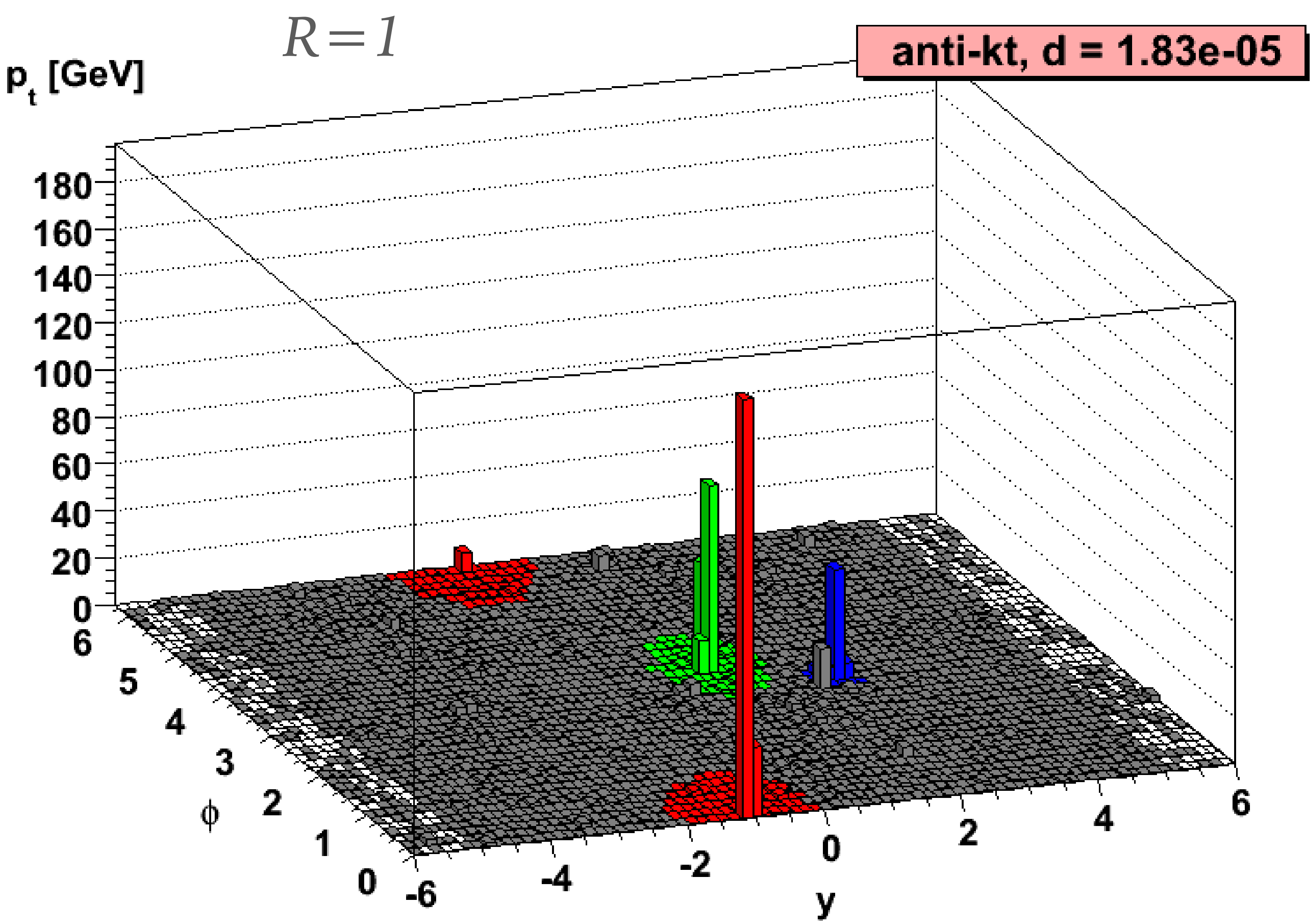
$R=1$



anti-k_t in action [full simulated event]

Clustering grows around hard cores

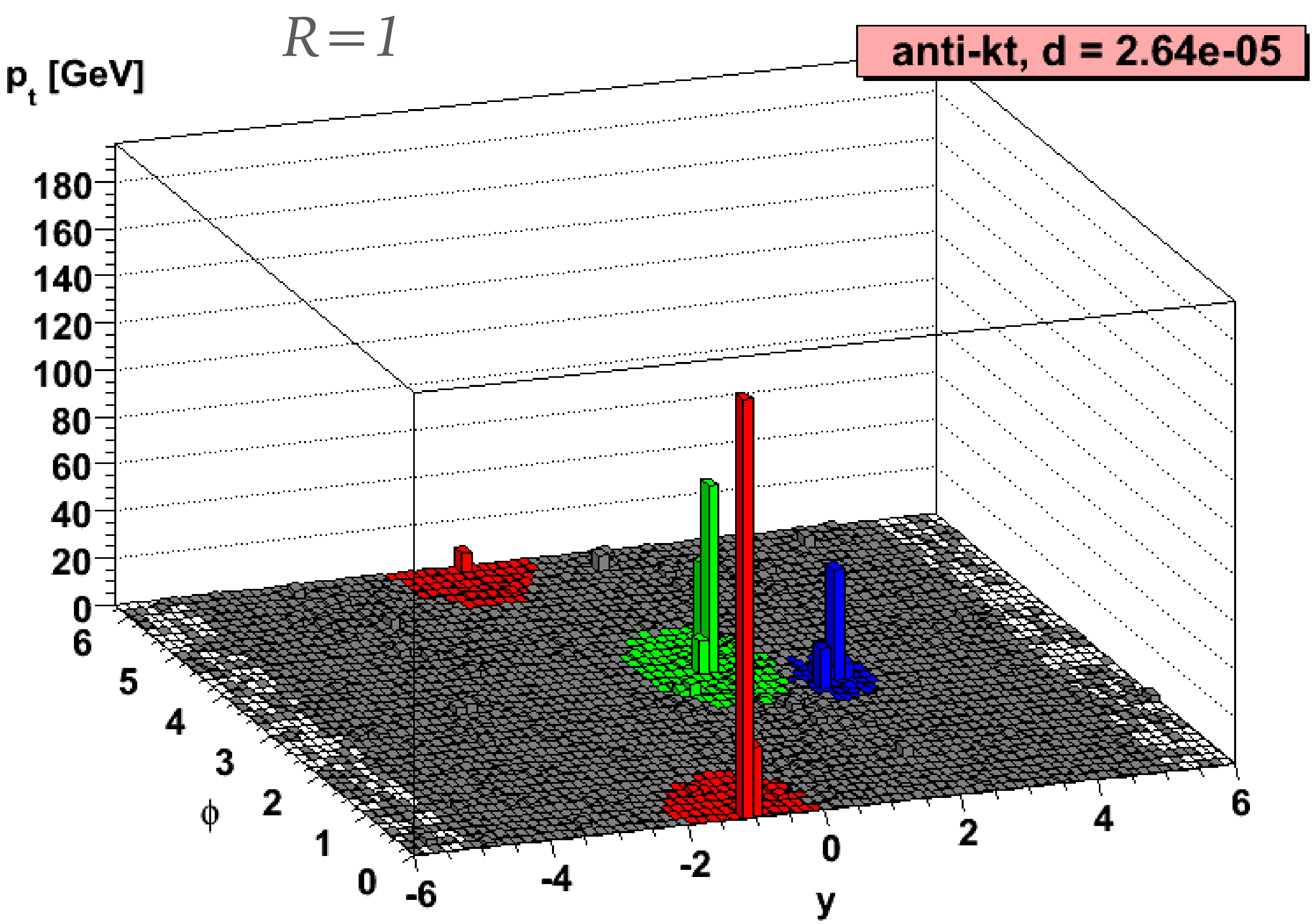
$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$



anti-k_t in action [full simulated event]

Clustering grows around hard cores

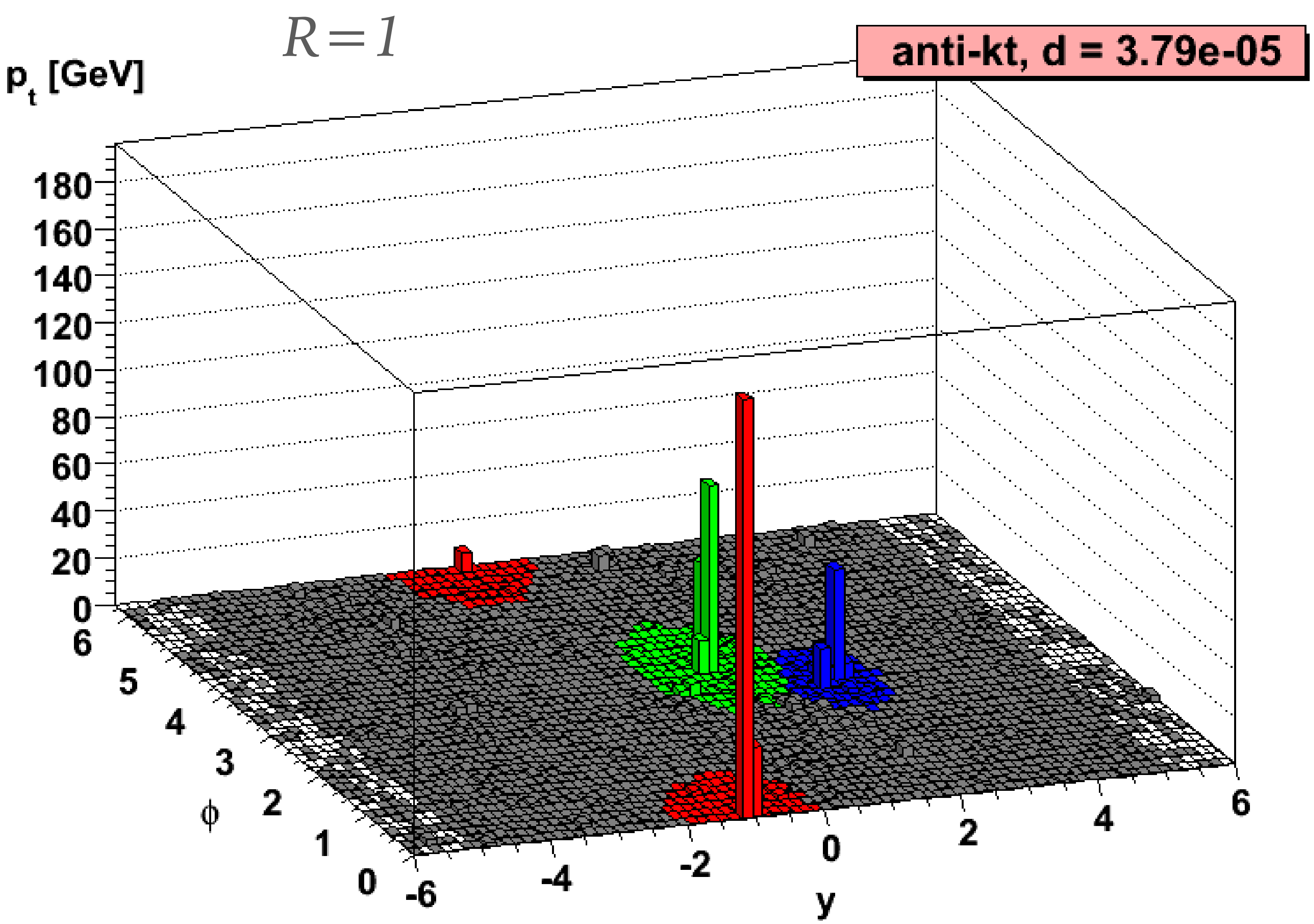
$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$



anti-k_t in action [full simulated event]

Clustering grows around hard cores

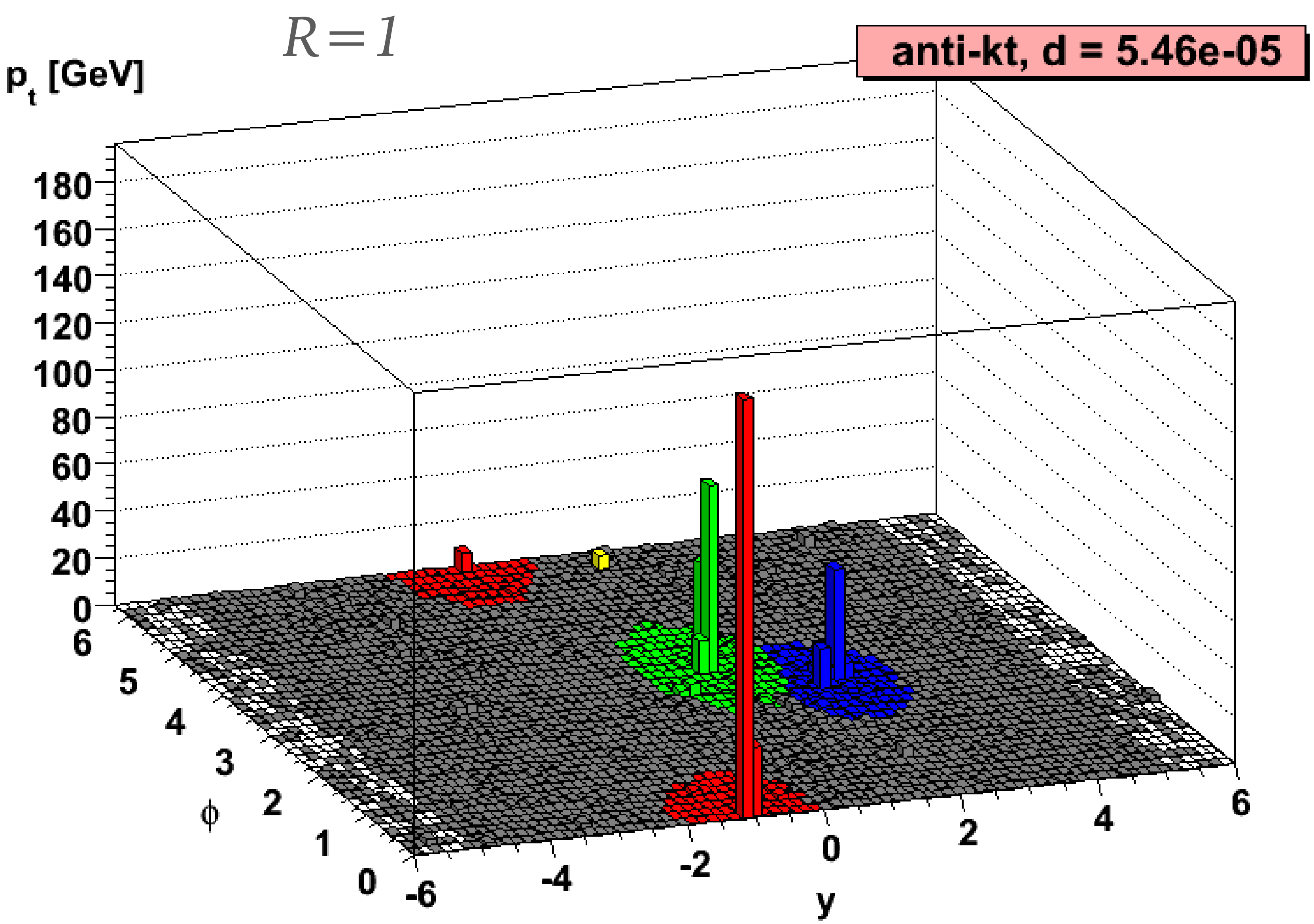
$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$



anti-k_t in action [full simulated event]

Clustering grows around hard cores

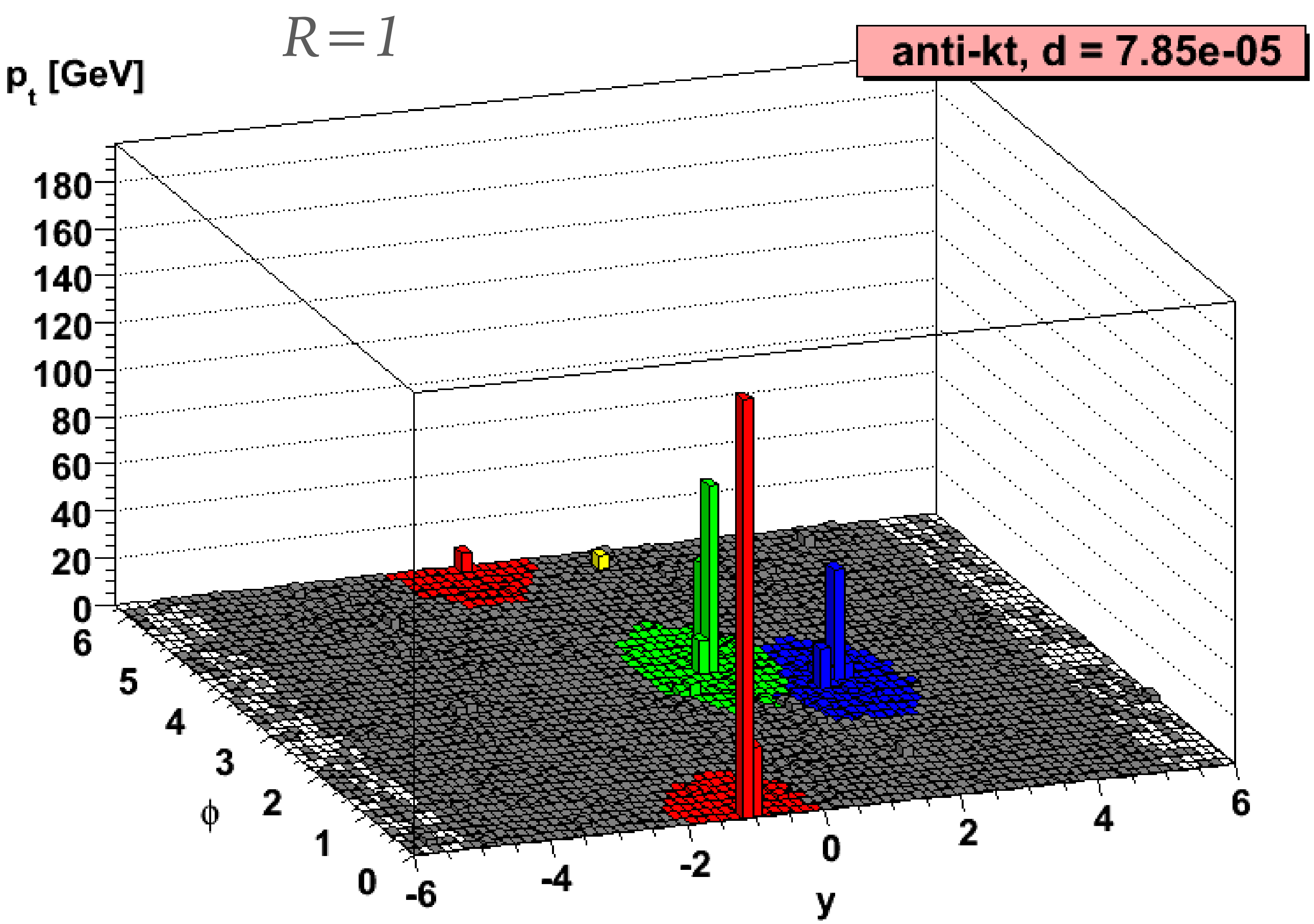
$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$



anti-k$_t$ in action [full simulated event]

Clustering grows around hard cores

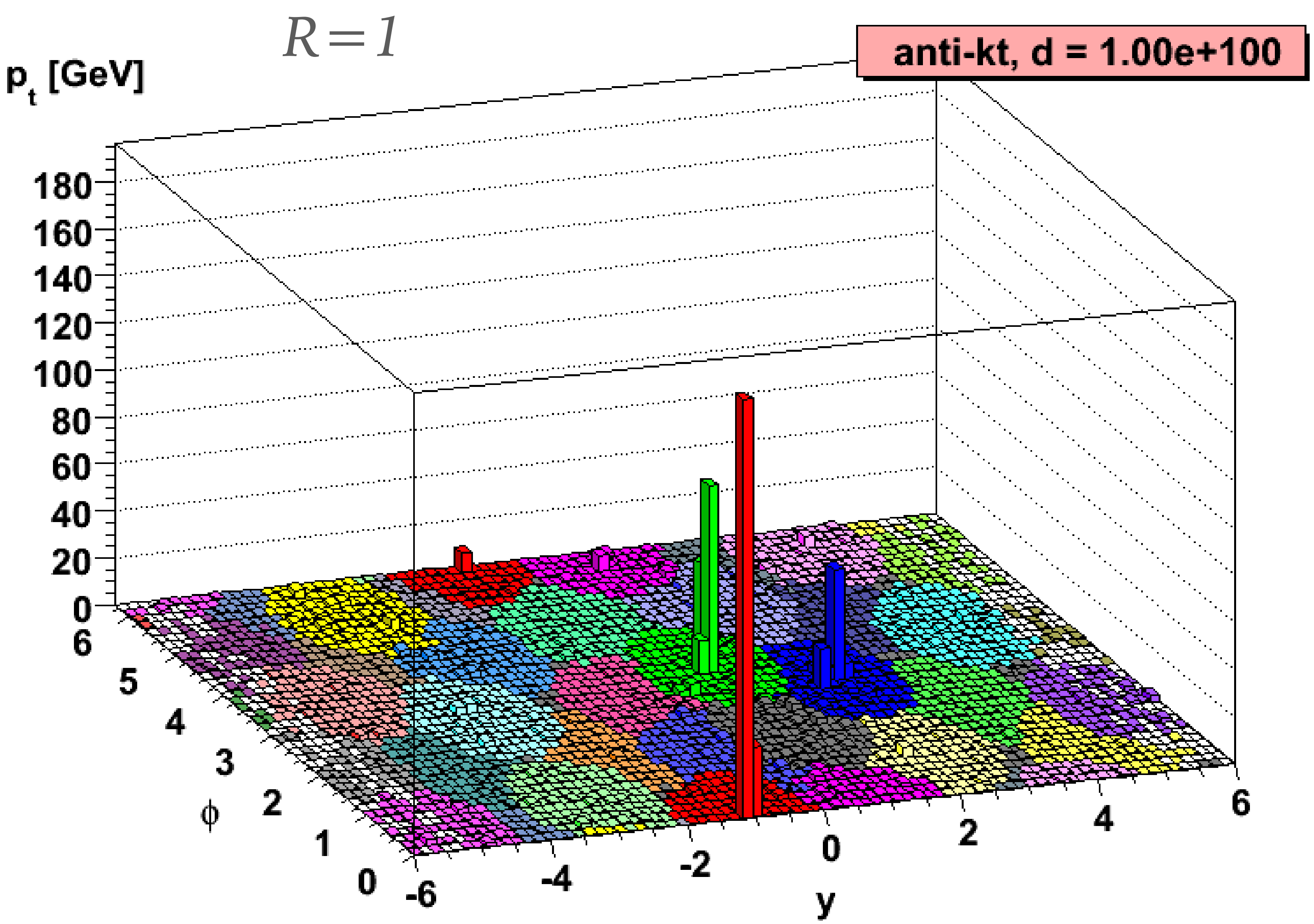
$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \qquad d_{iB} = \frac{1}{p_{ti}^2}$$



anti-k_t in action [full simulated event]

Clustering grows around hard cores

$$d_{ij} = \frac{1}{\max(p_{ti}^2, p_{tj}^2)} \frac{\Delta R_{ij}^2}{R^2}, \quad d_{iB} = \frac{1}{p_{ti}^2}$$



Anti-k_t gives circular jets ("cone-like") in a way that's infrared safe

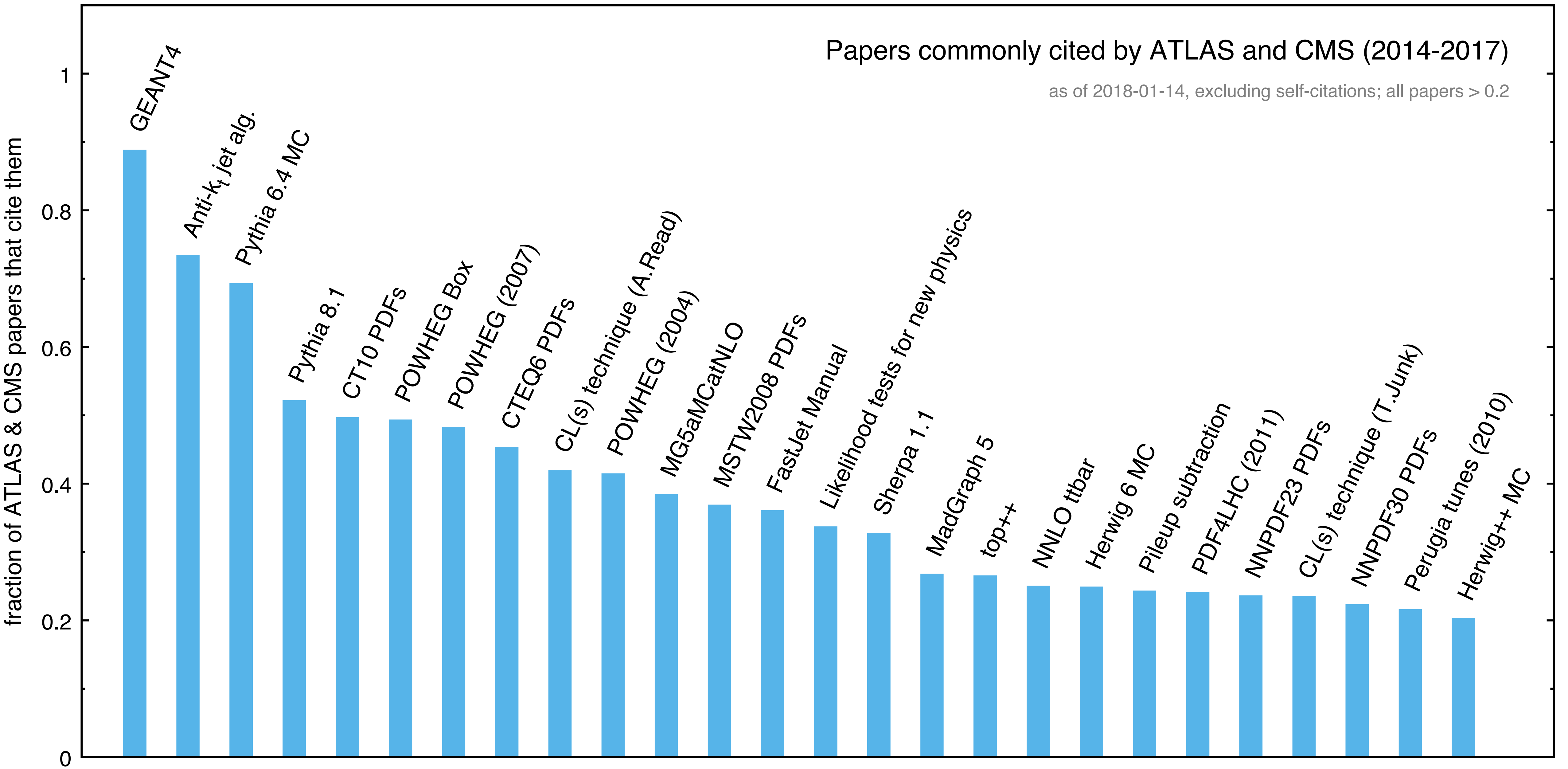
conclusions

3 Signal and background models

The ggF and VBF production modes for $H \to WW^*$ are modelled at next-to-leading order (NLO) in the strong coupling α_S with the Powheg MC generator [22–25], interfaced with Pythia8 [26] (version 8.165) for the parton shower, hadronisation, and underlying event. The CT10 [27] PDF set is used and the parameters of the Pythia8 generator controlling the modelling of the parton shower and the underlying event are those corresponding to the AU2 set [28]. The Higgs boson mass set in the generation is 125.0 GeV, which is close to the measured value. The Powheg ggF model takes into account finite quark masses and a running-width Breit–Wigner distribution that includes electroweak corrections at NLO [29]. To improve the modelling of the Higgs boson p_T distribution, a reweighting scheme is applied to reproduce the prediction of the next-to-next-to-leading-order (NNLO) and next-to-next-to-leading-logarithm (NNLL) dynamic-scale calculation given by the HRes 2.1 program [30]. Events with ≥ 2 jets are further reweighted to reproduce the p_T^H spectrum predicted by the NLO Powheg simulation of Higgs boson production in association with two jets ($H + 2$ jets) [31]. Interference with continuum WW production [32, 33] has a negligible impact on this analysis due to the transverse-mass selection criteria described in Section 4 and is not included in the signal model.

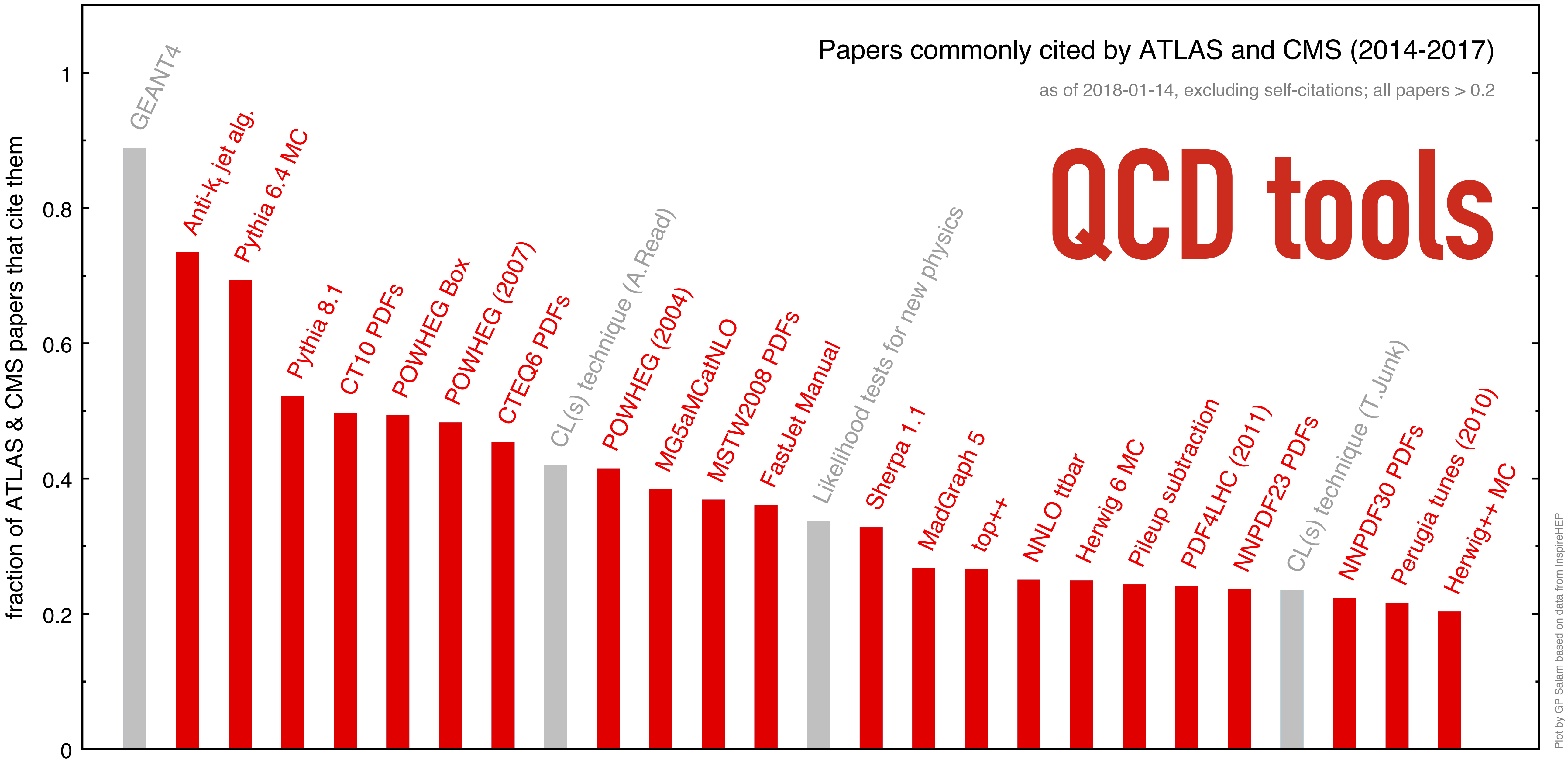
Jets are reconstructed from topological clusters of calorimeter cells [50–52] using the anti-k_t algorithm with a radius parameter of $R = 0.4$ [53]. Jet energies are corrected for the effects of calorimeter non-

WHAT DO ATLAS & CMS USE MOST FREQUENTLY?



WHAT DO ATLAS & CMS USE MOST FREQUENTLY?

in the last 2 lectures we've seen a good number of the tools used at LHC



CONCLUSIONS

- A huge number of ingredients goes into hadron-collider predictions and studies (α_s, PDFs, matrix elements, resummation, parton showers, non-perturbative models, jet algorithms, etc.)
- a key idea is the separation of (time) scales, "factorisation"
  - short timescales: the hard process
  - long timescales: hadronic physics
  - in between: parton showers, resummation, DGLAP
- as long as you ask the right questions (e.g. look at jets, not individual hadrons), you can exploit this separation for quantitative, accurate, collider physics
- maximising accuracy and information extracted is today's research frontier

Extra resources

Introductory level

QCD lecture notes from CERN schools, e.g.

- Peter Skands, arXiv:1207.2389
- GPS, arXiv:1011.5131

More advanced

Slides from QCD and Monte Carlo specific schools

- CTEQ schools: https://www.physics.smu.edu/scalise/cteq/#Summer
- MCNet schools: http://www.montecarlonet.org/index.php?p=MCSchool/main&sub=MCSchools

Books

- QCD and collider physics, Ellis, Stirling & Webber
- The Black Book of Quantum Chromodynamics, Campbell, Huston & Krauss

EXTRA SLIDES

It turns out you can calculate the gluon multiplicity analytically, by summing all orders (n) of perturbation theory:

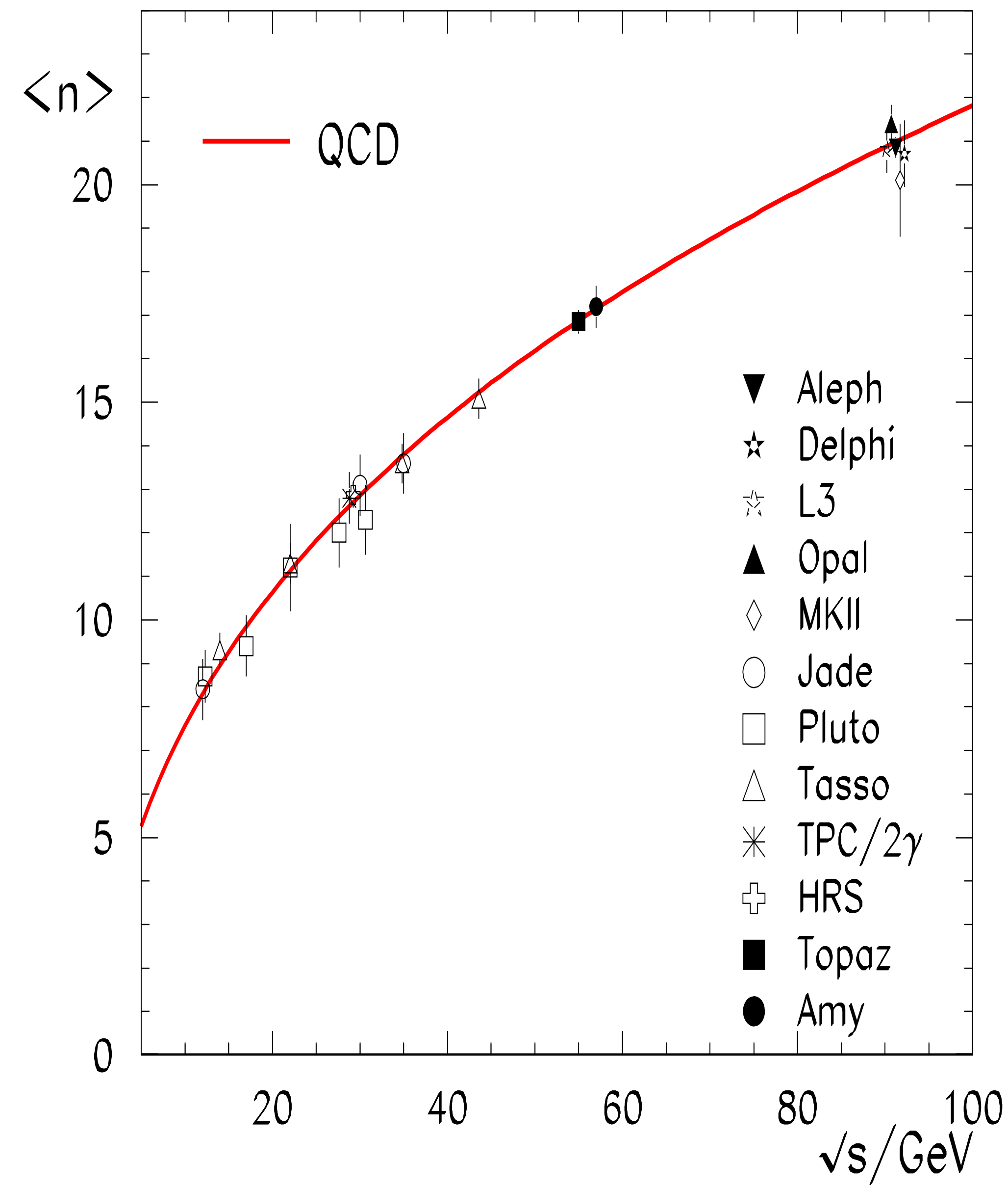
$$\langle N_g \rangle \sim \sum_n \frac{1}{(n!)^2} \left(\frac{C_A}{\pi b} \ln \frac{Q}{\Lambda} \right)^n$$

$$\sim \exp \sqrt{\frac{4C_A}{\pi b} \ln \frac{Q}{\Lambda}}$$

Compare to data for **hadron** multiplicity ($Q \equiv \sqrt{s}$)

Including some other higher-order terms and fitting overall normalisation

Agreement is amazing!



charged hadron multiplicity in e^+e^- events adapted from ESW

Resummation of p_T: steps along the way

$$v(p_1, \ldots, p_m) = \left| \sum_{i=1}^{m} \vec{p}_{T,i} \right| \simeq \max(p_{T1}, \ldots, p_{Tm})$$

$$\Theta(V - v(p_1, \ldots, p_m)) = \prod_{i=1}^{m} \Theta(V - p_{Ti})$$

$$\sum_{m=0}^{\infty} \frac{1}{m!} \prod_{i=1}^{m} \left(\frac{2\alpha_s C_F}{\pi} \int_{\epsilon} \frac{dE_i}{E_i} \int_{\epsilon} \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right) \times (V - v(p_1, \ldots, p_m))$$

$$= \exp\left[\frac{2\alpha_s C_F}{\pi} \int_{\epsilon} \frac{dE}{E} \int_{\epsilon} \frac{d\theta}{\theta} \Theta(V - E\theta) \right]$$

$$\sum_{n=0}^{\infty} \frac{1}{n!} \prod_{i=m+1}^{m+n} \left(- \frac{2\alpha_s C_F}{\pi} \int_{\epsilon} \frac{dE_i}{E_i} \int_{\epsilon} \frac{d\theta_i}{\theta_i} \int_0^{2\pi} \frac{d\phi_i}{2\pi} \right)$$

$$= \exp\left[- \frac{2\alpha_s C_F}{\pi} \int_{\epsilon} \frac{dE}{E} \int_{\epsilon} \frac{d\theta}{\theta} \right]$$