Jets playing hide and seek

Grégory Soyez

Brookhaven National Laboratory

Los Alamos National Laboratory — May 12 2009 — Los Alamos, NM, USA
Plan

- $e^+e^-$ collisions:
  - the most simple case

- $pp$ collisions:
  - jets in hadronic environments
    - $\rightarrow$ 20 years of jet definitions

- pileup:
  - jets in the background
    - $\rightarrow$ jet areas, background subtraction

- $AA$:
  - more background!
    - $\rightarrow$ improved techniques
QCD collinear divergence $\rightarrow$ collimated showers

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

Example: LEP (OPAL) events

Idea: jet $\equiv$ collimated shower $\simeq$ initial hard partons
**Foreword: what are jets?**

QCD collinear divergence → collimated showers

\[ dP \propto \alpha_s \frac{d\theta}{\theta} \]

Example: LEP (OPAL) events

---

**In practice: use of a jet definition**

particles \( \{p_i\} \) → jet definition → jets \( \{j_k\} \)

---

<table>
<thead>
<tr>
<th>2 jets</th>
<th>3 jets</th>
</tr>
</thead>
</table>

**Idea:** jet \( \equiv \) collimated shower \( \simeq \) initial hard partons
Foreword: an illustrative example

Example process to illustrate various effects:

\[ Z' \rightarrow q\bar{q} \rightarrow 2 \text{ jets} \]

- \( M_{Z'} = 300 \text{ GeV} \) (width < 1 GeV)

- 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
$e^+e^-$ collisions: the most simple case
Recombination algorithms

Recipe:

- find the pair \((i, j)\) with the smallest \(d_{ij}\) distance
- recombine \(i\) and \(j\)
- repeat (until objects more than \(R\) apart)

Distance: \(k_t\) algorithm

\[ d_{ij} = \min(E_{i}^2, E_{j}^2)[1 - \cos(\theta_{ij})] \]

\(d_{ij} \to 0\) for soft and collinear splittings

\[ \frac{dP}{d\theta dp_t} = \alpha_s \frac{d\theta}{\theta} \frac{dp_t}{p_t} \]

without the prefactor: Cambridge/Aachen
**Part 2**

**pp collisions:** jets in hadronic environments

\[ e^+ e^- \]

\[
d_{ij} = \min(E_i^2, E_j^2)[1 - \cos(\theta_{ij})]
\]

\[
d_{ij} = \min(k_{t,i}^2, k_{t,j}^2)(\Delta y_{ij}^2 + \Delta \phi_{ij}^2)
\]
Our example: moving to jets in \textit{pp}

\textbf{e}^+\textbf{e}^-

\textbf{pp}

\begin{figure}
\centering
\includegraphics[width=\textwidth]{example.png}
\end{figure}

width = 0.9 GeV

width = 19.7 GeV
Our example: moving to jets in $pp$

$e^+ e^-$

$pp$

width = 0.9 GeV

width = 19.7 GeV

width = 14.2 GeV

Reduce $R$
Our example: moving to jets in $pp$

$e^+e^-$

$pp$

$M_{Z'} = 300$ GeV

$k_t (R=1)$

$k_t (R=0.6)$

CDF Midpoint (R=0.6)

width = 0.9 GeV

width = 19.7 GeV

width = 14.2 GeV

width = 11.8 GeV

- Reduce $R$
- Use the cone algorithm
**The cone algorithm**

- **Idea**: find directions of energy flow

- **Stable cone** (fixed radius $R$): sum of all constituents points in the direction of the centre

- **Search**: iterate from initial directions (seeds)

- **Stable cones → jets**: deal with overlaps
  - **Solution 1**: split/merge depending on the amount of overlap
    Examples: CDF JetClu, CDF MidPoint, D0 runII, ATLAS Cone
  - **Solution 2**: progressive removal starting from the hardest seed
    Examples: CMS Iterative Cone

Benchmark: circular/rigid jets
1990: SNOWMASS accords – constraints to fulfil

Several important properties that should be met by a jet definition are [3]:

1. Simple to implement in an experimental analysis;
2. Simple to implement in the theoretical calculation;
3. Defined at any order of perturbation theory;
4. Yields finite cross section at any order of perturbation theory;
5. Yields a cross section that is relatively insensitive to hadronization.

Infrared and collinear (IRC) safety:

✓ recombination algorithms: Ok

✗ Cone: problem inherent to the “seeded” search of stable cones
New algorithms

- **Cone with split–merge**
  - **Idea**: Find an efficient seedless implementation that provably identifies all stable cones
  - **Solution**: **SISCon**e: Seedless Infrared-Safe Cone
    
    [G.Salam, GS, 07]

- **Cone with progressive removal**
  - **Idea**: keep the rigidity of the jets + restore IRC-safety
  - **Solution**: **anti-**$k_t$:

    \[ d_{ij} = \min(k_{t,i}^{-2}, k_{t,j}^{-2})(\Delta y_{ij}^2 + \Delta \phi_{ij}^2) \]

    [M.Cacciari, G.Salam, GS, 08]
**Impact of these new algorithms**

### Summary

<table>
<thead>
<tr>
<th>Unsafe</th>
<th>Safe</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMS Iterative Cone</td>
<td>$k_t$</td>
</tr>
<tr>
<td>ATLAS Cone</td>
<td>Cambridge/Aachen</td>
</tr>
<tr>
<td>CDF JetClu</td>
<td>anti-$k_t$</td>
</tr>
<tr>
<td>CDF MidPoint</td>
<td>SISCone</td>
</tr>
<tr>
<td>D0 run II Cone</td>
<td></td>
</tr>
</tbody>
</table>

### Impact

<table>
<thead>
<tr>
<th>Observable</th>
<th>Last meaningful order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inclusive jet cross section</td>
<td>NLO</td>
</tr>
<tr>
<td>3 jet cross section</td>
<td>LO (NLO in NLOJet)</td>
</tr>
<tr>
<td>$W/Z/H + 2$ jet cross sect.</td>
<td>LO (NLO in MCFM)</td>
</tr>
<tr>
<td>jet masses in 3 jets</td>
<td>none (LO in NLOJet)</td>
</tr>
</tbody>
</table>

Huge effort ($\sim$ 100 M$\$$) to compute processes in pQCD
Impact of these new algorithms

Summary

Unsafe

- CMS Iterative Cone
- ATLAS Cone
- CDF JetClu
- CDF MidPoint
- D0 run II Cone

Safe

- $k_t$
- Cambridge/Aachen
- anti-$k_t$
- SISCones

Impact

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

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

Huge effort (∼100 M$) to compute processes in pQCD
Our example: safe jet definitions

1990

M_{Z'} = 300 \text{ GeV}

K_t (R=1)

K_t (R=0.6)

CDF Midpoint (R=0.6)

width = 19.7/14.2 \text{ GeV}

width = 11.8 \text{ GeV}

For the width
SIS Cone slightly preferred

2009

M_{Z'} = 300 \text{ GeV}

Cam/Aachen

anti-k_t

SIS Cone

width = 14.2 \text{ GeV}

width = 12.2 \text{ GeV}

width = 13.1 \text{ GeV}

width = 11.6 \text{ GeV}
Part 3

pileup: jets in the background

No pileup

With pileup
Our example: the effect of pileup

No pileup

width = 14.2 GeV
width = 11.6 GeV

✗ shifted towards larger masses
✗ width increased

With pileup

width = 29.5 GeV
width = 21.0 GeV
Jet areas

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

Area ≡ 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 ≈ background particle
  - ⇒ active area ≈ uniform background
  - passive area ≈ pointlike background

- **Notes**:
  - passive = active for large multiplicities
  - require an IR-safe algorithm!
**Jet area: examples**

**Example:** active area for a simple event

$k_t$  

anti-$k_t$

one ghost at every grid cell
**Example**: perturbative expansion of areas (at order $\alpha_s$)

$$\langle A(p_t, R) \rangle = A_0 + \frac{C_F A}{b_0 \pi} \pi R^2 d \log \left( \frac{\alpha_s(Q_0)}{\alpha_s(Rp_t)} \right)$$

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

<table>
<thead>
<tr>
<th>$A_0/\pi R^2$</th>
<th>$d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>passive</td>
<td>active</td>
</tr>
<tr>
<td>$k_t$</td>
<td>1</td>
</tr>
<tr>
<td>Cam/Aachen</td>
<td>1</td>
</tr>
<tr>
<td>anti-$k_t$</td>
<td>1</td>
</tr>
<tr>
<td>SISCone</td>
<td>1</td>
</tr>
</tbody>
</table>

- $Q_0 \equiv$ IR regulator $\propto$ background density
**Example**: perturbative expansion of areas (at order $\alpha_s$)

\[
\langle A(p_t, R) \rangle = A_0 + \frac{C_{F,A}}{b_0\pi} \pi R^2 d \log \left( \frac{\alpha_s(Q_0)}{\alpha_s(Rp_t)} \right)
\]

- area $\neq \pi R^2$, area $\neq$ const.
- in agreement with Monte-Carlo simulations
Pileup subtraction

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

\[ \hat{p}_t, \text{subtracted} = p_t, \text{jet} - \rho_{\text{pileup}} \times A_{\text{jet}} \]

- **Jet area:** [M.Cacciari, G.Salam, G.S., 08]
  - region where the jet catches infinitely soft particles (active/passive)
  - tractable analytically in pQCD

- **Pileup density per unit area:** \( \rho_{\text{pileup}} \)
  - e.g. estimated from the median of \( p_t, \text{jet} / A_{\text{jet}} \)

---

- **Median**
- **Eta**

---

- **Pt, jet / Areajet**
- **Eta**
Pileup subtraction

**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)
  - tractable analytically in pQCD

- **Pileup density per unit area:** \( \rho_{\text{pileup}} \)
  - e.g. estimated from the median of \( p_{t, \text{jet}} / \text{Area}_{\text{jet}} \)
**Pileup subtraction**

**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)
  - tractable analytically in pQCD

- **Pileup density per unit area:** \( \rho_{\text{pileup}} \)
  - e.g. estimated from the median of \( p_{t,\text{jet}} / \text{Area}_{\text{jet}} \)

implemented in FastJet on an event-by-event basis
Our example: subtracting pileup

Pileup unsubtracted

Pileup subtracted

width = 29.5 GeV
width = 21.0 GeV

✓ position reasonable
✓ dispersion reduced
$AA$: more background!

$pp$

$AA$
Complications

Problem:
- Much larger background (∼100 GeV/unit area at RHIC, ∼250 GeV/unit area at the LHC)
- With large fluctuations

Question: how well can we measure the “hard jet” (quenched or not) in the heavy-ion background?
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 $\Delta p_t$ shift, smallest $\Delta p_t$ dispersion

Note: in what follows, $R$ fixed to 0.4
Framework for study

Hard event (quenched or unquenched) → cluster subtract → Hard jets

Δ$pt$

average dispersion

- Generic trends under control
- Final numbers may change

[Generic trends under control, final numbers may change]

Analysis: FastJet(v2.4)

Ideally: smallest $Δpt$ shift, smallest $Δpt$ dispersion

Note: in what follows, $R$ fixed to 0.4

[M.Cacciari, J.Rojo, G.Salam, GS, in prep.]
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 $> \text{SISCones} \gg$ anti-$k_t$
Idea #1: use a local range to compute $\rho_{bkg}$

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

- Exclude the hardest jets from the determination of $\rho_{bkg}$
  $\Rightarrow$ reduce the bias in the computation median
Effect of choosing a local range

Effect $\sim 0.5$-1 GeV

Differences between local ranges $\rightarrow$ uncertainty

For limited acceptance, global range $\approx$ local range

Analytic control would be nice
Idea #2: use filtering

cluster with Cambridge/Aachen(R)
Idea #2: use filtering

- p. 27
Idea #2: use filtering

- cluster with Cambridge/Aachen(R)
- for each jet
- recluster with Cambridge/Aachen(R/2)
Idea #2: use filtering

- cluster with Cambridge/Aachen(R)

  for each jet

  recluster with Cambridge/Aachen(R/2)

  keep the 2 hardest subjets
Idea #2: use 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 $H \rightarrow b\bar{b}$ tagging
[J.Butterworth, A.Davison, M.Rubin, G.Salam, 08]

Proven useful for kinematic reconstructions
[M.Cacciari, J.Rojo, G.Salam, GS, 08]
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
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
Conclusions

**Cone algorithm:** use and infrared-and-collinear-safe one

- ATLAS Cone
- CDF/D0 MidPoint

\[ \{ \text{ATLAS Cone, CDF/D0 MidPoint} \} \rightarrow \text{SIS Cone} \]

\[ \text{SIS Cone} \quad \checkmark \text{fast} \quad \checkmark \text{safe} \]

- CMS Lt. Cone

\[ \rightarrow \text{anti-}k_t \]

\[ \text{anti-}k_t \quad \checkmark \text{fast} \quad \checkmark \text{safe} \]

**Background subtraction:** use jet areas

- properly defined, under analytic control
- simple and generic subtraction method

**More refined techniques:** use local ranges and filtering techniques

- decrease sensitivity to the background and its fluctuations
- RHIC and LHC may behave differently
- **Try BOTH** anti-\(k_t\) (reliable because of its rigidity)
  AND Cambridge/Aachen+filtering (many nice features)