Defining jets at the dawn of the LHC

Grégory Soyez

CERN

In collaboration with Gavin Salam, Matteo Cacciari and Juan Rojo

EPFL, Lausanne — January 11 2010

Plan

- Jet algorithms and jet definitions
  - basic ideas: why jets? recombinations and cones
  - failures of the $20^{\rm th}$-century cone algorithms
  - new algorithms without the failures

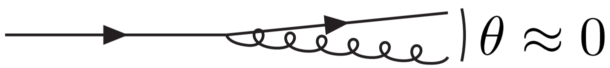
- More advanced topics: how to better use the tools we have?
  - jet areas: tool for pileup subtraction
  - new generation of algorithms
  - optimal choice (for kinematic reconstructions)

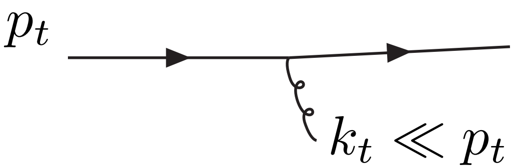
Unavoidable theory

QCD probability for gluon emission (angle θ and $\perp$-mom. k_t):

$$dP \propto \alpha_s \frac{d\theta}{\theta} \frac{dk_t}{k_t}$$

Two divergences:





collinear

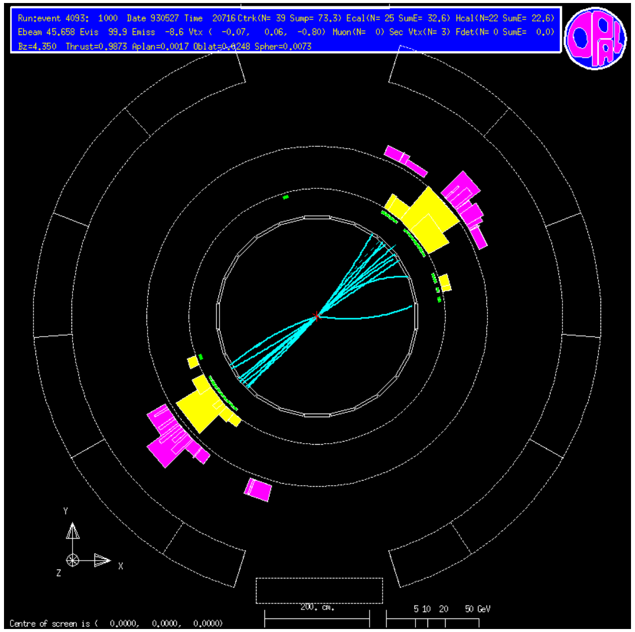
soft

Divergences cancelled by virtual corrections

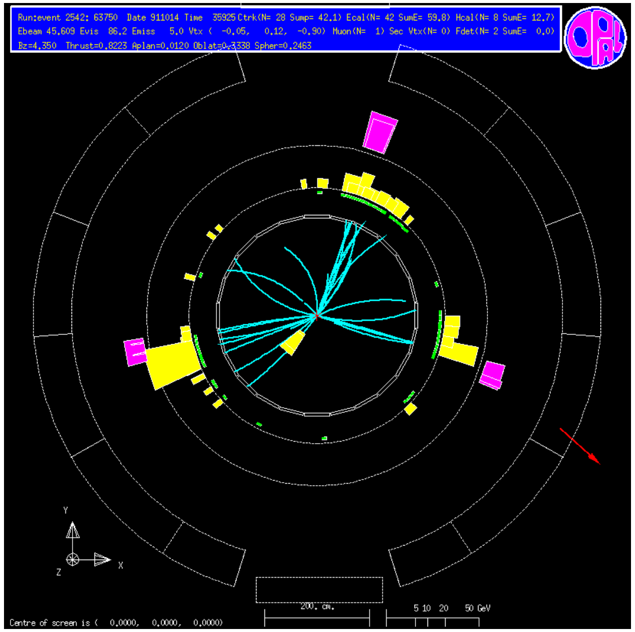
Motivation: why jets

Collinear divergence $\Rightarrow$ QCD produces “jetty” showers

Example: LEP (OPAL) events



2 jets



3 jets

“Jets” $\equiv$ bunch of collimated particles $\cong$ hard partons

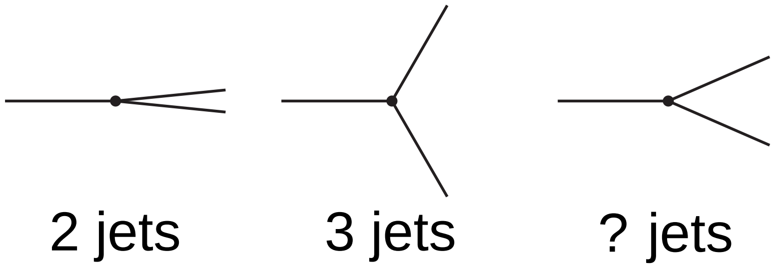
Motivation: why jets

Collinear divergence $\Rightarrow$ QCD produces “jetty” showers

“Jets” $\equiv$ bunch of collimated particles $\cong$ hard partons

BUT

- a “parton” is an ambiguous concept (NLO)
- “collinear” has some arbitraryness



Motivation: why jets

Collinear divergence $\Rightarrow$ QCD produces “jetty” showers

“Jets” $\equiv$ bunch of collimated particles $\cong$ hard partons

BUT

- a
- “c

In practice: use of a jet definition

$$\text{particles } \{p_i\} \xrightarrow[\text{definition}]{\text{jet}} \text{jets } \{j_k\}$$

Jet algorithm: the recipe (insufficient!)

Jet definition: algorithm + the parameters

20th century jet algorithms

Recombination:

- k_t algorithm
- Cambridge/Aachen alg.

Cone:

- CDF JetClu
- CDF MidPoint
- D0 (run II) Cone
- PxCone
- ATLAS Cone
- CMS Iterative Cone
- PyCell/CellJet
- GetJet

20th century jet algorithms

Recombination:

- k_t algorithm
- Cambridge/Aachen alg.

Idea: undo the showering

Successively

- find the closest pair of particles
- recombine them

Distance:

k_t:

$$d_{i,j} = \text{min}(k_{t,i}^2, k_{t,j}^2)(\Delta\phi_{i,j}^2 + \Delta y_{i,j}^2)$$

Cam/Aachen:

$$d_{i,j} = \Delta\phi_{i,j}^2 + \Delta y_{i,j}^2$$

stop at a distance R

20th century jet algorithms

Idea: dominant flow of energy

Stable cone (radius R):
sum of particles in the cone points towards the cone centre

All these are iterative cones:

- start from a seed
- iterate until stable

seeds = {particles, midpoints}

Jet $\equiv$ stable cone
modulo overlapping

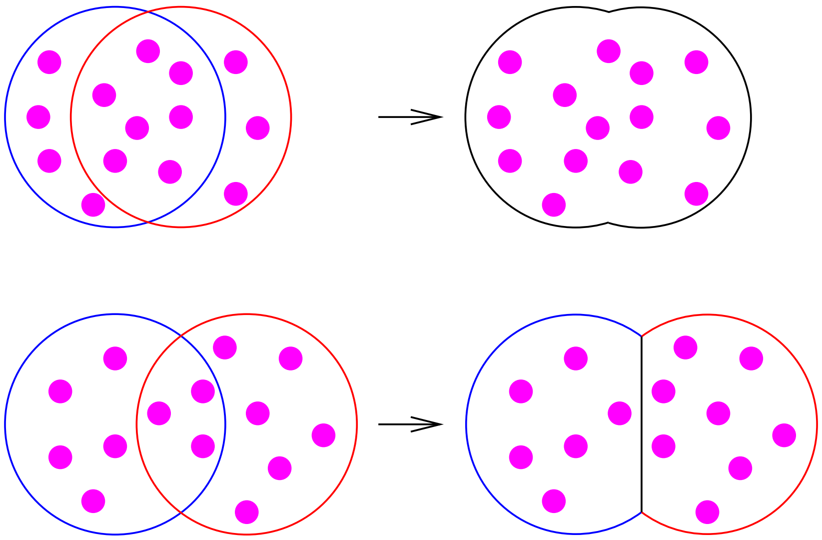
Cone:

- CDF JetClu
- CDF MidPoint
- D0 (run II) Cone
- PxCone
- ATLAS Cone
- CMS Iterative Cone
- PyCell/CellJet
- GetJet

20th century jet algorithms

Cone with split-merge

Split/merge if the overlap is smaller/larger than a threshold f



Cone:

- CDF JetClu
- CDF MidPoint
- D0 (run II) Cone
- PxCone
- ATLAS Cone
- CMS Iterative Cone
- PyCell/CellJet
- GetJet

20th century jet algorithms

Cone with progressive removal

Successively

- iterate from hardest particle
- call that a jet (remove particles)

Basic property:

hard circular jets

Cone:

- CDF JetClu
- CDF MidPoint
- D0 (run II) Cone
- PxCone
- ATLAS Cone
- CMS Iterative Cone
- PyCell/CellJet
- GetJet

20th century jet algorithms

Recombination:

- k_t algorithm
- Cambridge/Aachen alg.

✓perturbative behaviour

Cone:

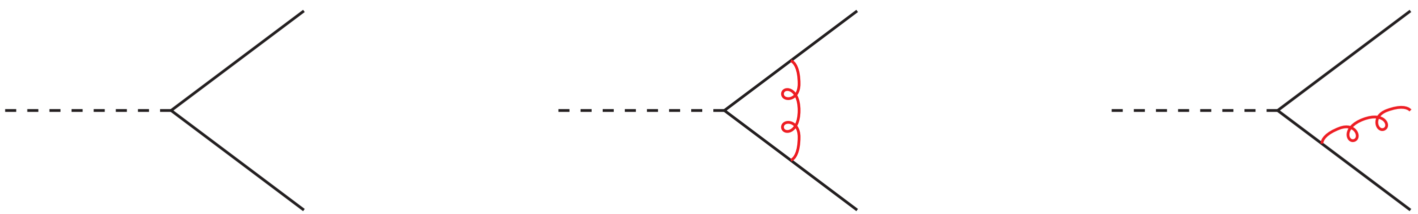
- CDF JetClu
- CDF MidPoint
- D0 (run II) Cone
- PxCone
- ATLAS Cone
- CMS Iterative Cone
- PyCell/CellJet
- GetJet

✓UE sensitivity

21st century: how does that picture change?

QCD divergences

Ingredient: QCD soft and collinear divergencies



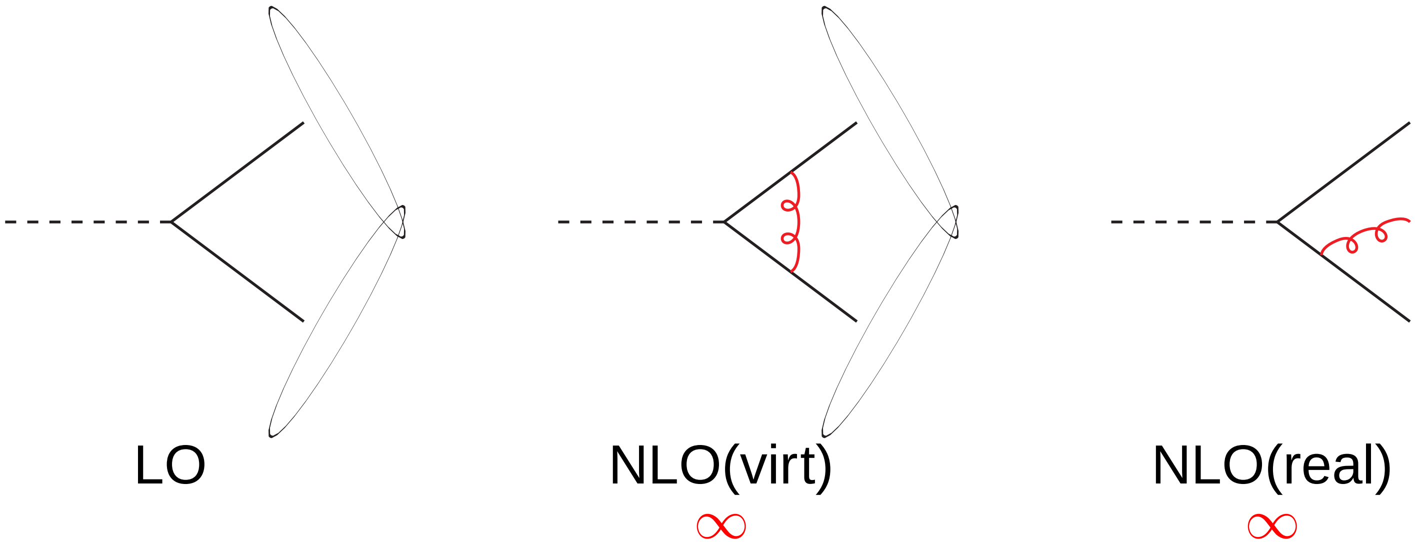
LO NLO(virt) NLO(real)

∞ ∞

- ∞ (from soft gluons) cancel (inclusive x-section)

QCD divergences

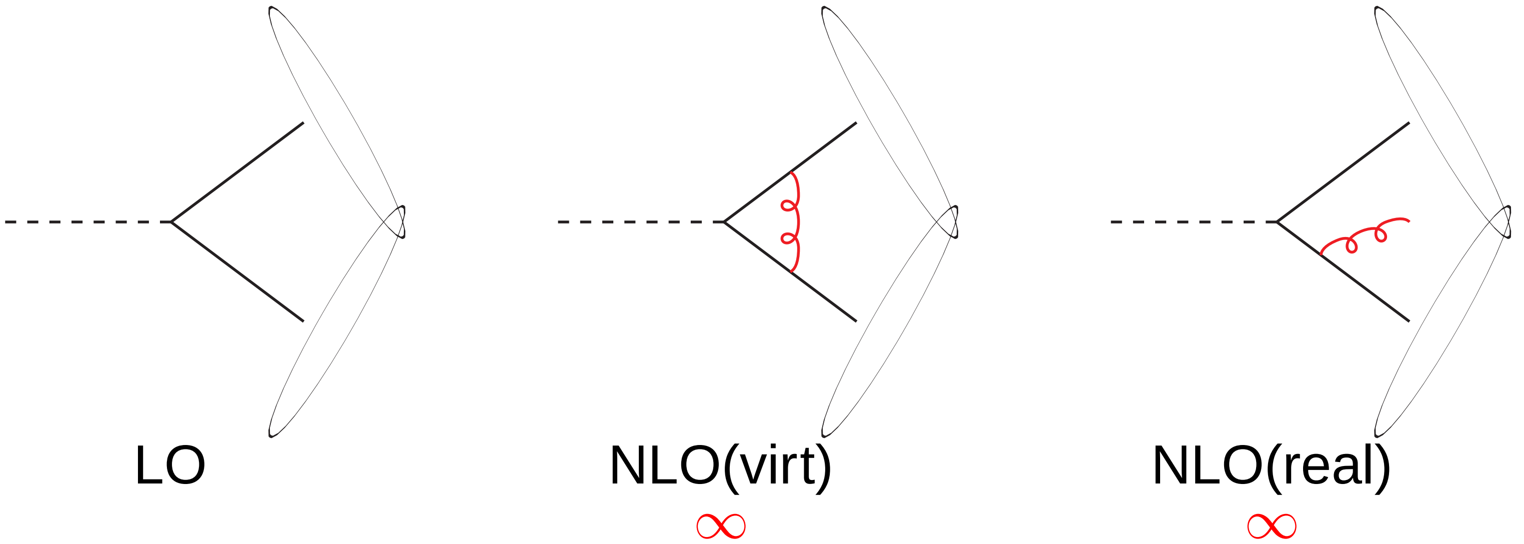
Ingredient: QCD soft and collinear divergencies



- Consider an extra (NLO) soft gluon
- Assume LO gives 2 jets $\Rightarrow$ NLO(virt) gives 2 jets

QCD divergences

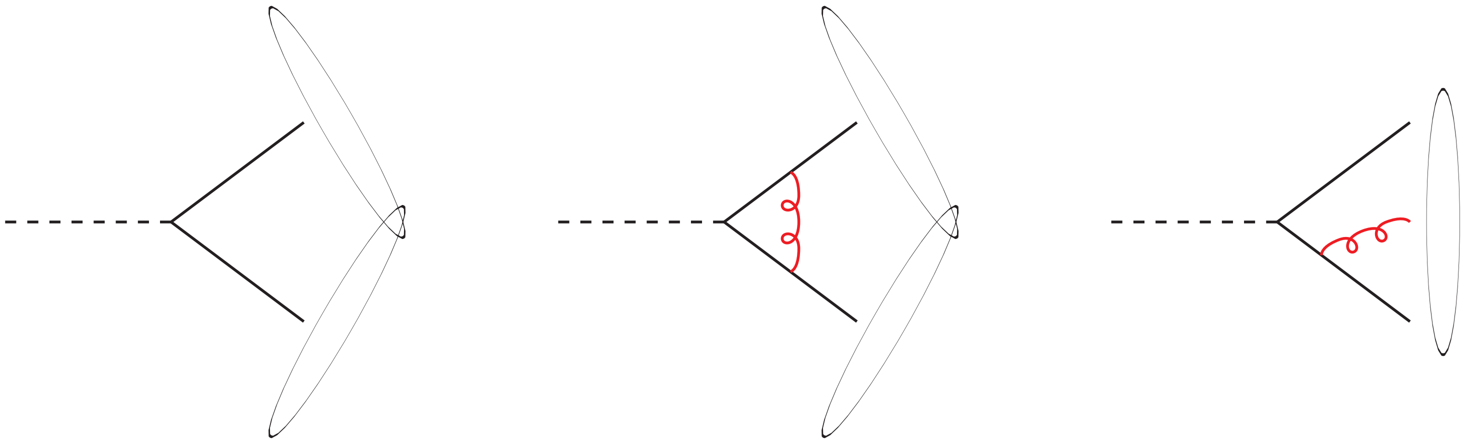
Ingredient: QCD soft and collinear divergencies



- Consider an extra (NLO) soft gluon
- Assume LO gives 2 jets $\Rightarrow$ NLO(virt) gives 2 jets
- NLO(real) gives 2 jets $\Rightarrow$ ∞ cancel $\Rightarrow$ finite jet cross-section

QCD divergences

Ingredient: QCD soft and collinear divergencies

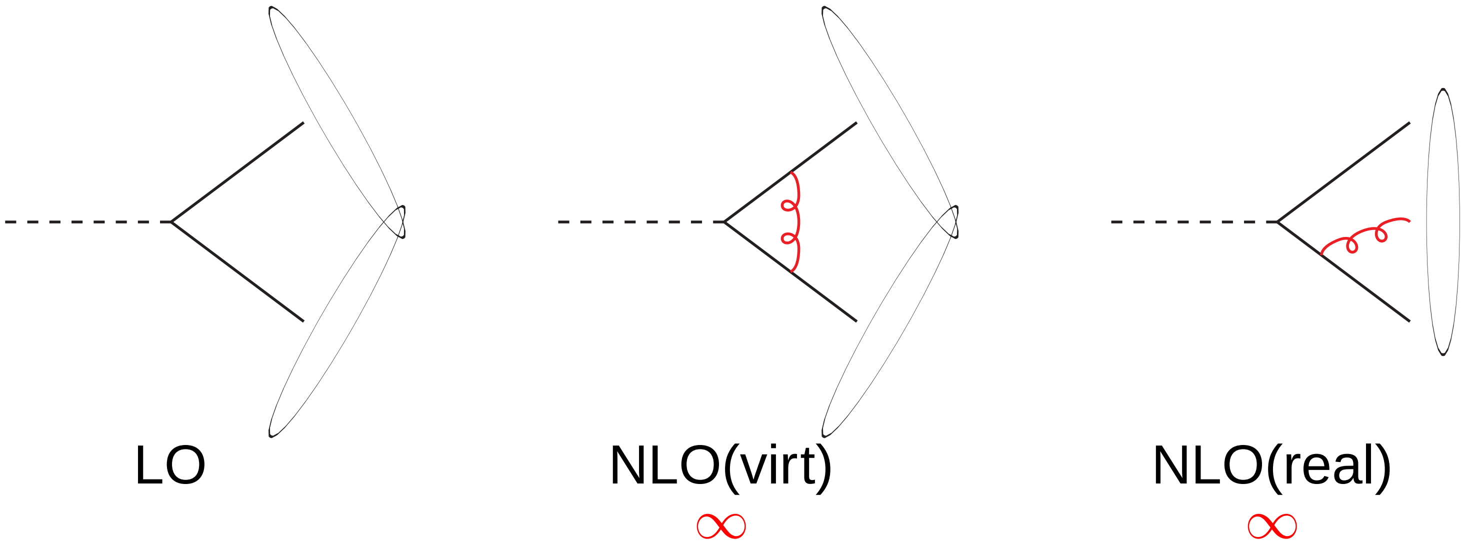


LO NLO(virt) ∞ NLO(real) ∞

- Consider an extra (NLO) soft gluon
- Assume LO gives 2 jets $\Rightarrow$ NLO(virt) gives 2 jets
- NLO(real) gives 2 jets $\Rightarrow \infty$ cancel $\Rightarrow$ finite jet cross-section
  NLO(real) gives 1 jets $\Rightarrow \infty$ do not cancel $\Rightarrow$ infinite jet x-section

QCD divergences

Ingredient: QCD soft and collinear divergencies

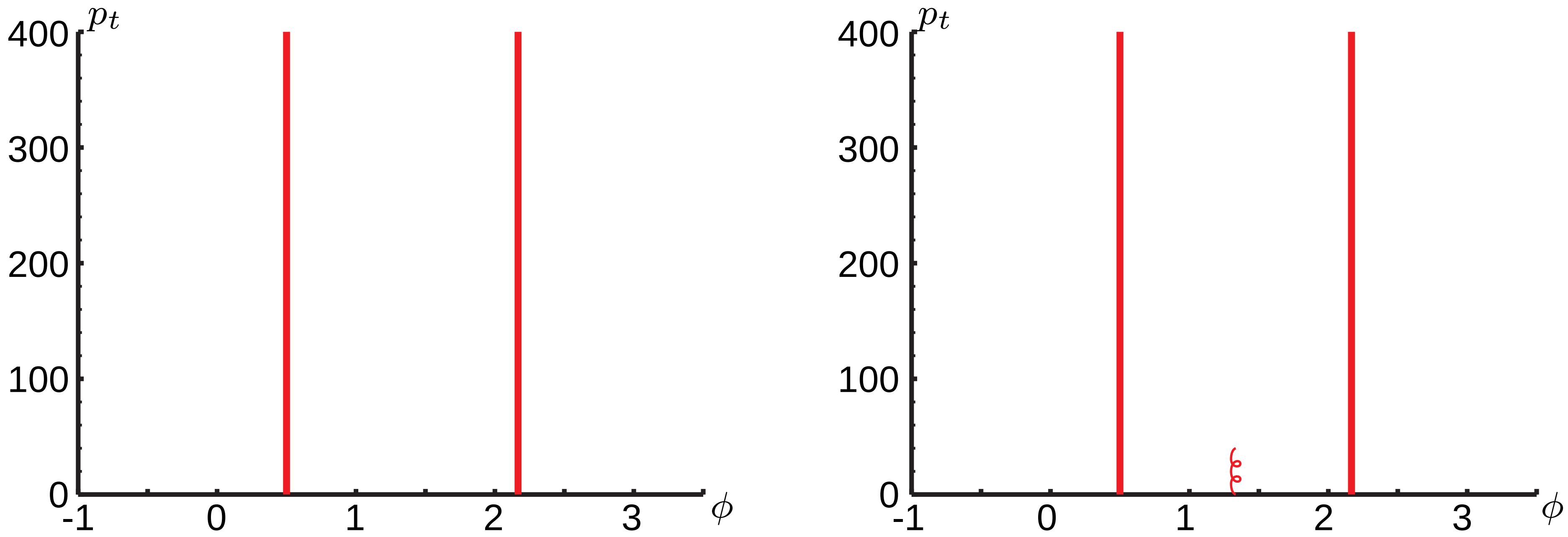


For pQCD to make sense, the (hard) jets should not change when

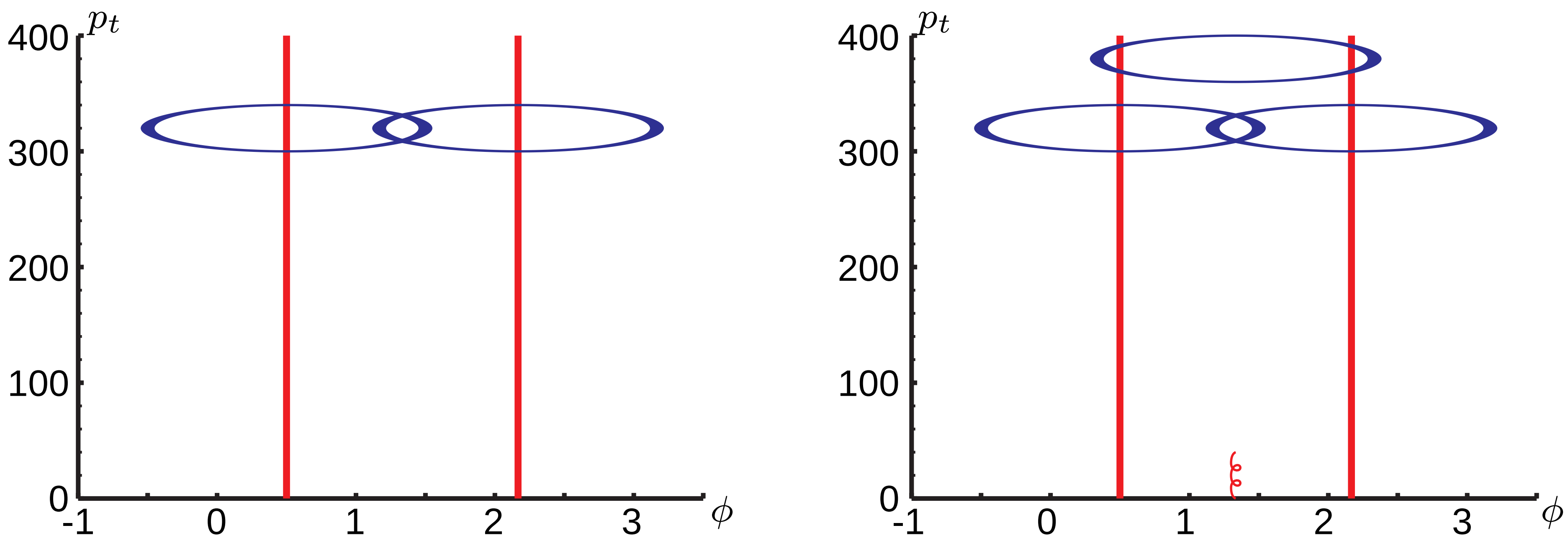
- one has a soft emission *i.e.* adds a very soft gluon
- one has a collinear splitting
  i.e. replaces one parton by two at the same place (η, ϕ)

[SNOWMASS Accords, Fermilab, 1990]

IR (un)safety? JetClu and Atlas Cone

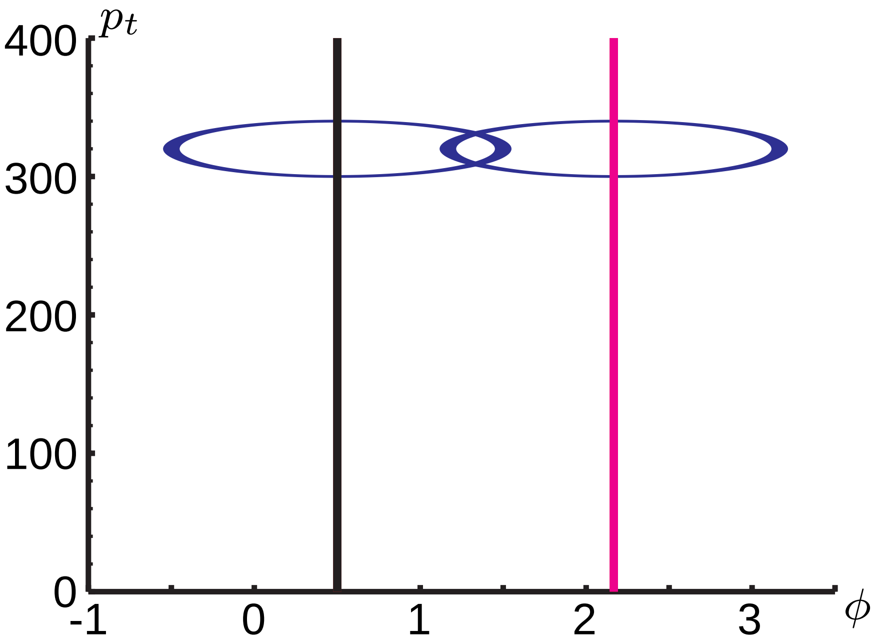


IR (un)safety? JetClu and Atlas Cone

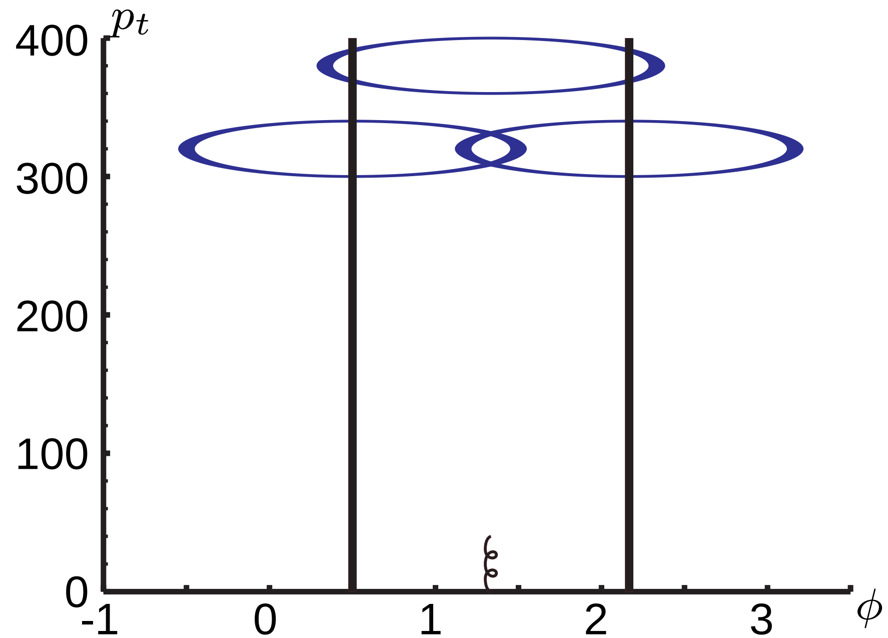


Stable cones found

IR (un)safety? JetClu and Atlas Cone



2 jets

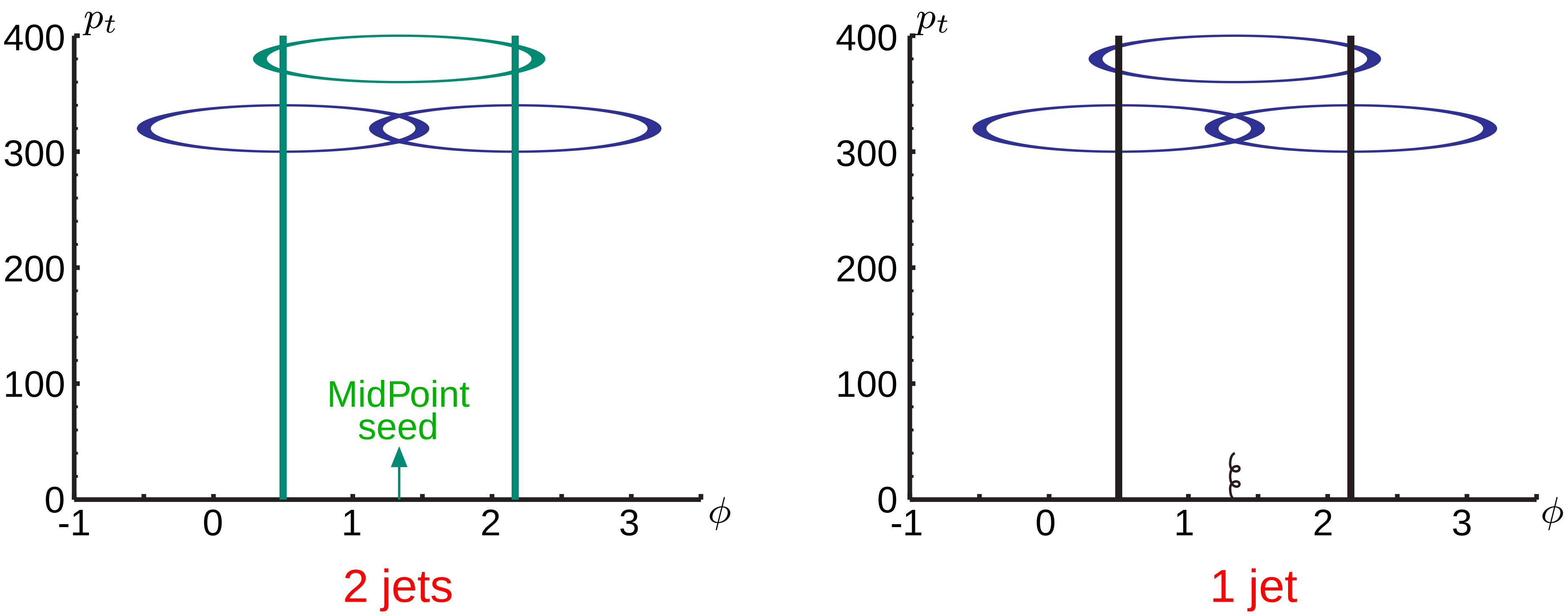


1 jet

A soft gluon changed the number of jets

$\Rightarrow$ IR unsafety of JetClu and the ATLAS Cone

IR (un)safety? JetClu and Atlas Cone



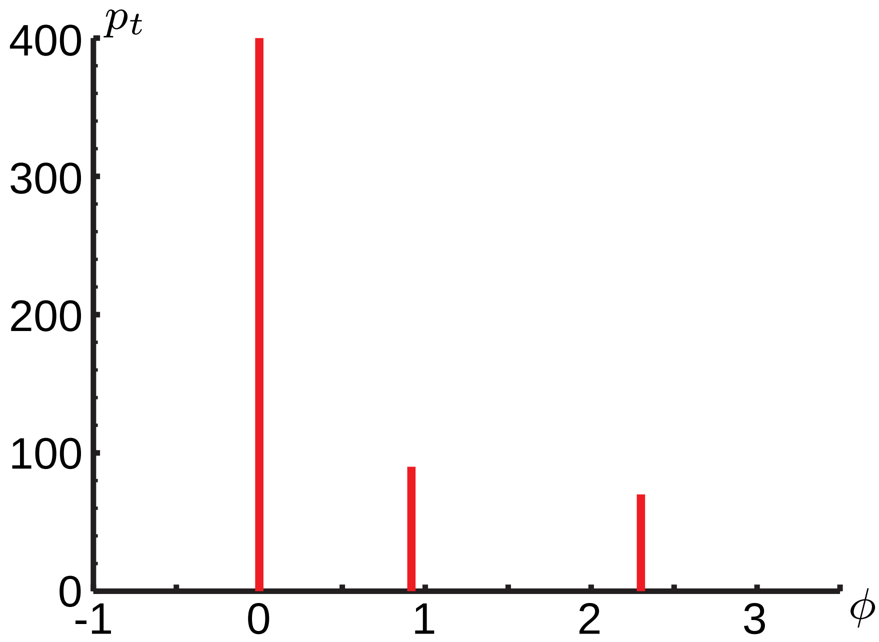
A soft gluon changed the number of jets

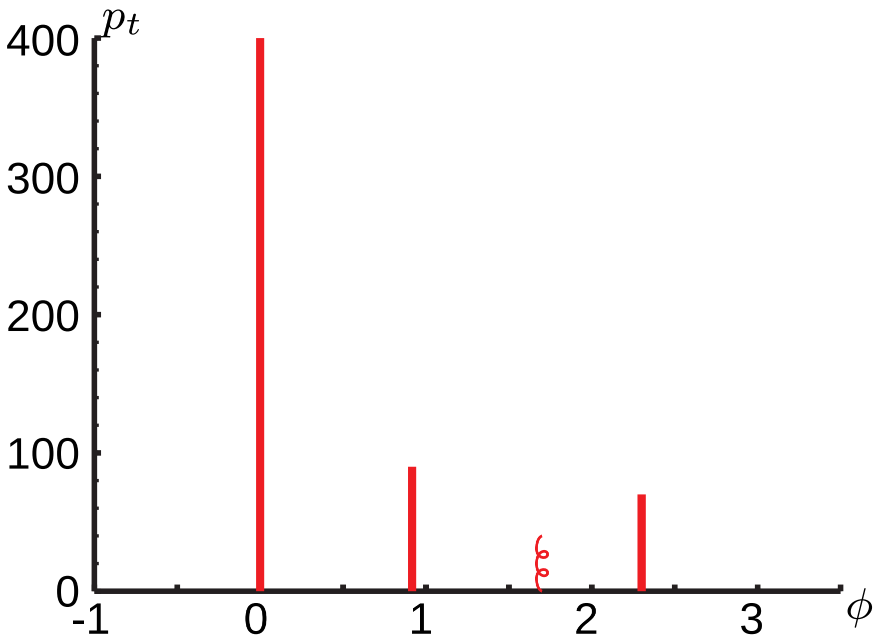
⇒ IR unsafety of JetClu and the ATLAS Cone

Fixed by MidPoint

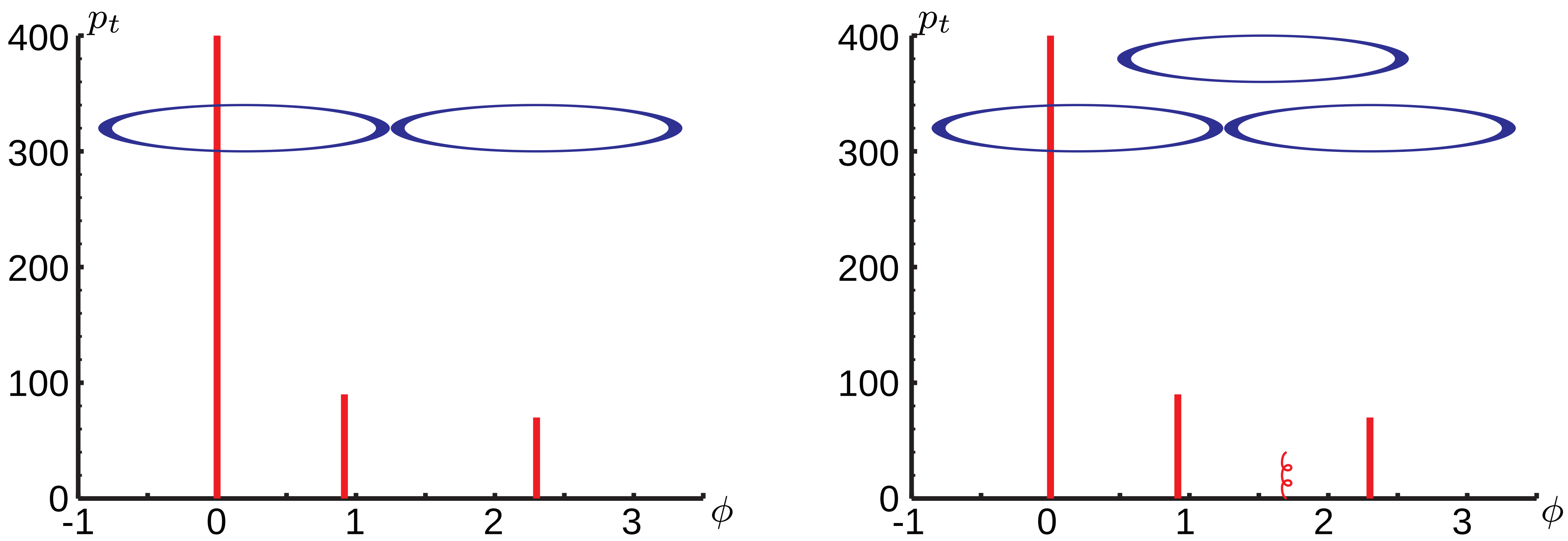
[Blazey *et al.*, 00]

IR (un)safety? MidPoint



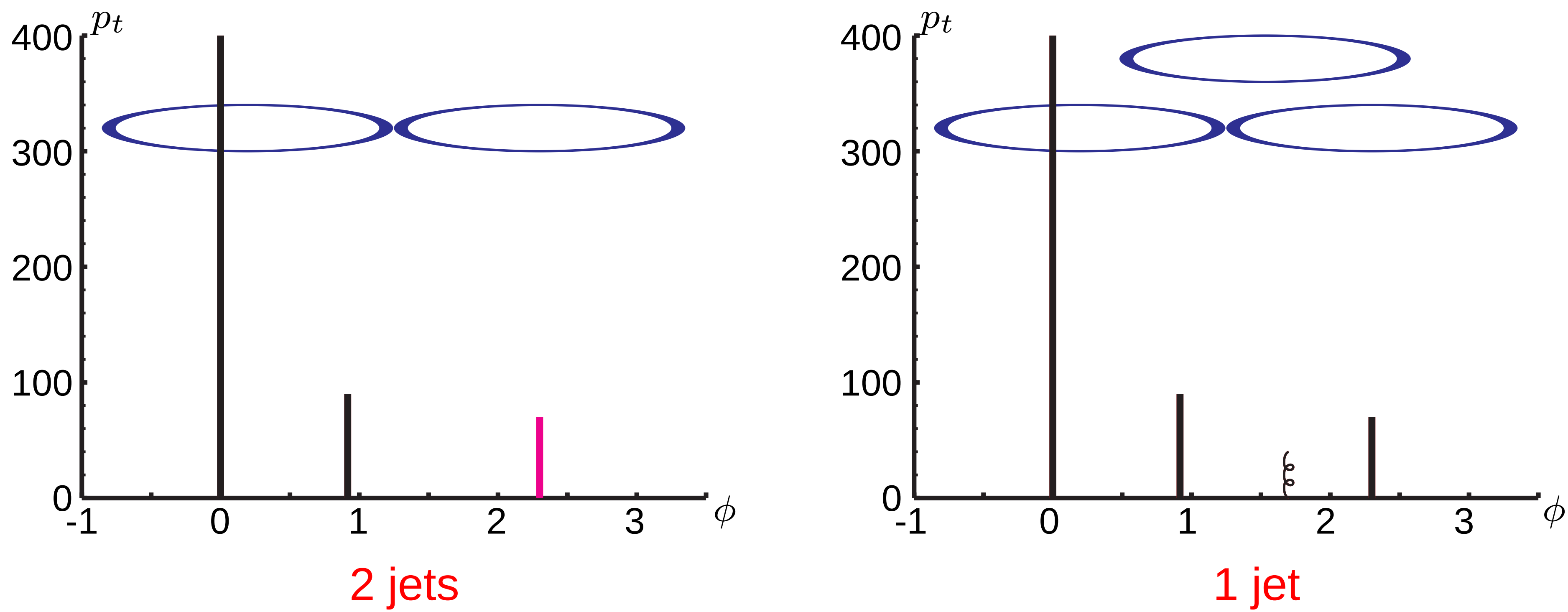


IR (un)safety? MidPoint



Stable cones found

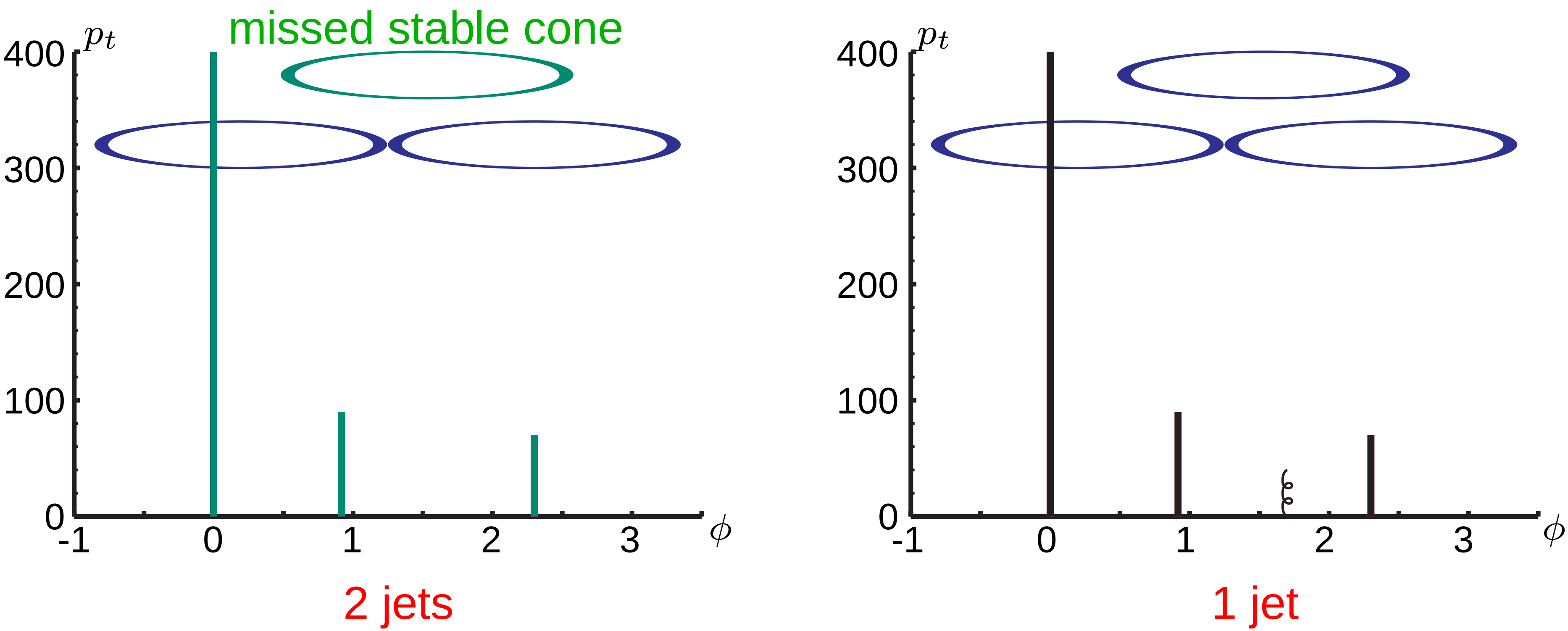
IR (un)safety? MidPoint



A soft gluon changed the number of jets

$\Rightarrow$ IR unsafety of MidPoint (1 order in α_s later than JetClu)

IR (un)safety? MidPoint



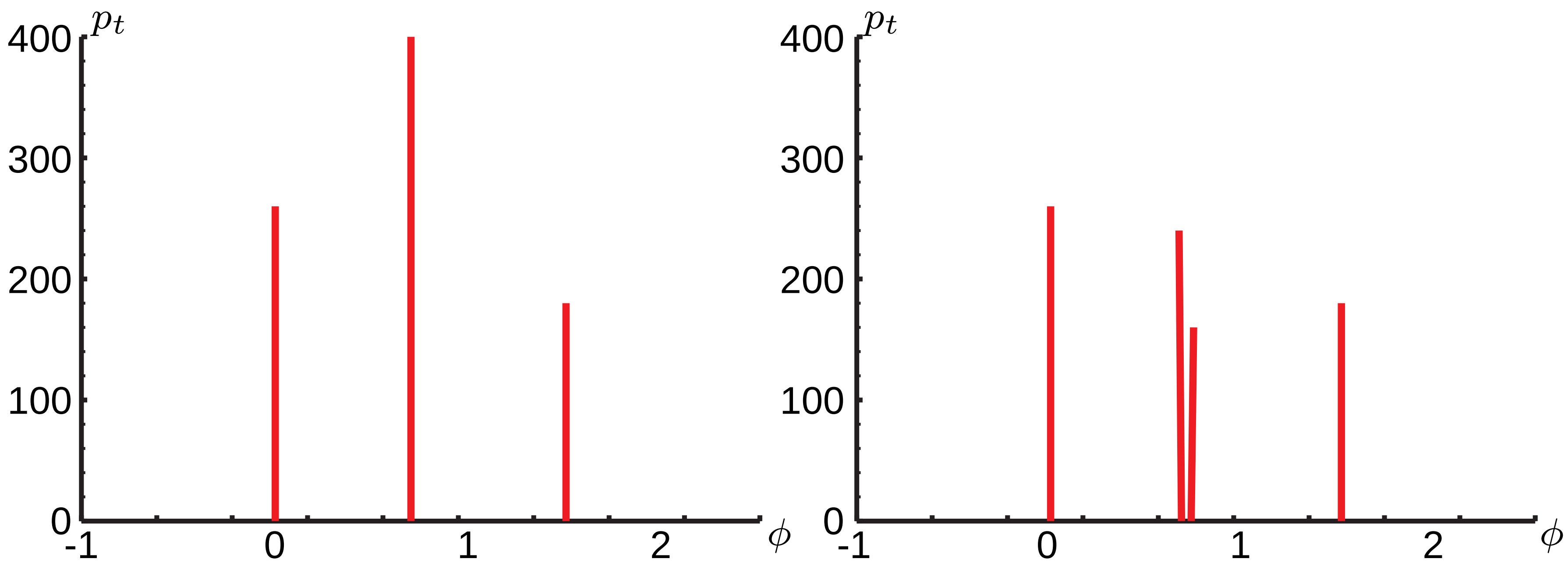
Solution: be sure to find **all** stable cones

> SISCone: Seedless Infrared-Safe Cone algorithm
> `http://projects.hepforge.org/siscone`

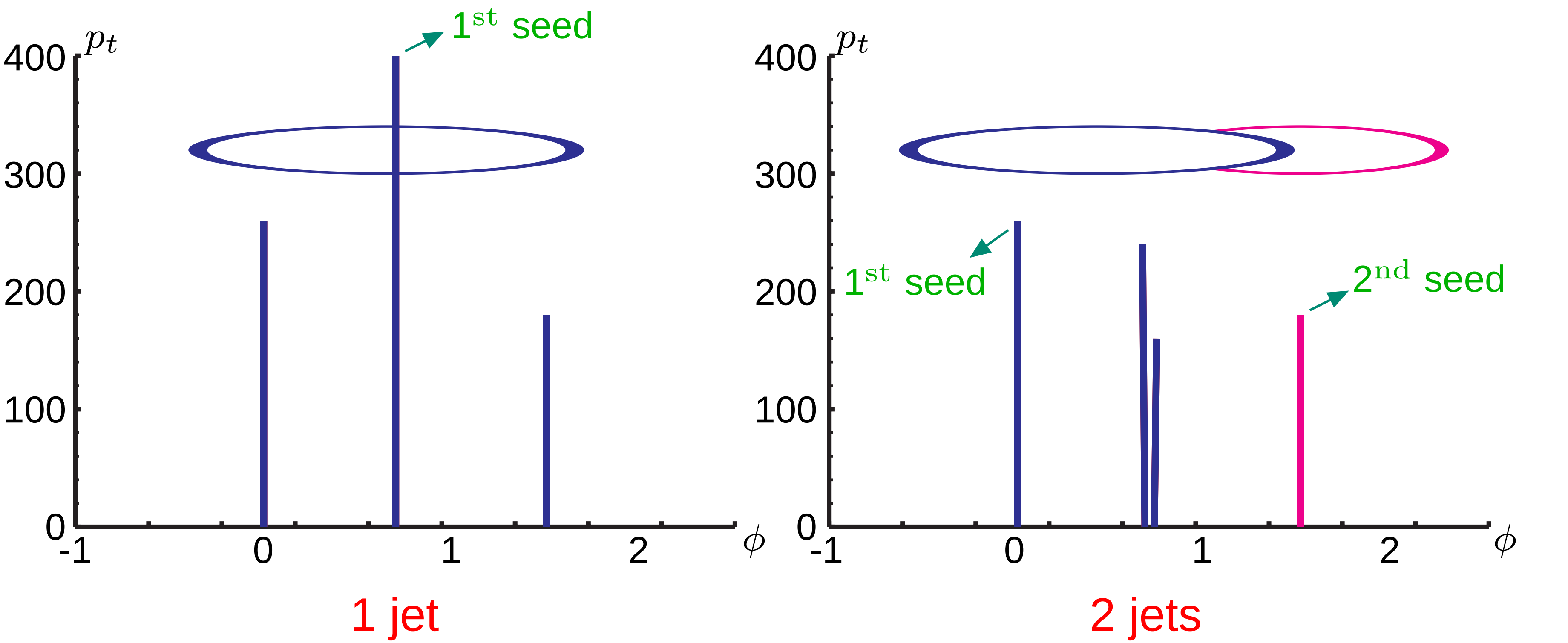
[G.Salam, G.S., 07]

Idea: enumerate enclosures by enumerating pairs of particles

Collinear (un)safety? the CMS iterative cone



Collinear (un)safety? the CMS iterative cone



A colinear splitting changed the number of jets

$\Rightarrow$ Collinear unsafety of the CMS iterative cone

Anti-k_t

Come back to recombination-type algorithms:

$$d_{ij} = \min(k_{t,i}^{2p}, k_{t,j}^{2p}) \left(\Delta\phi_{ij}^2 + \Delta\eta_{ij}^2\right)$$

- $p = 1$: k_t algorithm
- $p = 0$: Aachen/Cambridge algorithm

Anti-k_t

Come back to recombination-type algorithms:

$$d_{ij} = \min(k_{t,i}^{2p}, k_{t,j}^{2p}) \ (\Delta\phi_{ij}^2 + \Delta\eta_{ij}^2)$$

- $p = 1$: k_t algorithm
- $p = 0$: Aachen/Cambridge algorithm
- $p = -1$: anti-k_t algorithm [M.Cacciari, G.Salam, G.S., 08]

Why should that be related to the iterative cone ?!?

- "large $k_t \Rightarrow$ small distance"
  i.e. hard partons "eat" everything up to a distance R
  i.e. circular/regular jets, jet borders unmodified by soft radiation
- infrared and collinear safe

21st century jet finders

Recombination:

- k_t algorithm
- Cambridge/Aachen alg.
- anti-k_t algorithm

Cone:

- CDF JetClu
- CDF MidPoint
- D0 (run II) Cone
- PxCone
- ATLAS Cone
- CMS Iterative Cone
- PyCell/CellJet
- GetJet
- SISCone

21st century jet finders

Recombination:

- k_t algorithm
- Cambridge/Aachen alg.
- anti-k_t algorithm

Cone:

- CDF JetClu
- CDF MidPoint
- D0 (run II) Cone
- PxCone
- ATLAS Cone
- CMS Iterative Cone
- PyCell/CellJet
- GetJet
- SISCone

4 available
safe algorithms

21st century jet finders

Recombination:

- k_t algorithm
- Cambridge/Aachen alg.
- anti-k_t algorithm

Cone:

- CDF JetClu
- CDF MidPoint
- D0 (run II) Cone
- SISCone

```
#--------------------------------------------------------------------------
#                          FastJet release 2.4
#          Written by M. Cacciari, G.P. Salam and G. Soyez
#                         http://www.fastjet.fr
#--------------------------------------------------------------------------
```

All those algorithms (and much more) implemented (efficiently) in FastJet

21st century jet finders

Recombination:

- k_t algorithm
- Cambridge/Aachen alg.
- anti-k_t algorithm

Cone:

- CDF JetClu
- CDF MidPoint
- D0 (run II) Cone
- PxCone
- ...e
- GetJet
- SISCone

safe algorithms

anti-k_t adopted as the default jet algorithm by both CMS and ATLAS

When does IRC safety matters?

Take *e.g.* the MidPoint cone

$$\overbrace{\alpha_s^2 \times \ldots}^{\text{2 particles}} + \overbrace{\alpha_s^3 \times \ldots}^{\text{3 particles}} + \overbrace{\alpha_s^4 \times \ldots}^{\text{4 particles}} + \overbrace{\alpha_s^5 \times \qquad\qquad\qquad\qquad \ldots}^{\text{4 particles + 1 soft}} + \ldots$$

- QCD expansion (one α_s can be replaced by α_{EW})

When does IRC safety matters?

Take *e.g.* the MidPoint cone

$$\overbrace{\alpha_s^2 \times \ldots}^{\text{2 particles}} + \overbrace{\alpha_s^3 \times \ldots}^{\text{3 particles}} + \overbrace{\alpha_s^4 \times \ldots}^{\text{4 particles}} + \overbrace{\alpha_s^5 \times \log(p_t/\Lambda_{\rm QCD}) \ldots}^{\text{4 particles + 1 soft}} + \ldots$$

- QCD expansion (one α_s can be replaced by $\alpha_{\rm EW}$)
- IRC unsafety (regulated at the hadronic scale $\sim \Lambda_{\rm QCD}$)

When does IRC safety matters?

Take *e.g.* the MidPoint cone

$$\underbrace{\overbrace{\alpha_s^2 \times \ldots}^{\text{2 particles}} + \overbrace{\alpha_s^3 \times \ldots}^{\text{3 particles}} + \overbrace{\alpha_s^4 \times \ldots}^{\text{4 particles}} + \overbrace{\alpha_s^5 \times \log(p_t/\Lambda_{\rm QCD}) \ldots}^{\text{4 particles + 1 soft}} + \ldots}_{\text{cannot be trusted (from } \alpha_s^4 \text{ on)}}$$

- QCD expansion (one α_s can be replaced by $\alpha_{\rm EW}$)
- IRC unsafety (regulated at the hadronic scale $\sim \Lambda_{\rm QCD}$)
- $\alpha_s \log(p_t/\Lambda_{\rm QCD}) \sim 1$
- **last meaningful order =** $\boldsymbol{\alpha_s^3}$ **or** $\boldsymbol{\alpha_{\rm EW}\alpha_s^2}$

When does IRC safety matters?

Take *e.g.* the MidPoint cone

$$\overbrace{\alpha_s^2 \times \ldots}^{\text{2 particles}} + \overbrace{\alpha_s^3 \times \ldots}^{\text{3 particles}} + \underbrace{\overbrace{\alpha_s^4 \times \ldots}^{\text{4 particles}} + \overbrace{\alpha_s^5 \times \log(p_t/\Lambda_{\text{QCD}}) \ldots}^{\text{4 particles + 1 soft}} + \ldots}_{\text{cannot be trusted}}$$

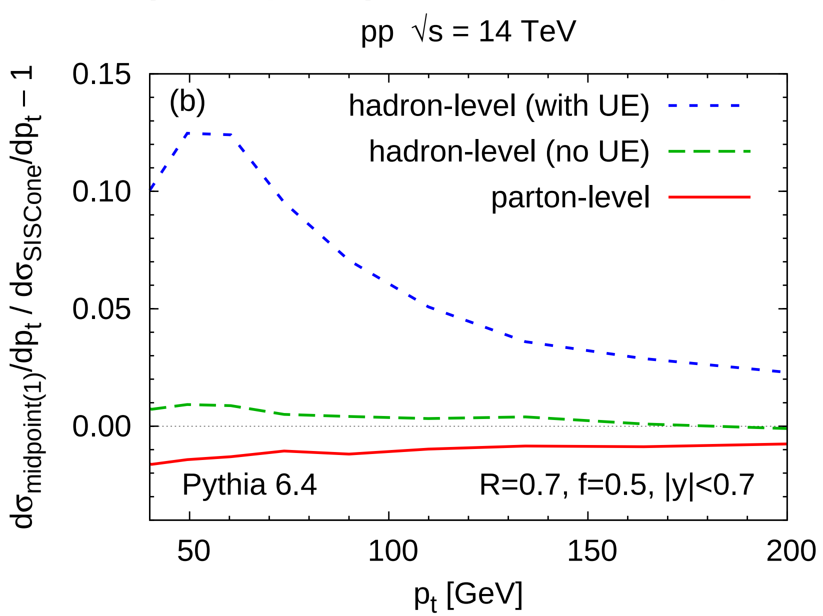
- QCD expansion (one α_s can be replaced by α_{EW})
- IRC unsafety (regulated at the hadronic scale $\sim \Lambda_{\text{QCD}}$)
- $\alpha_s \log(p_t/\Lambda_{\text{QCD}}) \sim 1$
- **last meaningful order = α_s^3 or $\alpha_{\text{EW}}\alpha_s^2$**
- same argument for the Iterative Cone
- 1 order worse for JetClu or the ATLAS cone

Physical impact

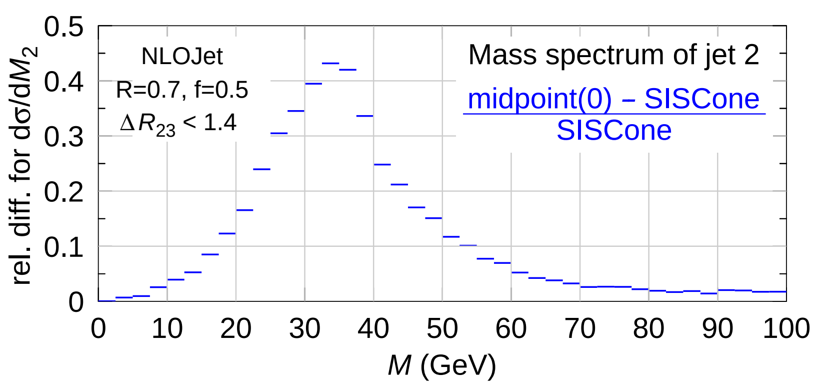
MidPoint/CMS iterative cone unsafe at $\mathcal{O}(\alpha_s^4)$ (or $\mathcal{O}(\alpha_{\rm ew}\alpha_s^3)$)

Physical observable	IRC-safe until		
	JetClu/ATLAS c.	MidPoint/CMS it. c.	SISCone/recomb.
Inclusive jet cross section	LO	NLO	any
3-jet cross section	none	LO	any
$W/Z/H$ + 2 jet cross sect.	none	LO	any
jet masses in 3 jets	none	none	any

Example: (Midpoint-SISCone)/SISCone



- Incl. cross-section: a few %
- Masses in 3-jet events: $\sim 45\%$



Physical impact

MidPoint/CMS iterative cone unsafe at $\mathcal{O}(\alpha_s^4)$ (or $\mathcal{O}(\alpha_{\rm ew}\alpha_s^3)$)

Physical observable	IRC-safe until		
	JetClu/ATLAS c.	MidPoint/CMS it. c.	SISCone/recomb.
Inclusive jet cross section	LO	NLO	any
3-jet cross section	**none**	LO	any
$W/Z/H$ + 2 jet cross sect.	**none**	LO	any
jet masses in 3 jets	**none**	**none**	any

Huge effort (~ 50 M€) to compute processes in pQCD

Note:

- arXiv:0903.0814: $W + 2$ jets vs. LO QCD using CDF JetClu
- arXiv:0903.1748: $Z + 2$ jets vs. NLO QCD using the D0runII cone
- arXiv:0903.1801: $Z + 2$ jets vs. NLO QCD using the CMS iterative cone

Summary of IRC safe algorithms

	Pros	Cons
k_t	matches QCD branchings	large UE sensitivity
Cam/Aa	good QCD behaviour easily look at $\neq$ scales	poor UE sensitivity
anti-k_t	easy algorithm easy calibration	poor UE sensitivity
SISCone	small UE sensitivity	poor QCD behaviour

We (finally) have a good set of tools

Can we do better?

A growing list

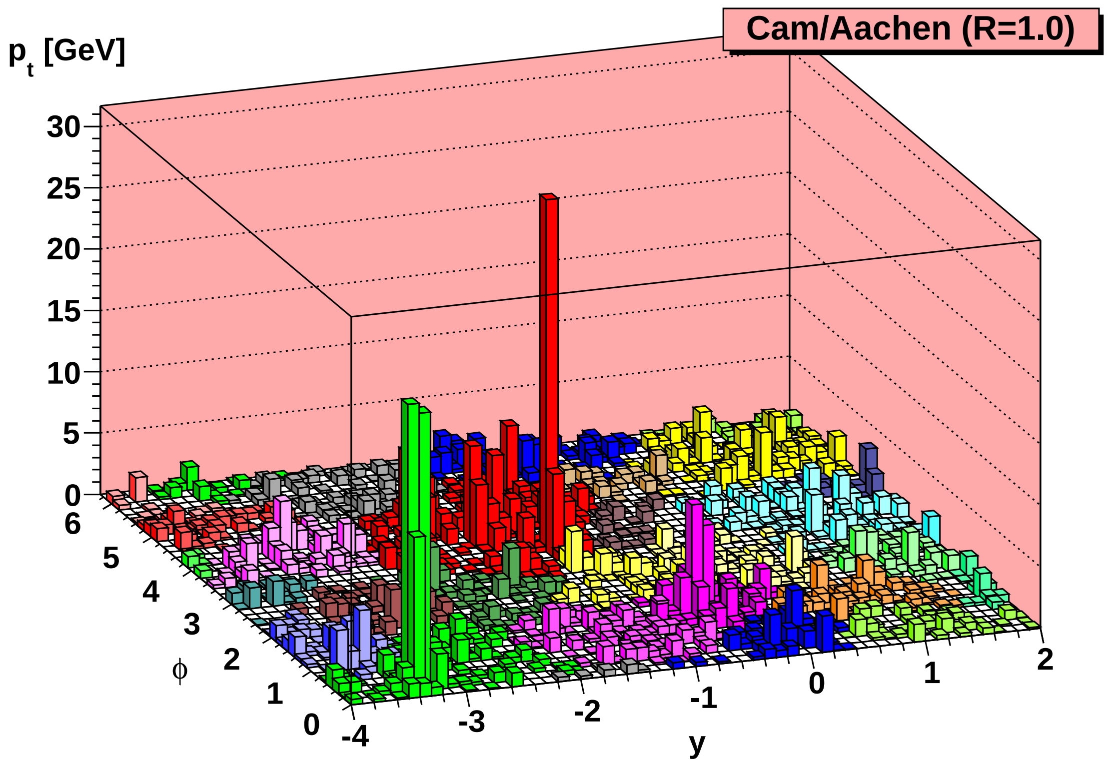
Many ideas and applications:

- ✓ jet areas and background subtraction
  $\longrightarrow$ UE, pileup, heavy-ion background subtraction
- ✓ jet substructure and filtering
  $\longrightarrow$ see below
- ✓ “best” jet definition
  $\longrightarrow$ kinematic dijet reconstruction
- ✓ boosted objects tagging
  $\longrightarrow H \to b\bar{b}$, t, $\tilde{\chi}_0^1 \to qqq$, $\ldots$

I will cover the first three (see *e.g.* Gavin Salam’s talk here for the 4th)

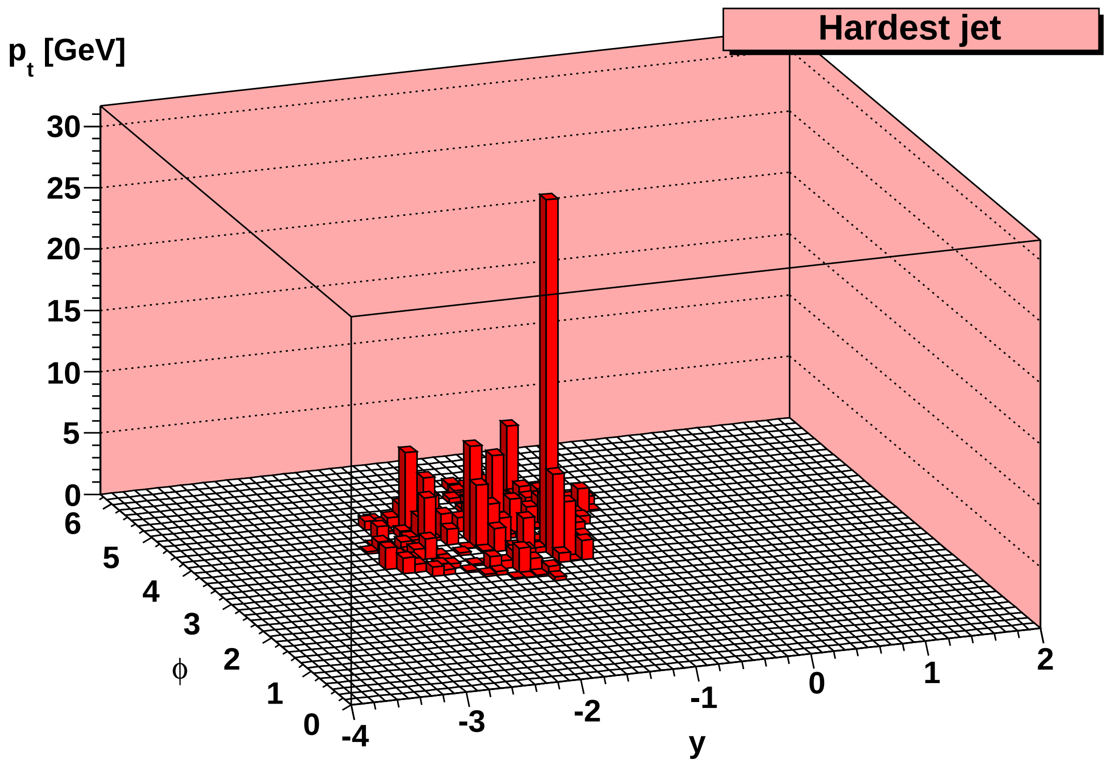
New idea #1: filtering

Filtering



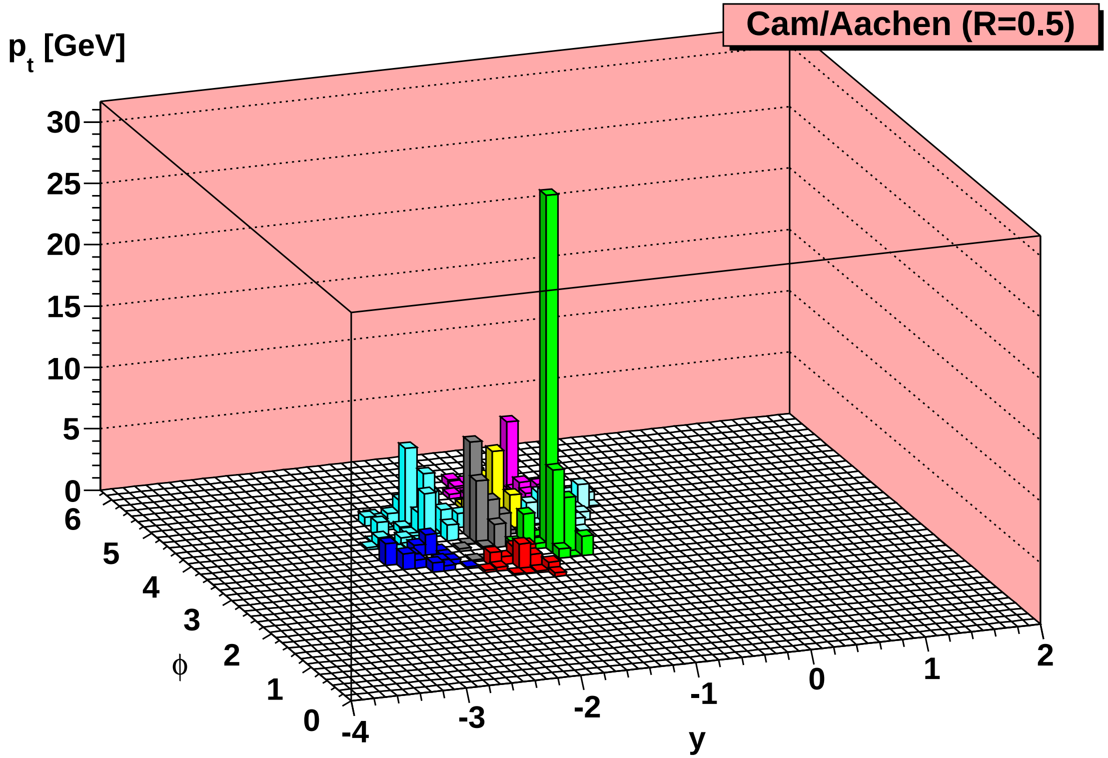
- cluster with Cambridge/Aachen(R)

Filtering



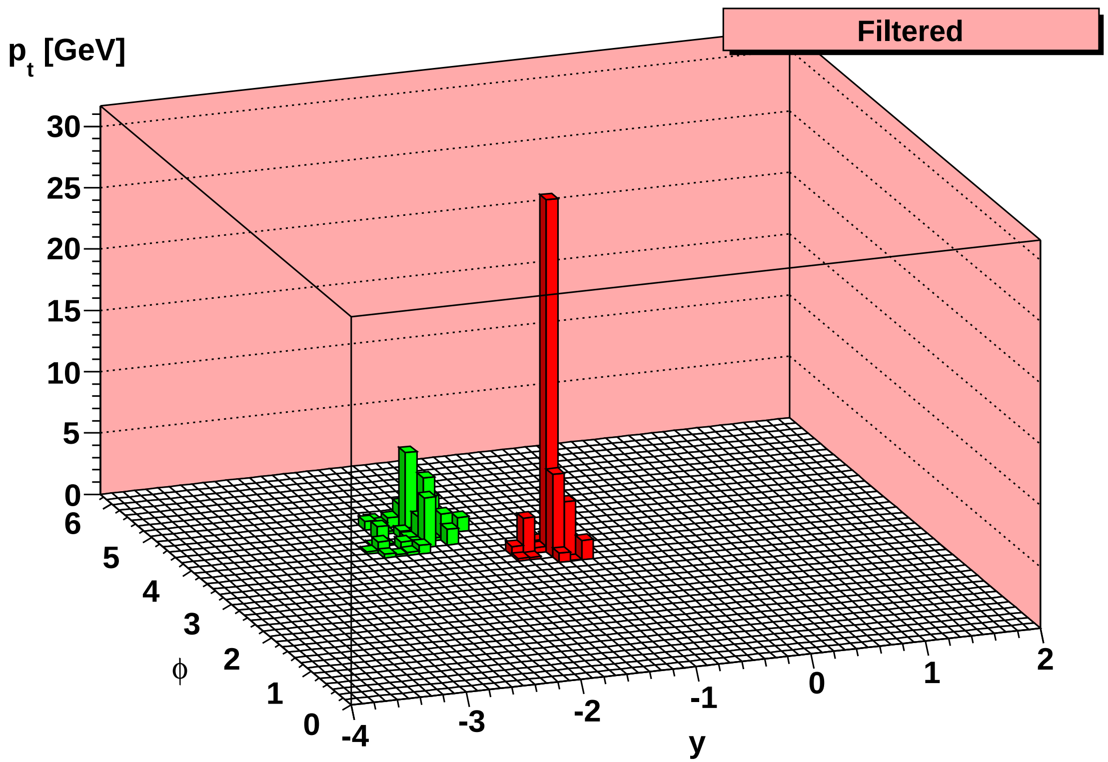
- cluster with Cambridge/Aachen(R)
- for each jet

Filtering



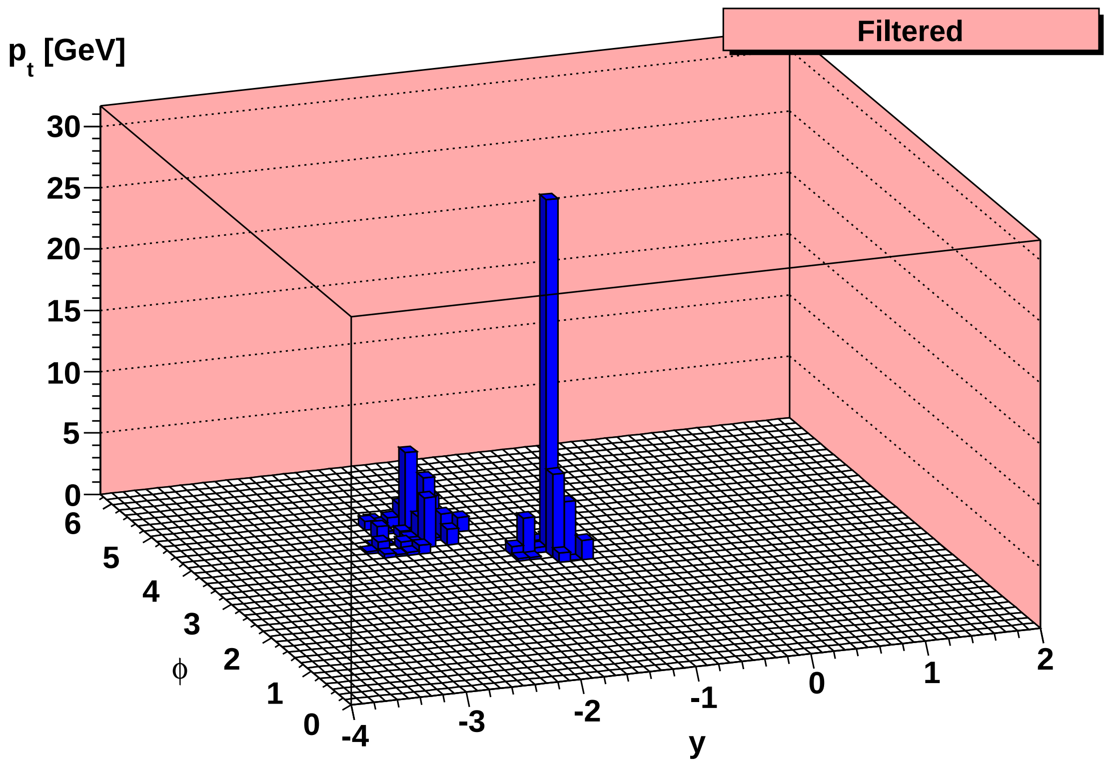
- cluster with Cambridge/Aachen(R)
- for each jet
  - recluster with Cambridge/Aachen(R/2)

Filtering



- cluster with Cambridge/Aachen(R)
- for each jet
  - recluster with Cambridge/Aachen(R/2)
  - keep the 2 hardest subjets

Filtering



- cluster with Cambridge/Aachen(R)
- for each jet
  - recluster with Cambridge/Aachen(R/2)
  - keep the 2 hardest subjets

Idea:

✓ keep perturb. radiation

✓ remove UE

- Proven useful for boosted jet $H \to b\bar{b}$ tagging

[J.Butterworth, A.Davison, M.Rubin, G.Salam, 08]

- Proven useful for kinematic reconstructions

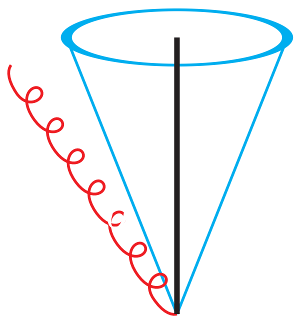
[M.Cacciari, J.Rojo, G.Salam, GS, 08]

New idea #2: jet definition optimisation

Optimisation: underlying idea

Competition between

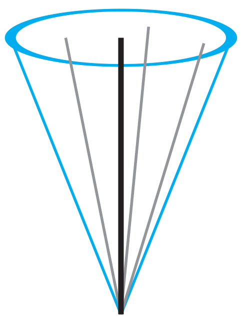
- catching perturbative radiation



Out-of-cone radiation:

$$\langle \delta p_t \rangle \propto -\int_R \frac{d\theta}{\theta} \sim -\log(1/R)$$

- not catching soft background radiation (underlying event)



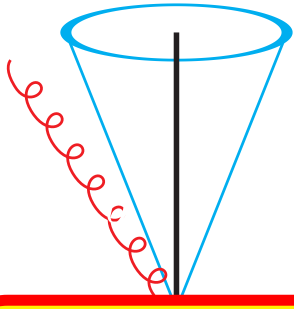
$\langle \delta p_t \rangle \sim$ Soft contents $\propto$ jet area $\sim R^2$

the coefficients depend on the algorithm

Optimisation: underlying idea

Competition between

- catching perturbative radiation

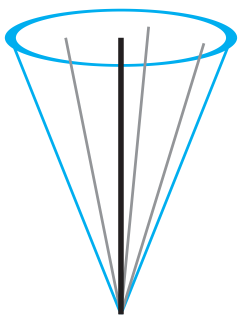


Out-of-cone radiation:

$$\langle \delta p_t \rangle \propto -\int_R \frac{d\theta}{\theta} \sim -\log(1/R)$$

What is the optimal jet definition (algo+R!)?

- not



$\langle \delta p_t \rangle \sim$ Soft contents $\propto$ jet area $\sim R^2$

the coefficients depend on the algorithm

Optimisation: dijet reconstruction

Example process to illustrate various effects:

$$Z' \to q\bar{q} \to 2 \text{ jets}$$

- $M_{Z'}$ can be varied (between 100 GeV and 4 TeV)
- Also valid for $H \to gg$ to study gluon jets
- Reconstruction method:
  - get the 2 hardest jets: j_1 and j_2
  - reconstruct the Z': $m_{Z'} = (j_1 + j_2)^2$

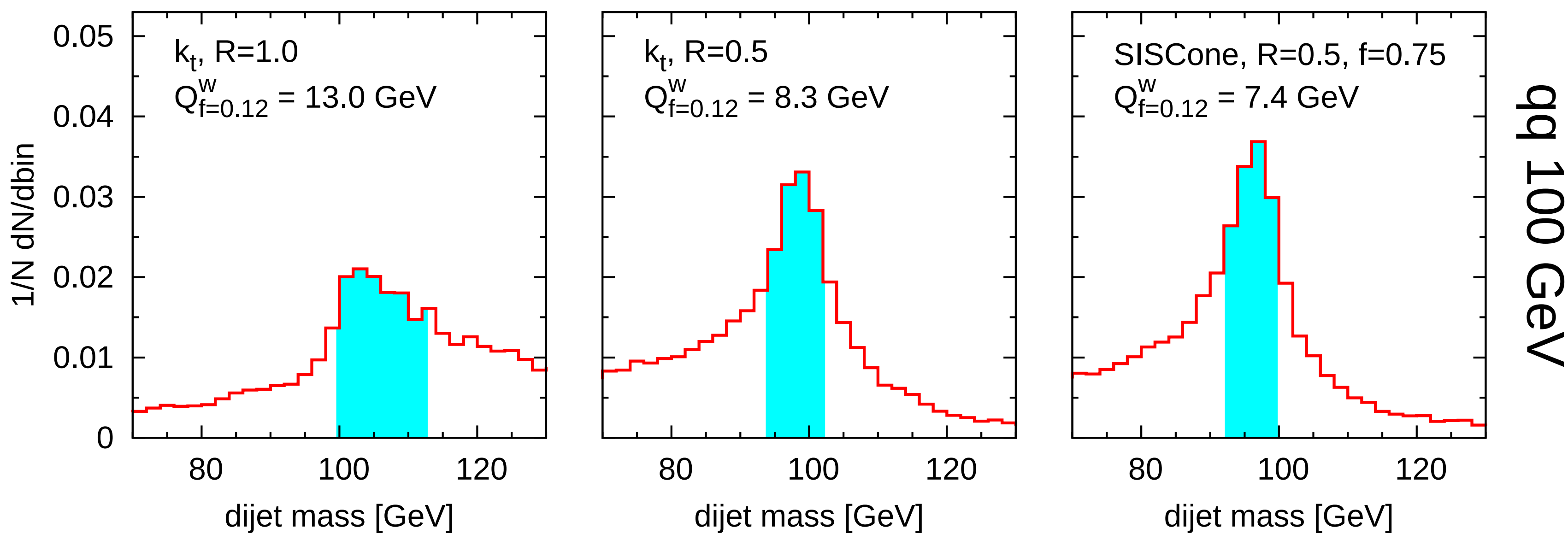
  Look how the mass peak is reconstructed
- Also $t\bar{t}$ with full hadronic decay for multijet tests

Optimisation: quality measure (1)

Measure of the jet reconstruction efficiency:

- Forget about measures related to parton-jet matching
- Forget about fits depending on the shape of the peak

⇒ maximise the signal over background ratio ($S/\sqrt{B}$)
a narrower peak is better.

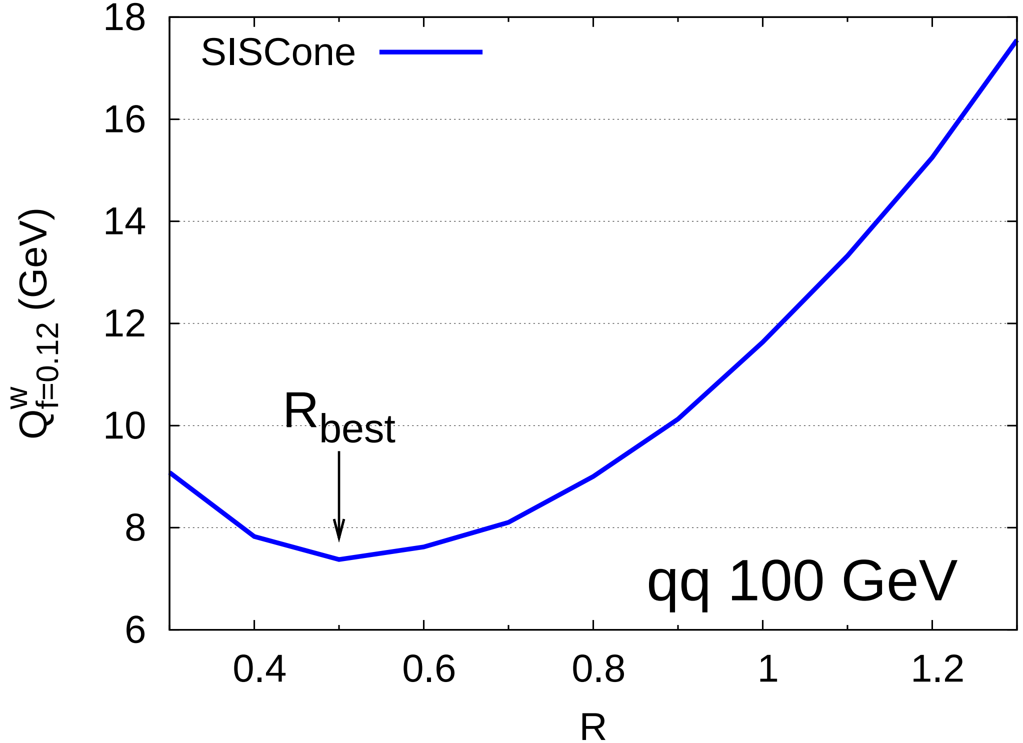


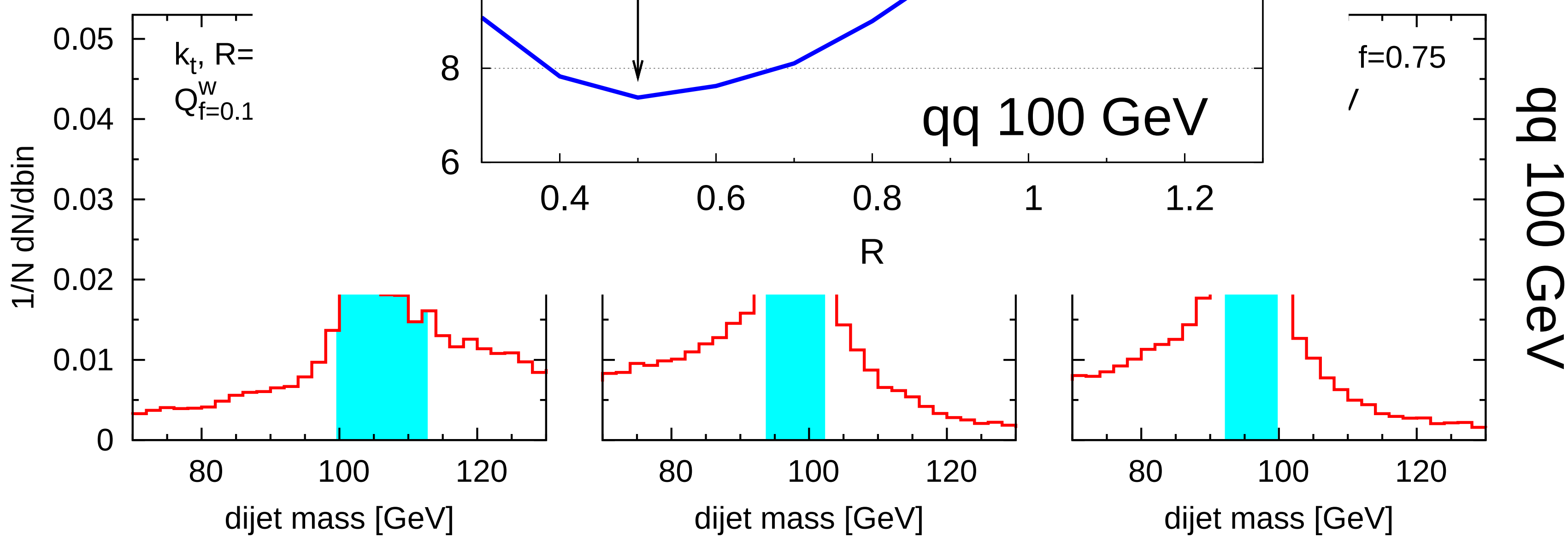
Optimisation: quality measure (1)

Measure of the jet reconstruction efficiency:

- Forget
- Forget

⇒ maxim
a narr





Optimisation: quality measure (2)

Assuming a constant background,

quality measure $\longrightarrow$ effective luminosity ratio

$$\rho_{\mathcal{L}}(\mathrm{JD}_2/\mathrm{JD}_1) = \frac{\mathcal{L} \text{ needed with } \mathrm{JD}_2}{\mathcal{L} \text{ needed with } \mathrm{JD}_1} = \frac{Q^w_{f=z}(\mathrm{JD}_2)}{Q^w_{f=z}(\mathrm{JD}_1)}$$

e.g. $\rho_{\mathcal{L}}(\mathrm{JD}_2/\mathrm{JD}_1) = 2$

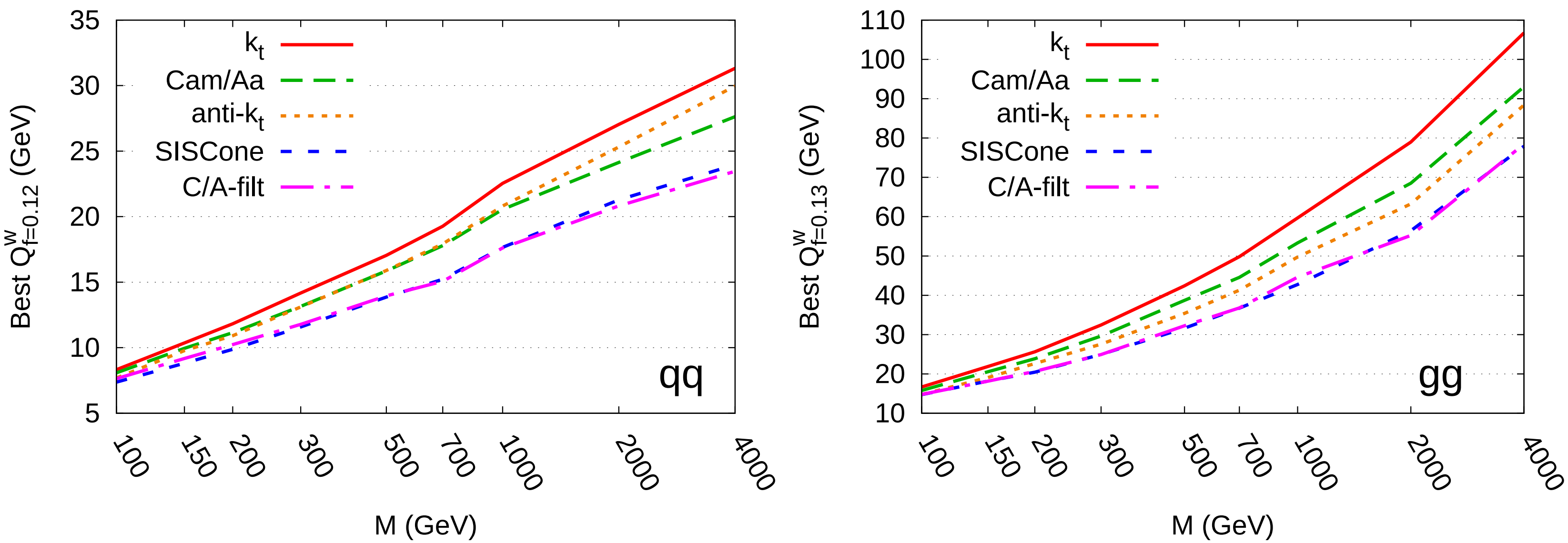
$\Leftrightarrow$ JD_2 requires 2 times the integrated luminosity of JD_1 to achieve the same discriminative power.

Note: results cross-checked with 2 different definitions of the quality measure

Optimisation: best definition

[M.Cacciari, J.Rojo, G.Salam, GS, 08]

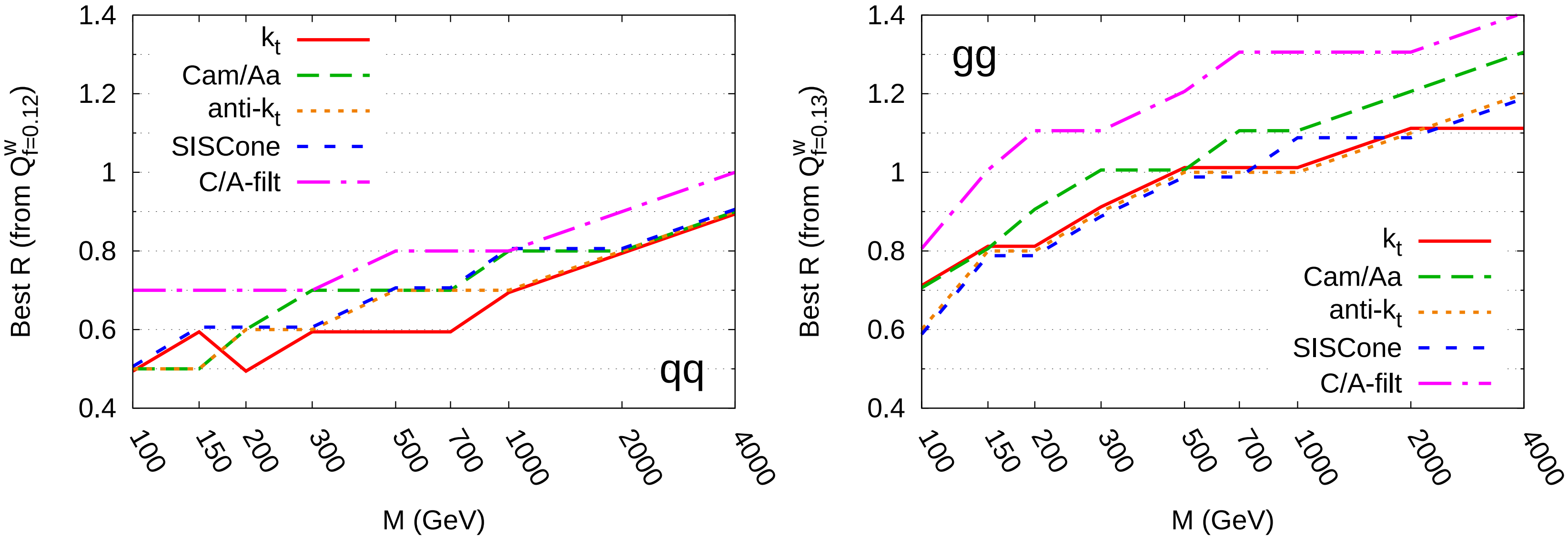
- SISCone and C/A+filt. do slightly better than k_t, C/A or anti-k_t



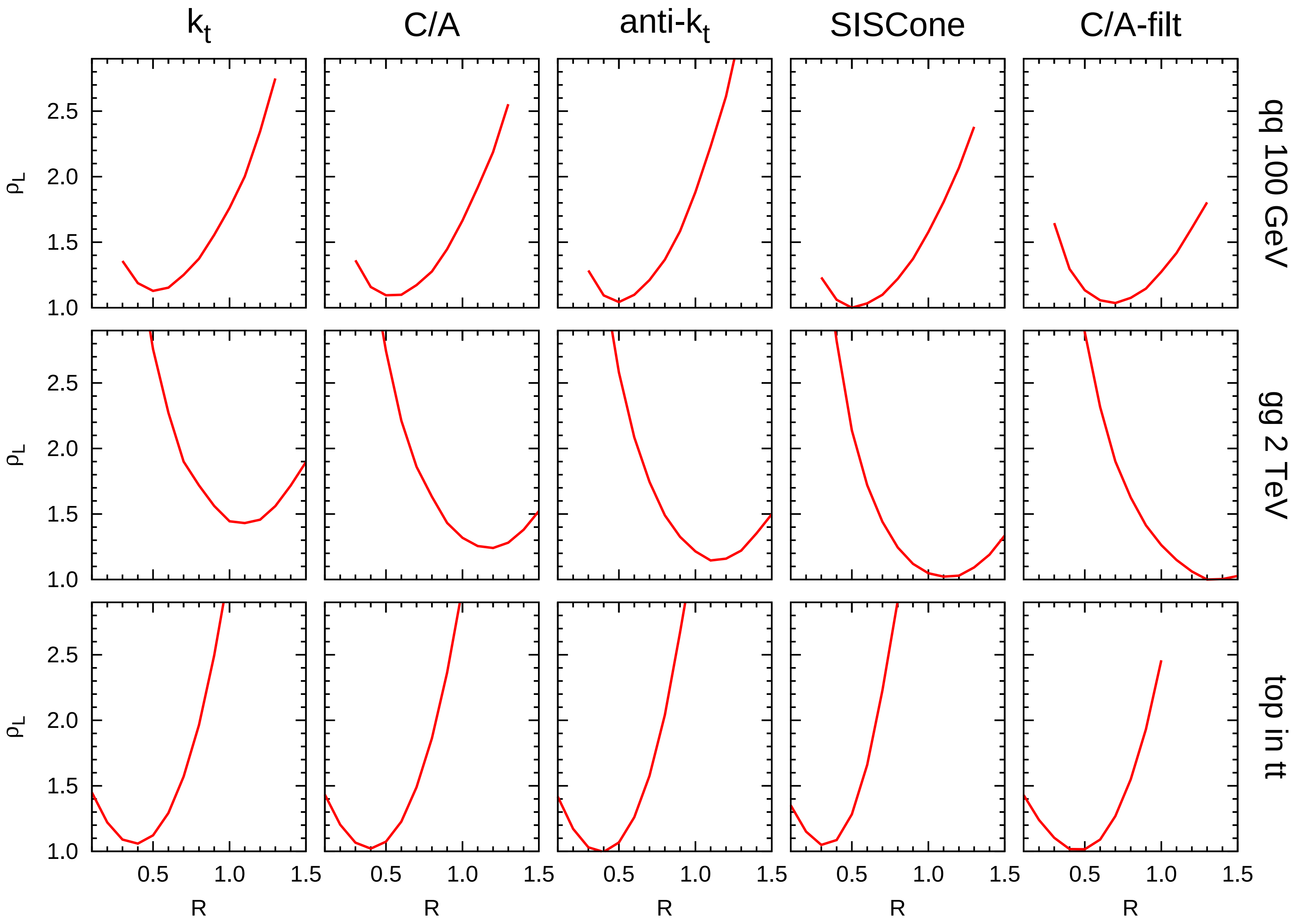
Optimisation: best definition

[M.Cacciari, J.Rojo, G.Salam, GS, 08]

- SISCone and C/A+filt. do slightly better than k_t, C/A or anti-k_t
- $M \nearrow \Rightarrow R_{\rm best} \nearrow$ (and $R_{\rm best}(g) > R_{\rm best}(q)$)

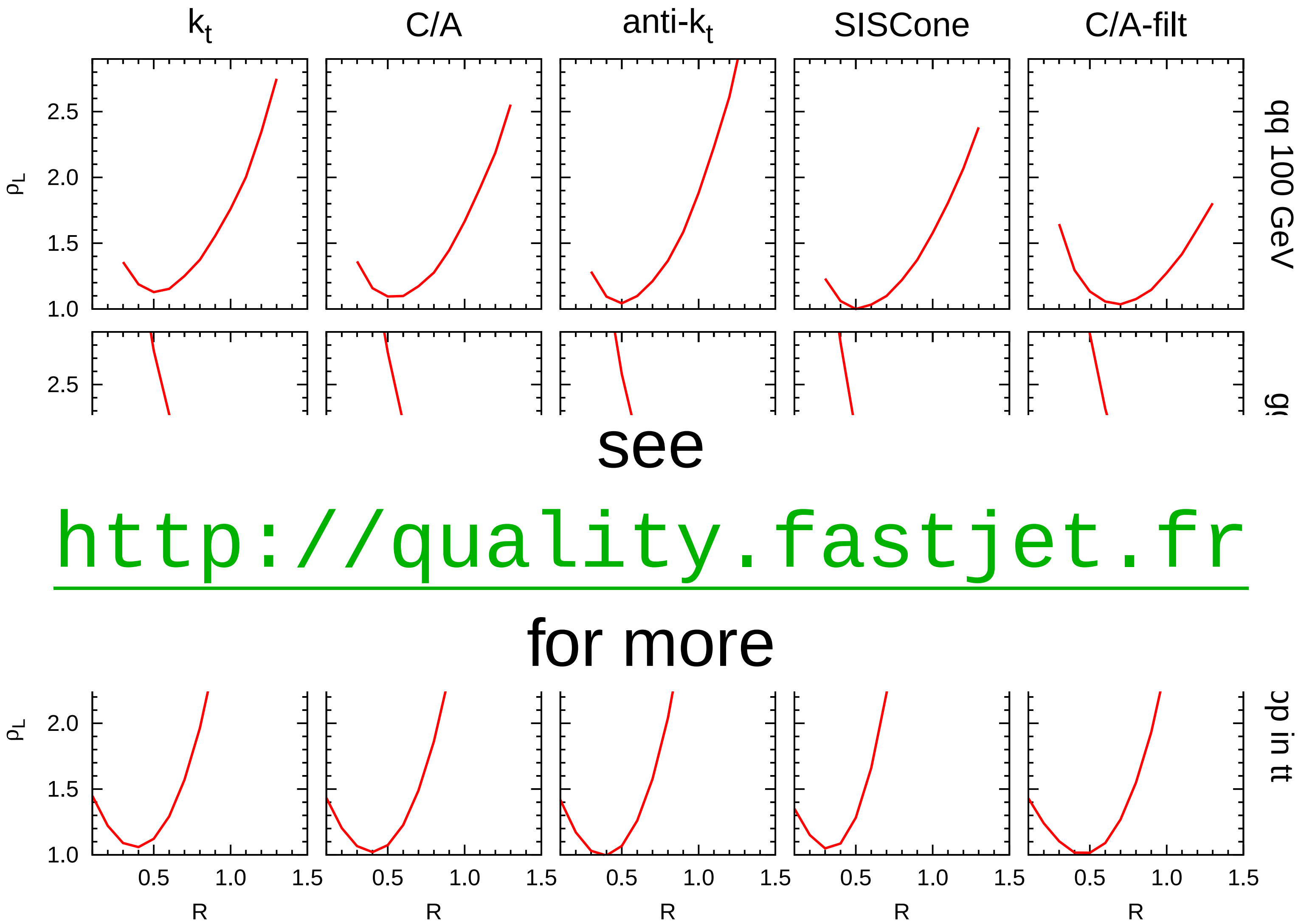


Optimisation: consequences



Using a single jet definition for all processes
may cost a factor ~ 2 in time for early discoveries at the LHC

Optimisation: consequences



see

http://quality.fastjet.fr

for more

Using a single jet definition for all processes

may cost a factor ~ 2 in time for early discoveries at the LHC

New idea #3: jet area and soft background subtraction

Jet areas

[M.Cacciari, G.Salam, GS, 08]

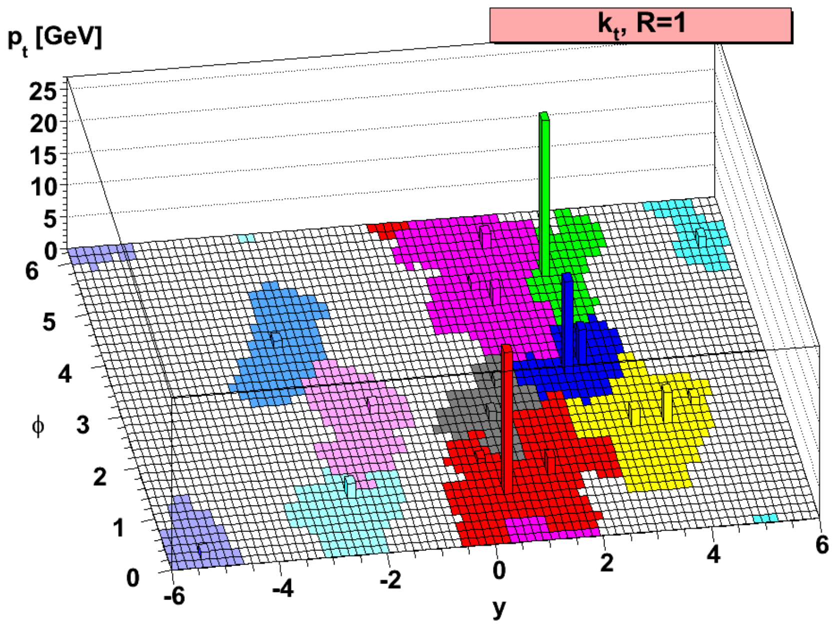
Area $\equiv$ region where the jet catches soft particles

- Recipe: add infinitely soft particles (aka *ghosts*) and see in which jet they are clustered
- 2 methods:
  - Passive area: add one ghost at a time and repeat many times
  - Active area: add a set of ghosts and cluster once
- Idea: ghost $\approx$ background particle
  $\Rightarrow$ active area $\approx$ uniform background
  passive area $\approx$ pointlike background
- Notes:
  - passive = active for large multiplicities
  - require an IR-safe algorithm!
  - generic/universal definition (*e.g.* independent of a calorimeter)

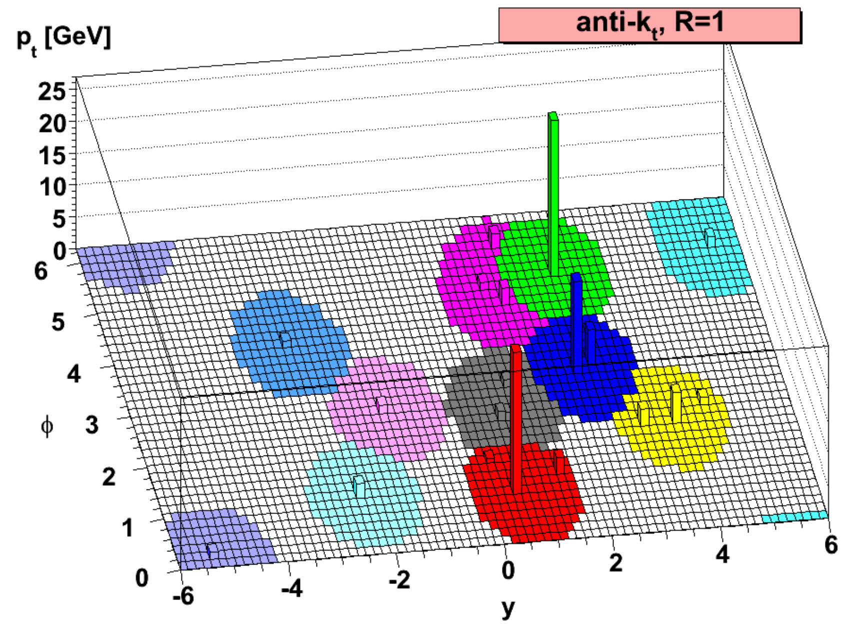
Jet area: examples

Example: active area for a simple event

k_t



anti-k_t



one ghost at every grid cell

Note: analytic control

Example: perturbative expansion of areas (at order α_s)

$$\langle \mathcal{A}(p_t, R) \rangle = \mathcal{A}_0 + \frac{C_{F,A}}{b_0 \pi} \pi R^2 \, d \, \log\left(\frac{\alpha_s(Q_0)}{\alpha_s(R p_t)}\right)$$

- area $\neq \pi R^2$, area $\neq$ const.
- coefficients computable

	$\mathcal{A}_0/(\pi R^2)$		d	
	passive	active	passive	active
k_t	1	0.81	0.56	0.52
Cam/Aachen	1	0.81	0.08	0.08
anti-k_t	1	1	0	0
SISCone	1	1/4	-0.06	0.12

- $Q_0 \equiv$ IR regulator $\propto$ background density

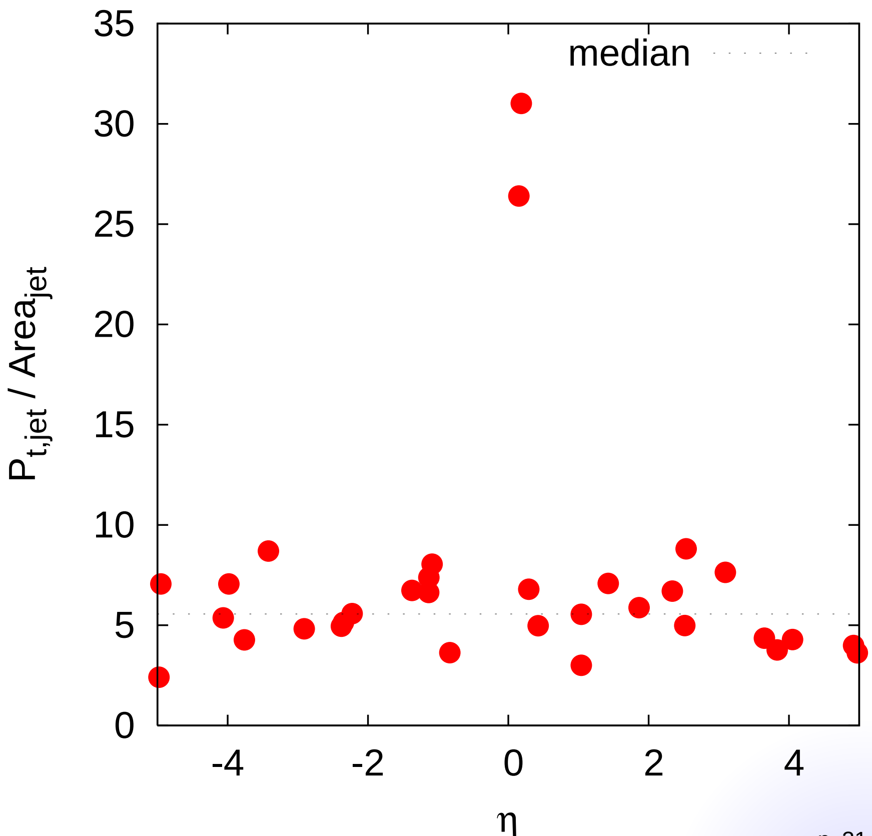
Pileup subtraction *(for uniform backgrounds)*

Basic idea: [M.Cacciari, G.Salam, 08]

$$p_{t,\text{subtracted}} = p_{t,\text{jet}} - \rho_{\text{pileup}} \times \text{Area}_{\text{jet}}$$

- Jet area: [M.Cacciari, G.Salam, G.S., 08]
  - region where the jet catches infinitely soft particles (active/passive)
  - analytic control and understanding in pQCD
- Pileup density per unit area: ρ_{pileup}

  e.g. estimated from the median of $p_{t,\text{jet}}/\text{Area}_{\text{jet}}$



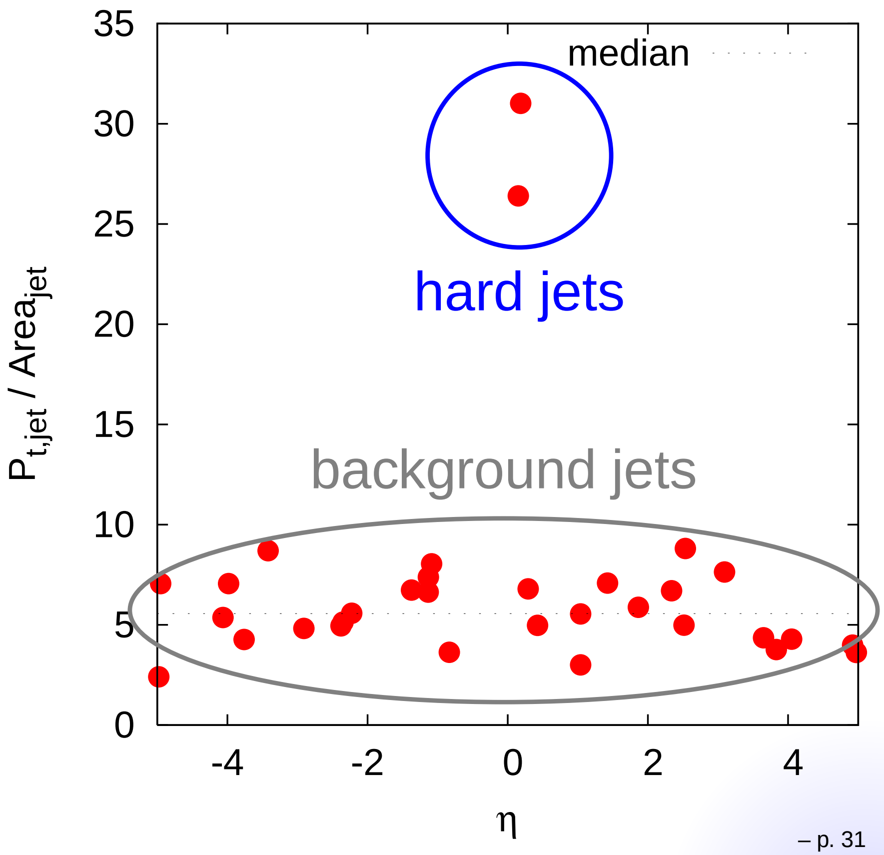
Pileup subtraction (for uniform backgrounds)

Basic idea: [M.Cacciari, G.Salam, 08]

$$p_{t,\text{subtracted}} = p_{t,\text{jet}} - \rho_{\text{pileup}} \times \text{Area}_{\text{jet}}$$

- Jet area: [M.Cacciari, G.Salam, G.S., 08]
  - region where the jet catches infinitely soft particles (active/passive)
  - analytic control and understanding in pQCD
- Pileup density per unit area: ρ_{pileup}

  e.g. estimated from the median of $p_{t,\text{jet}}/\text{Area}_{\text{jet}}$



Pileup subtraction *(for uniform backgrounds)*

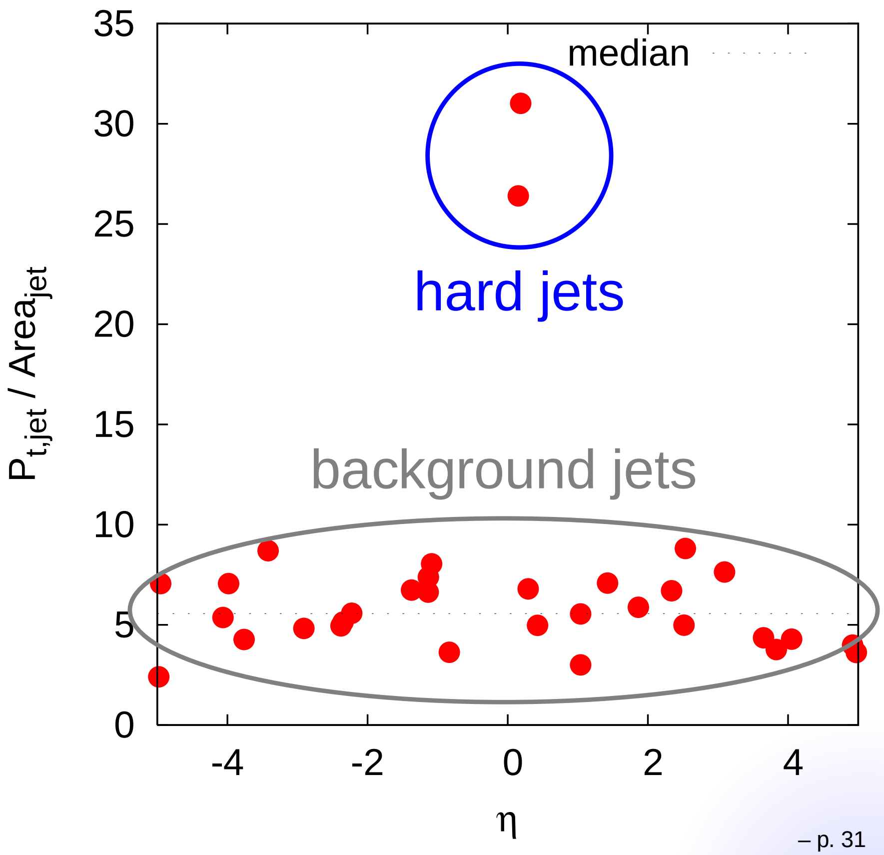
Basic idea: [M.Cacciari, G.Salam, 08]

$$p_{t,\text{subtracted}} = p_{t,\text{jet}} - \rho_{\text{pileup}} \times \text{Area}_{\text{jet}}$$

- Jet area: [M.Cacciari, G.Salam, G.S., 08]
  - region where the jet catches infinitely soft particles (active/passive)
  - analytic control and understanding in pQCD
- Pileup density per unit area: ρ_{pileup}

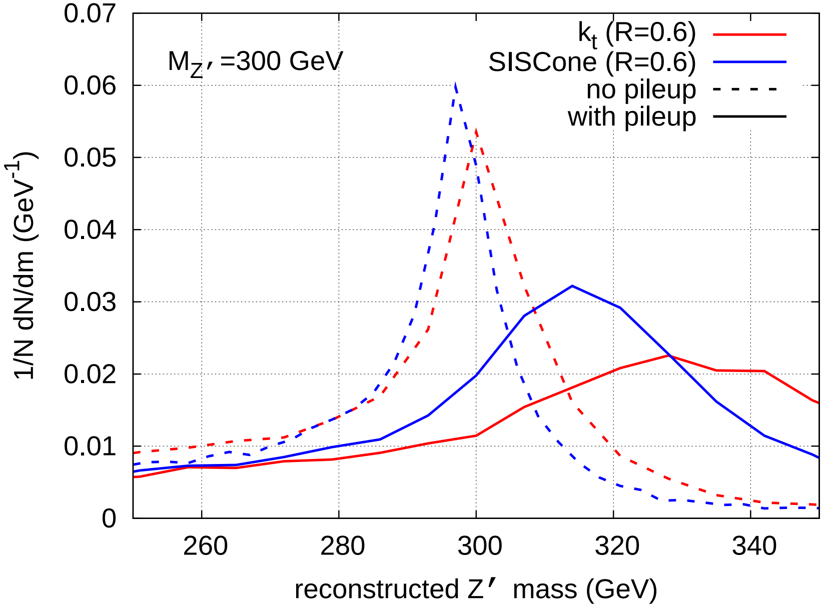
  e.g. estimated from the median of $p_{t,\text{jet}}/\text{Area}_{\text{jet}}$

implemented in FastJet
on an event-by-event basis



Effect on dijet reconstruction

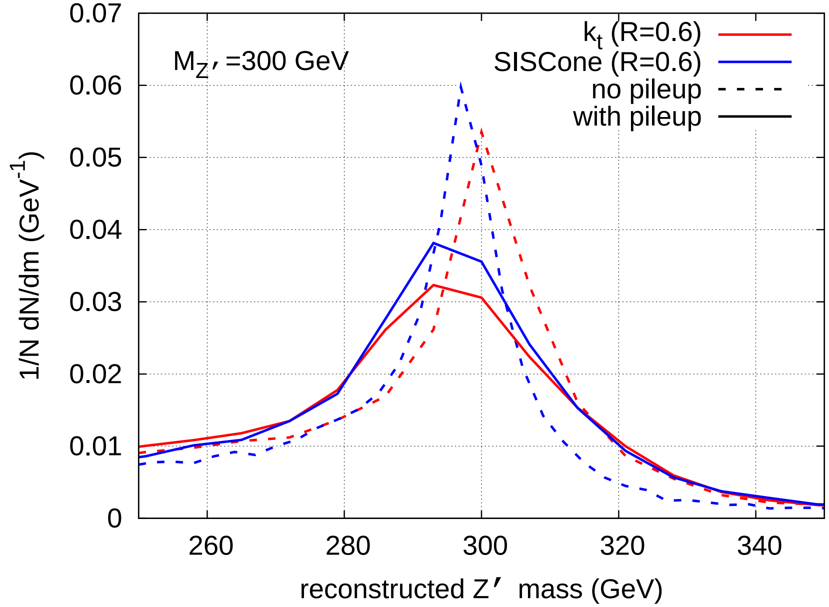
Pileup unsubtracted



width = 29.5 GeV

width = 21.0 GeV

pileup subtracted



width = 21.0 GeV

width = 17.7 GeV

✓ position reasonnable

✓ dispersion reduced (thanks to the event-by-event approach)

✓ used by STAR for the first jet analysis in heavy-ions

Summary (1)

Message #1:

Use infrared-and-collinear-safe algorithms

ATLAS Cone CDF/D0 MidPoint	$\rightarrow$	SISCone	$\surd$ fast $\surd$ safe
CMS It. Cone	$\rightarrow$	anti-k_t	$\surd$ fast $\surd$ safe

Important to benefit fully from pQCD multilegs/multiloops calculations

Summary (2)

Message #2:

correct tools $\Rightarrow$ new ideas, new concepts

$\Rightarrow$ new generation of jet definitions

- jet areas $\longrightarrow$ pileup and HI background subtraction
- jet substructure improves reconstruction (Higgs, top, SUSY, ...)

Message #3:

keep some flexibility in the jet definition choice

- optimisation $\longrightarrow$ luminosity gains for LHC searches
- different approaches $\longrightarrow$ better understanding of HI collisions

backup slides

The SISCone search for stable cones

- Solution: use a seedless approach, find ALL stable cones
- Naive approach: check stability of each subset of particle

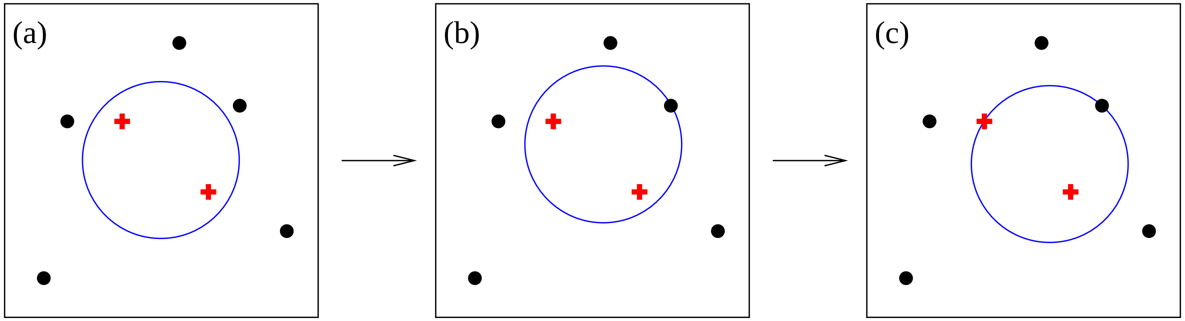
The SISCone search for stable cones

- Solution: use a seedless approach, find ALL stable cones
- Naive approach: check stability of each subset of particle
  Complexity is $\mathcal{O}\left(N2^N\right)$
  $\Rightarrow$ definitely unrealistic: 10^{17} years for $N = 100$
- Midpoint complexity: $\mathcal{O}\left(N^3\right)$

The SISCone search for stable cones

- Solution: use a seedless approach, find ALL stable cones
- Midpoint complexity: $\mathcal{O}\left(N^3\right)$

Idea: use geometric arguments



- Each enclosure can be moved (in any dir.) until it touches a point
- ... then rotated until it touches a second one

$\Rightarrow$ Enumerate all pairs of particles
with 2 circle orientations and 4 possible inclusion/exclusion
$\longrightarrow$ find all enclosures

The SISCone search for stable cones

- Solution: use a seedless approach, find ALL stable cones
- Midpoint complexity: $\mathcal{O}\left(N^3\right)$

Idea: use geometric arguments

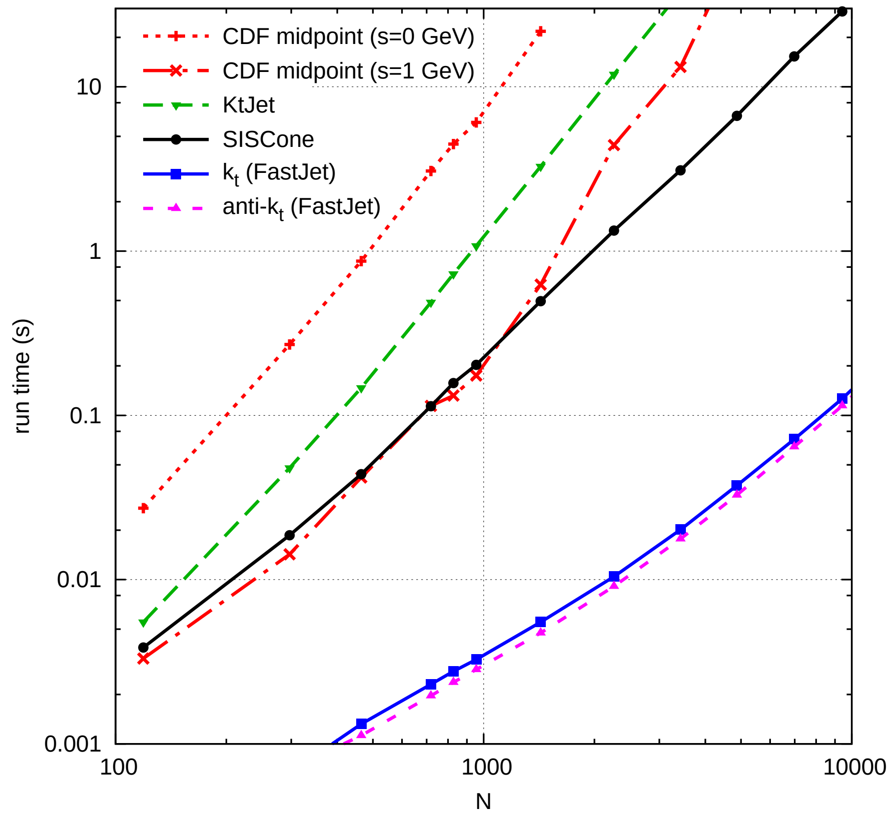
$\Rightarrow$ Enumerate all pairs of particles
with 2 circle orientations and 4 possible inclusion/exclusion
$\longrightarrow$ find all enclosures

- Complexity: $\mathcal{O}\left(N^3\right)$, with improvements: $\mathcal{O}\left(N^2 \log(N)\right)$

$\longrightarrow$ C++ implementation: Seedless Infrared-Safe Cone algorithm (SISCone)
G.Salam, G.S., JHEP 04 (2007) 086; http://projects.hepforge.org/siscone

NB.: also available from FastJet
[M.Cacciari, G.Salam, G.S.]; http://www.fastjet.fr

Algorithm timings



- Recombination algorithms very fast

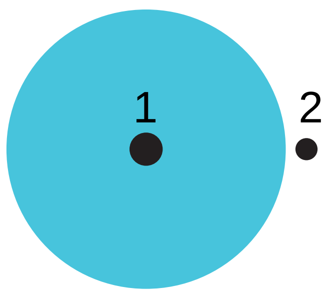
[M. Cacciari, G. Salam, 06]

- SISCone not slower than Midpoint (even with a 1 GeV seed threshold)

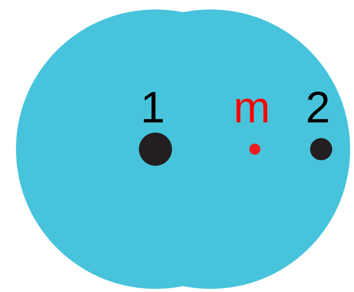
A technical point: Back-reaction

Additional soft background has 2 effects:

- Throw soft particles in the hard jet: dealt with by subtraction
- Modify the hard scattering (back-reaction)
  - can be pointlike or diffuse
  - gain:

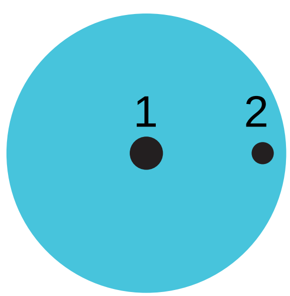


no medium: $p_t = p_{t1}$

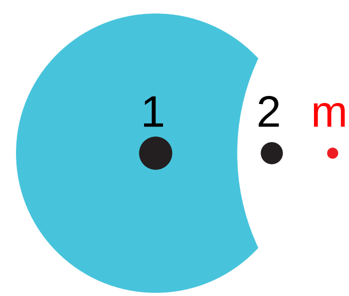


medium: $p_t = p_{t1} + p_{t2} + p_{tm}$

  - loss:



no medium: $p_t = p_{t1} + p_{t2}$

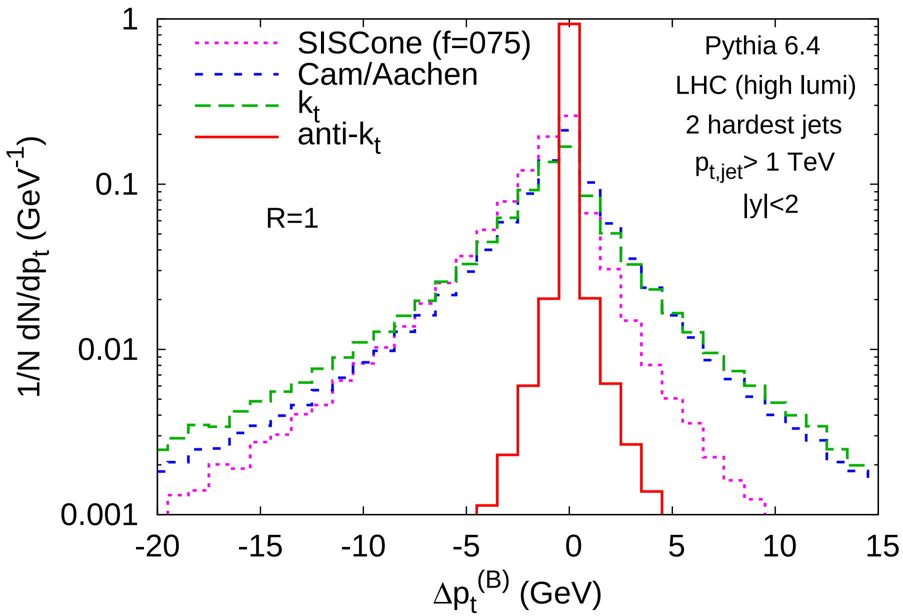


medium: $p_t = p_{t1} + p_{tm}$

A technical point: Back-reaction

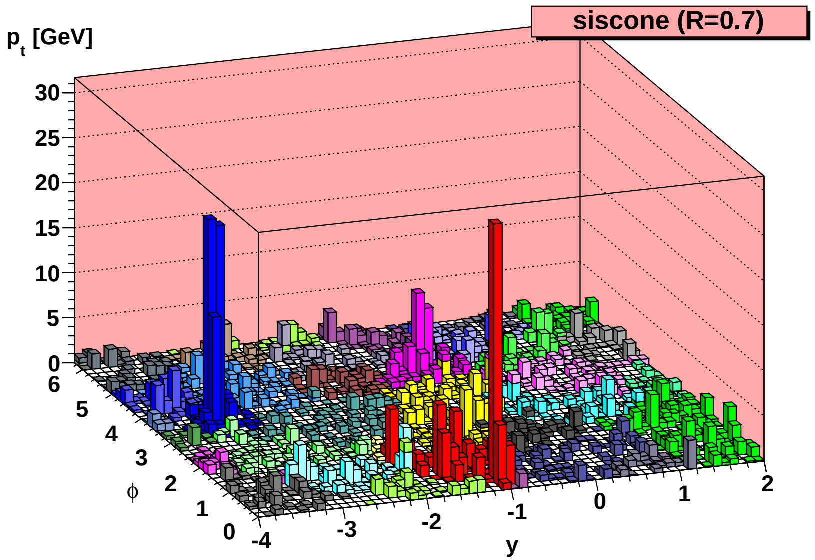
Additional soft background has 2 effects:

- Throw soft particles in the hard jet: dealt with by subtraction
- Modify the hard scattering (back-reaction)
  - can be pointlike or diffuse
  - tractable analytically (similar to areas)
  - $k_t \gtrsim$ Cambridge > SISCone $\gg$ anti-k_t

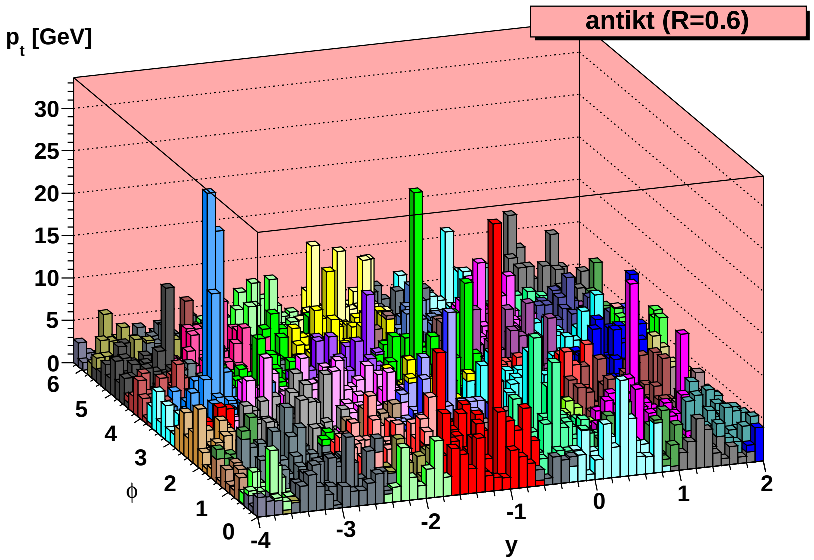


Example: application to HI collisions

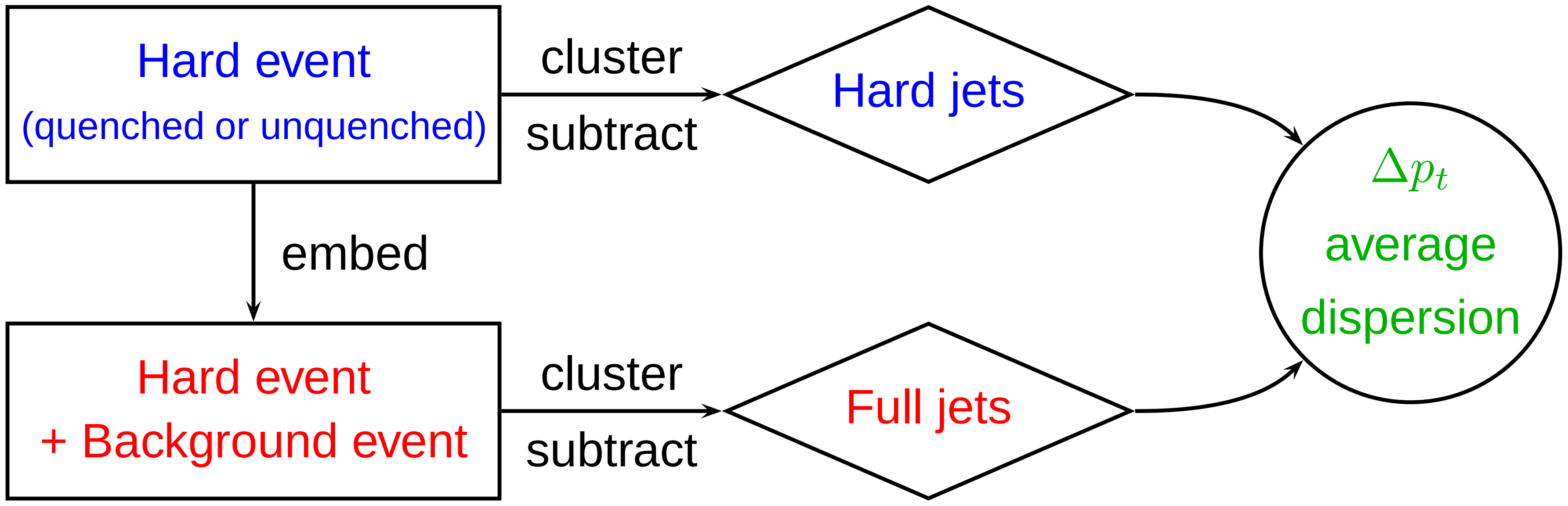
pp + pileup



AA

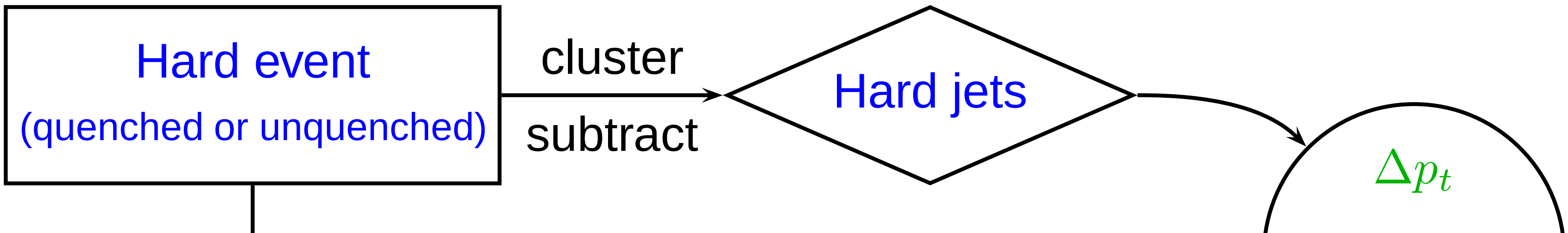


Framework for study



- Hard event: Pythia(v6.4) or Pythia(v6.4)+PyQuen(v1.5)
- Background: Hydjet(v1.5) (others under study)
- Analysis: FastJet(v2.4)
  Ideally: smallest Δp_t shift, smallest Δp_t dispersion
- Note: in what follows, R fixed to 0.4

Framework for study



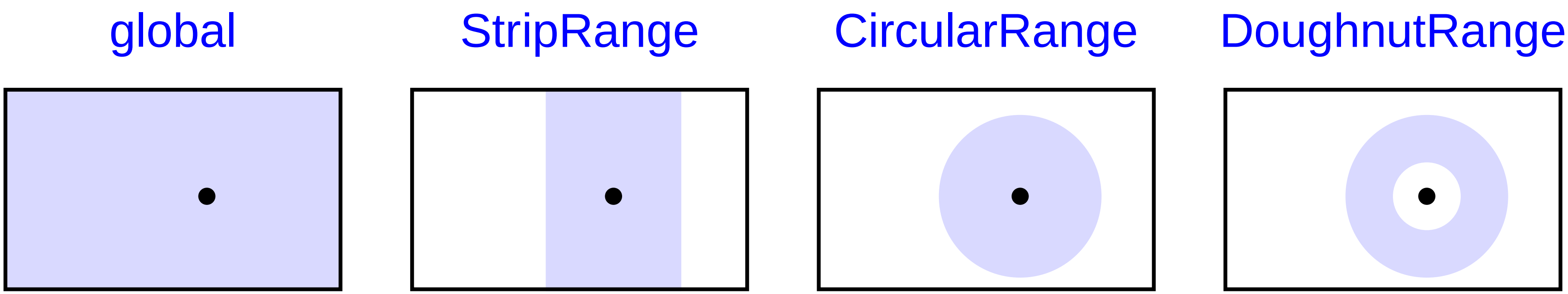
- Generic trends under control
- Final numbers may change

[M.Cacciari, J.Rojo, G.Salam, GS, in prep.]

- H
- B
- Analysis: FastJet(v2.4)
  Ideally: smallest Δp_t shift, smallest Δp_t dispersion
- Note: in what follows, R fixed to 0.4

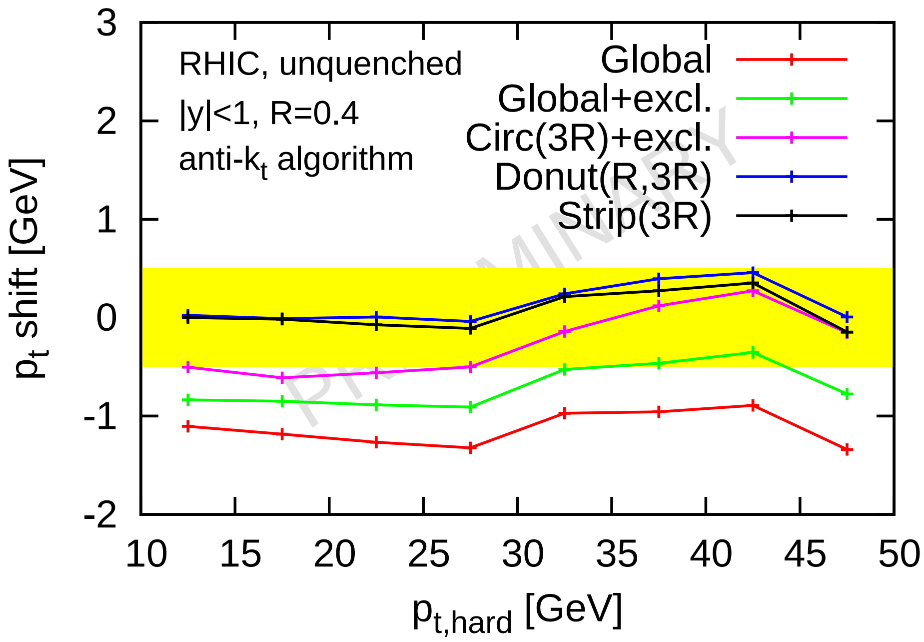
Idea #1: use a local range to compute $\rho_{\rm bkg}$

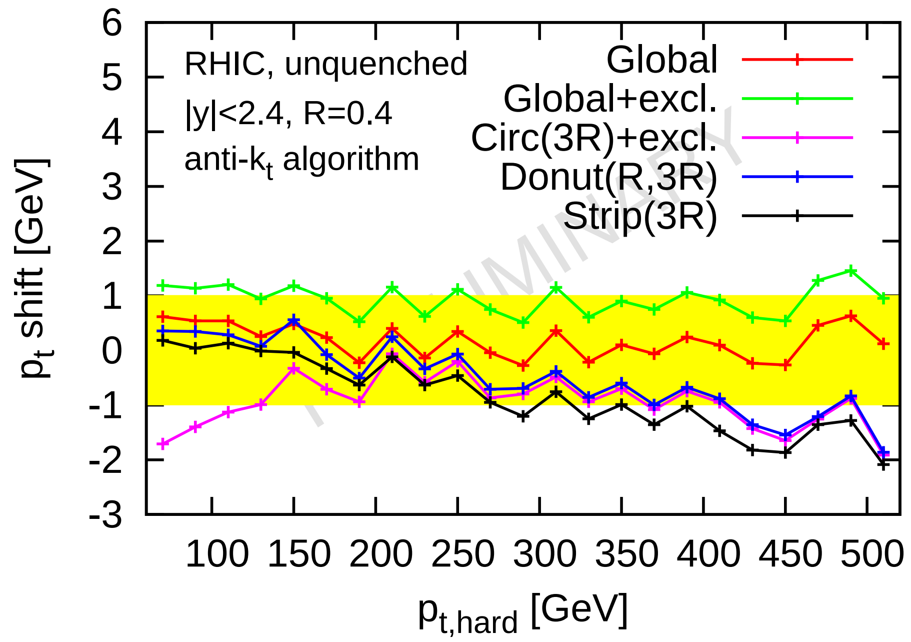
- Fluctuating background
  $\longrightarrow$ determine the background density $\rho_{\rm bkg}$
  from jets in the vicinity of the jet we want to subtract



- Exclude the hardest jets from the determination of $\rho_{\rm bkg}$
  $\Rightarrow$ reduce the bias in the computation median

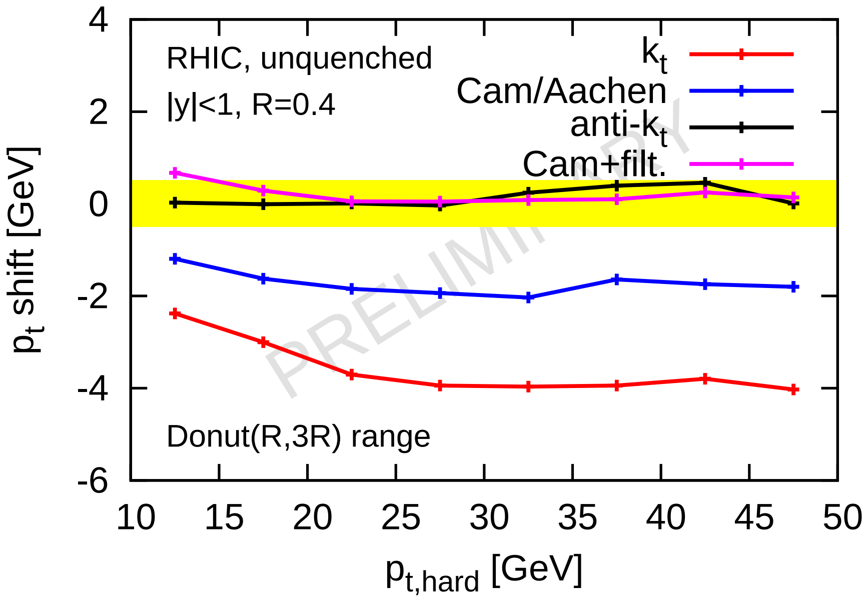
Effect of choosing a local range





- effect $\sim$ 0.5-1 GeV
- differences between local ranges $\longrightarrow$ uncertainty
- for limited acceptance, global range $\approx$ local range
- analytic control would be nice

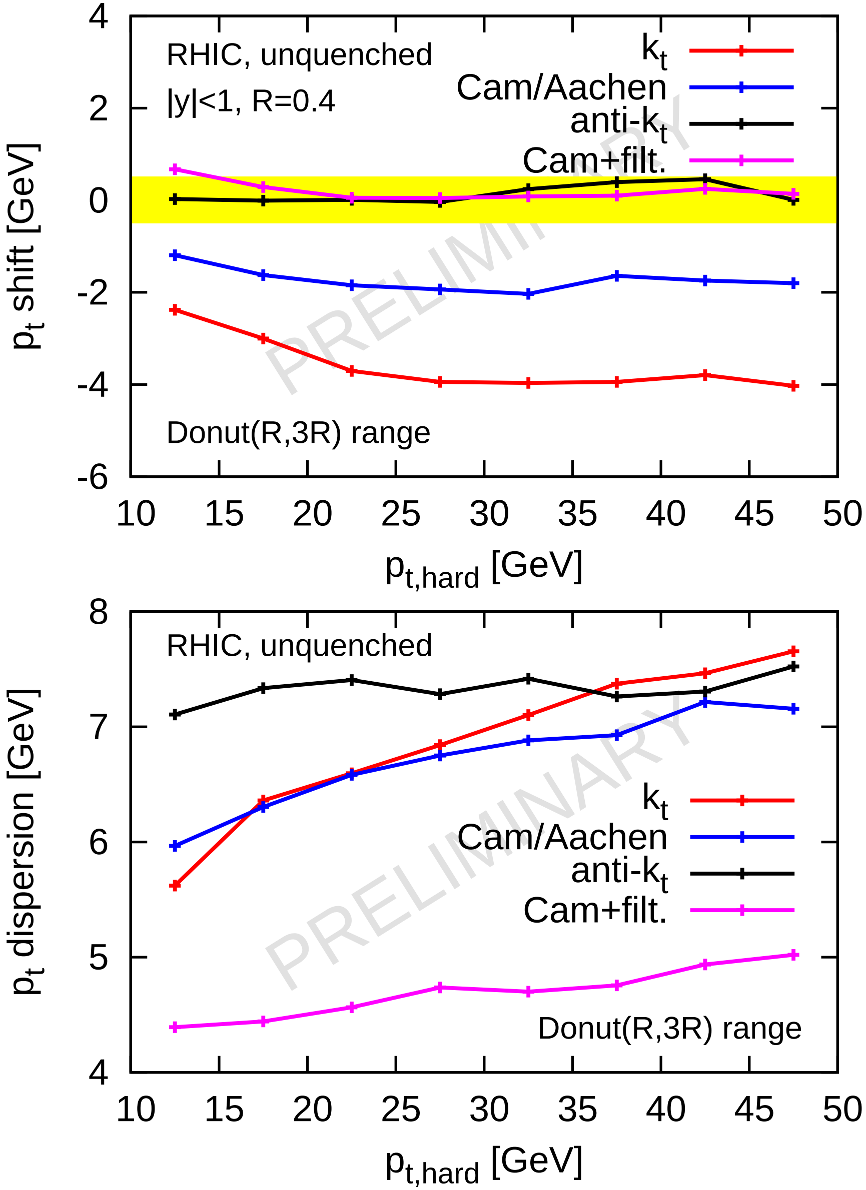
Results: RHIC kinematics



- average p_t shift:

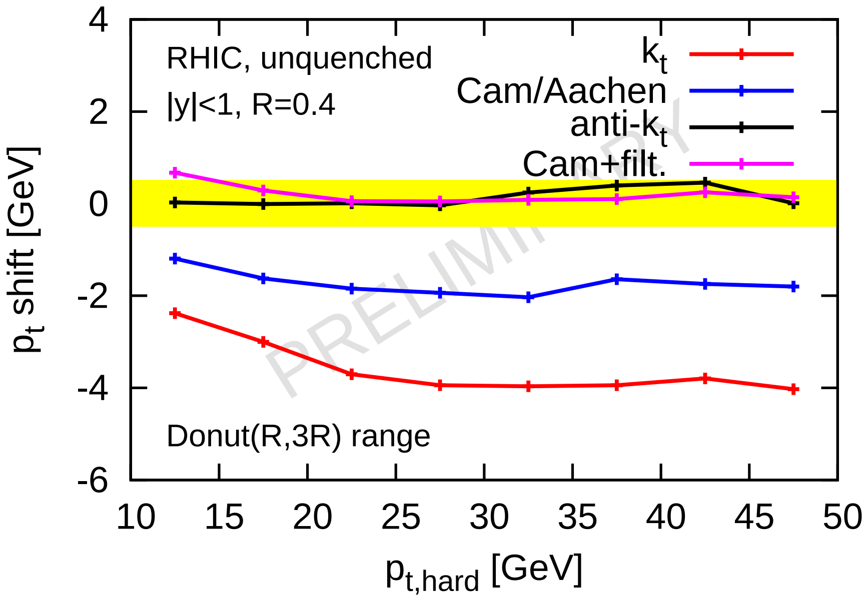
  anti-k_t and C/A+filt. Ok

Results: RHIC kinematics

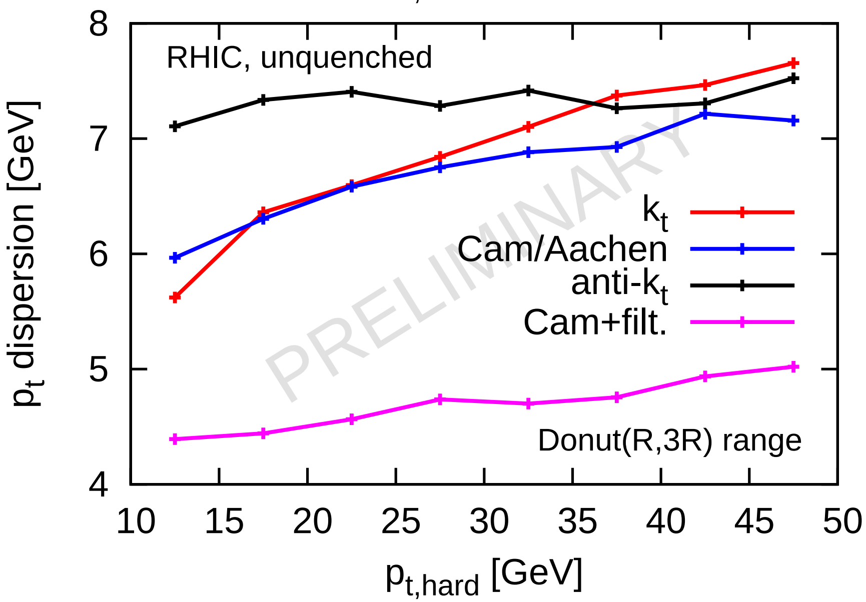


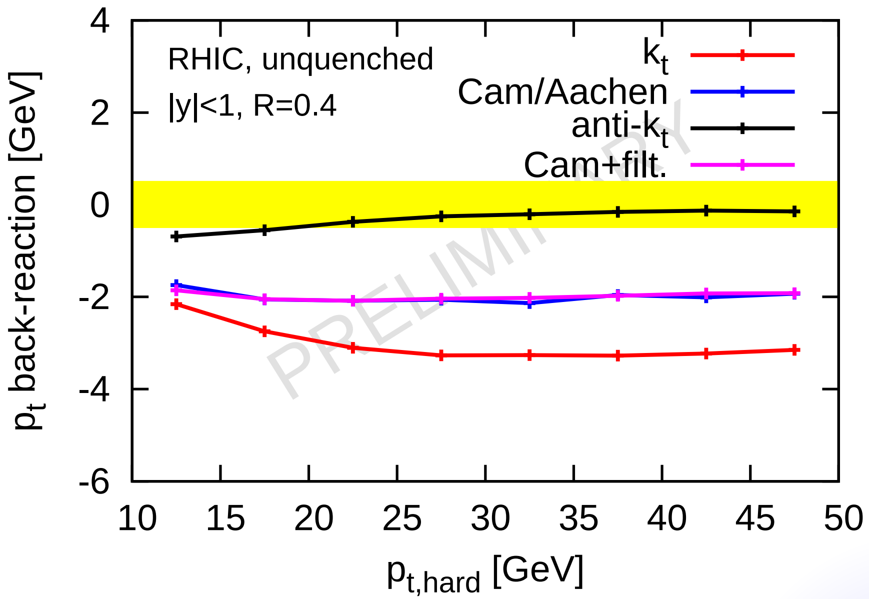
- average p_t shift:
  anti-k_t and C/A+filt. Ok
- p_t shift dispersion:
  C/A+filt. better

Results: RHIC kinematics



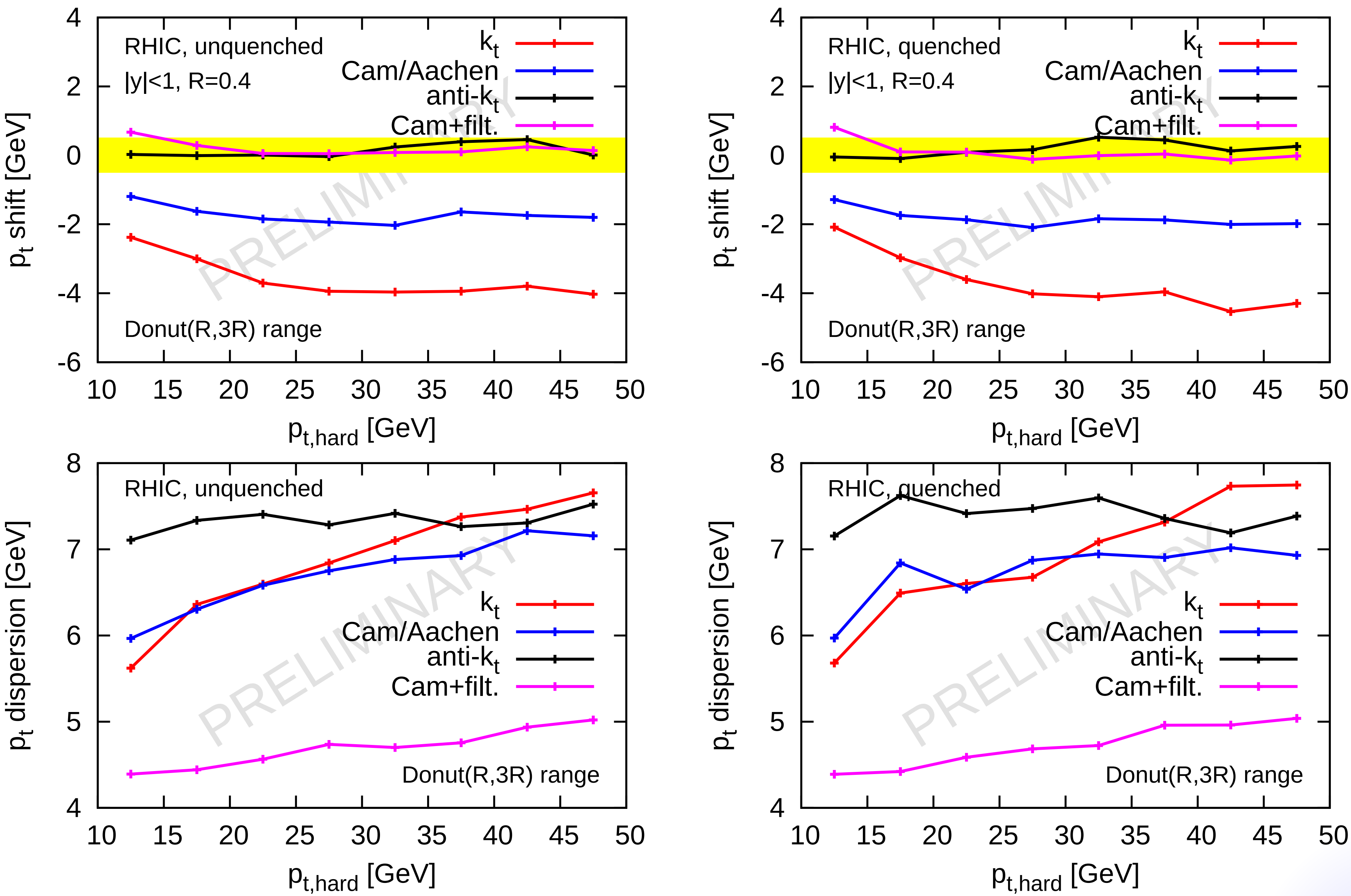
- average p_t shift:
  anti-k_t and C/A+filt. Ok
- p_t shift dispersion:
  C/A+filt. better
- watch out C/A+filt. average:
  back-reaction compensated



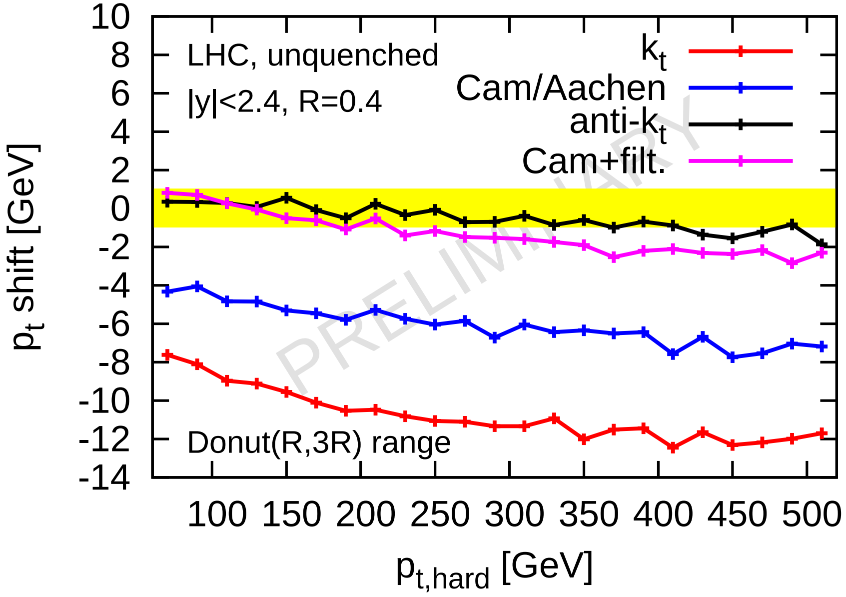


Results: RHIC kinematics – quenching

Performances not much affected by quenching (need more models)



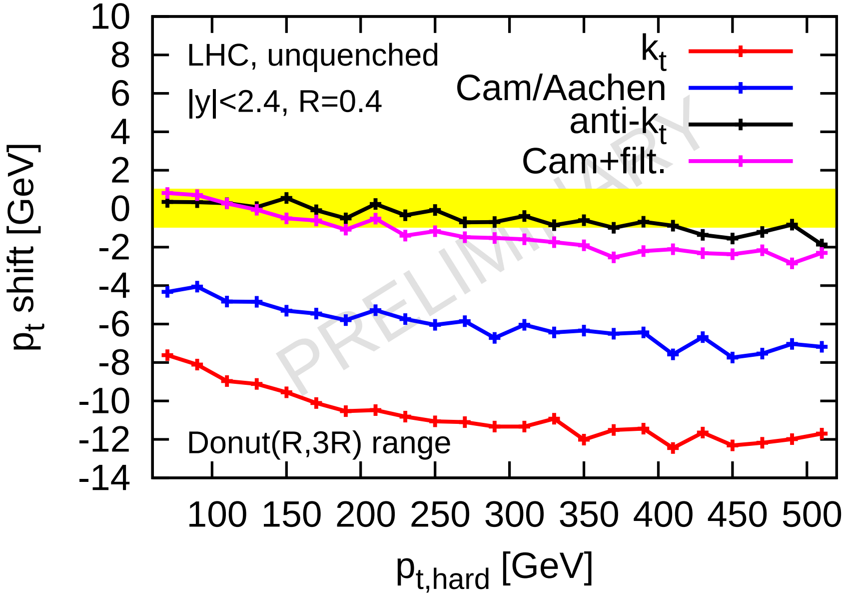
Results: LHC kinematics

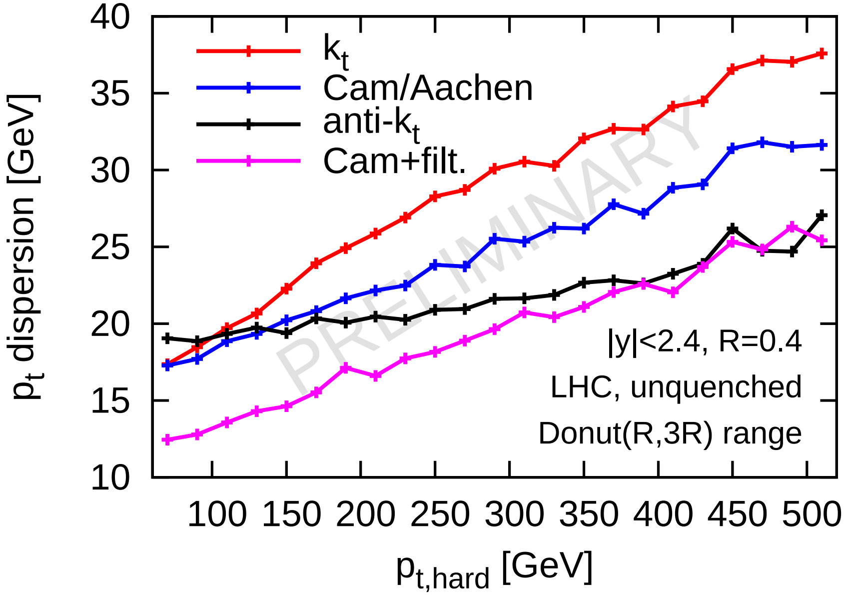


- average p_t shift:

  anti-k_t and C/A+filt. Ok

Results: LHC kinematics





- average p_t shift:
  anti-k_t and C/A+filt. Ok

- p_t shift dispersion:
  C/A+filt. better
  anti-k_t Ok

Results: LHC kinematics – quenching

Large quenching effect   but anti-k_t's rigidity plays for it