The DØ Run II Cone jet algorithm: problems and suggested changes

B.Andrieu (LPNHE, Paris)

Outline:

- Introduction
- Algorithm description
- Present and future issues
- Changes for low p_T
- Changes for high lumi
- Summary and outlook

Introduction: cone jet algorithm basics

What is a jet algorithm used for? **How is it defined?**

- Associate "close" to each other "particles"
  → **Clustering procedure**
  - "particles"
    - partons (analytical calculations or parton showers MC)
    - "hadrons" = final state particles (MC particles or charged particles in trackers)
    - towers (or cells or preclusters or any localized energy deposit)
  - "close" ? → **Distance**
    - $\Delta R = \sqrt{\Delta\eta^2+\Delta\phi^2}$ (RunI) or $\sqrt{\Delta Y^2+\Delta\phi^2}$ (RunII) for Cone Algorithm
- Calculate jet 4 - momentum from "particles" 4 - momenta
  → **Recombination scheme**
  - invariant under longitudinal boosts
    - Snowmass scheme (RunI): E_T -weighted recombination scheme in (η,ϕ)
    - covariant or E - scheme (RunII): 4 - momenta addition
  - used at the end of clustering but also during clustering process
    (not necessarily the same, still preferable)

Cone (proto-) jets are defined as "stable" cones, i.e. cones whose geometrical position (axis) coincides with jet direction (3-momentum)

→ How to find all stable cones in minimum CPU time?

DØ Run II Cone algorithm description (I)

Algorithm defined in "RunII Jet Physics" (hep-ex/0005012)
parameters in green, changes w.r.t. definition in red

- Reduce the number of seeds for clustering → **Preclustering**
  - seeds = towers with p_T > **0.5 GeV** ordered in decreasing p_T
  - cluster (and remove) all towers around seed in a R= 0.3 cone
- Search for stable cones starting from preclusters → **Clustering**
  - seeds = p_T ordered list of preclusters with p_T > **1 GeV**
    except those close to already found proto-jets: **ΔR (precluster, proto-jet)< 0.5 R_{cone}**
  - form **proto-jet**
    - cluster all towers within R_{cone} of the precluster axis in (Y ,ϕ) space
    - recalculate proto-jet 4-momentum (E - scheme)
  - iterate proto-jet formation (cone drifting) until
    - cone is stable (cone axis coincides with proto-jet direction) **OR**
    - **p_T < 0.5 Jet p_T^{min}** **OR**
    - # iterations = 50 (to avoid ∞ cycles)
  - remove duplicates
- Ensure infrared and collinear safety → **Clustering from midpoints**
  - repeat same clustering using midpoints (between previously found proto-jets) as seeds except
    - no condition on close proto-jet
    - no removal of duplicates

→ proto-jets (with possible overlaps)

DØ Run II Cone algorithm description (II)

- Treat overlapping proto-jets → **Merging/Splitting**
  - use p_T ordered list of proto-jets (from seeds and midpoints)
    - **a p_T cut at 8 GeV was used in RunI and suggested for RunII, but is not applied now**
    - note that p_T^{min} / 2 cut on proto-jets candidates implies a p_T cut at 4 GeV
  - attribute shared energy exclusively according to shared energy fraction
    - $E_{T,1\cap 2} > f \cdot \mathrm{Min}(E_{T,1}, E_{T,2})$ → Merge jets
    - $E_{T,1\cap 2} < f \cdot \mathrm{Min}(E_{T,1}, E_{T,2})$ → Split jets = assign each particle to its closest jet

    f = 50 %
  - at each merging/splitting step
    - recalculate 4-momenta of merged/splitted jets
    - re-order list of merged/splitted jets
  - treat all proto-jets until no shared energy is left

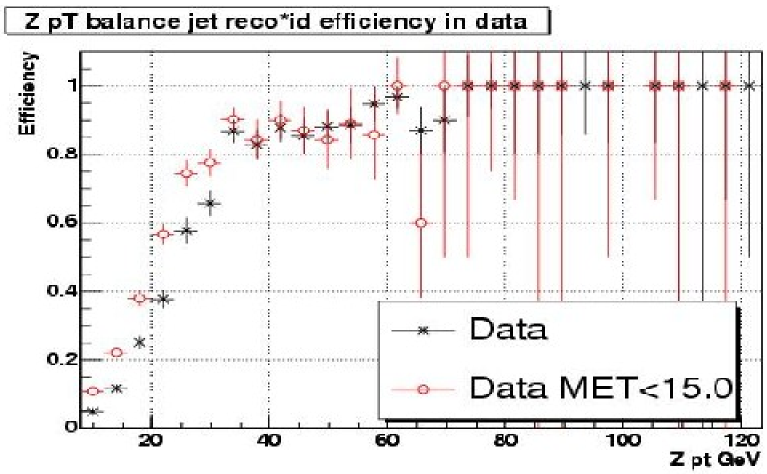
→ final list of proto-jets (without overlap)

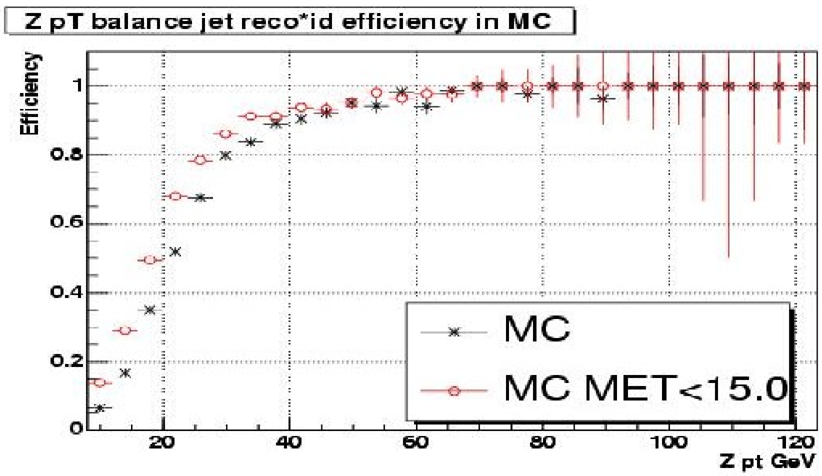
- Calculation of jet variables and final p_T cut
  - E-scheme recombination (4-momenta addition)
  - p_T ordering
  - final p_T cut: $p_T^{jet} > p_T^{min}$ (p_T^{min} = 8 GeV in p14, 6 GeV in p17)

→ final list of jets

Present and future problems

- **Efficiency at low p_T**





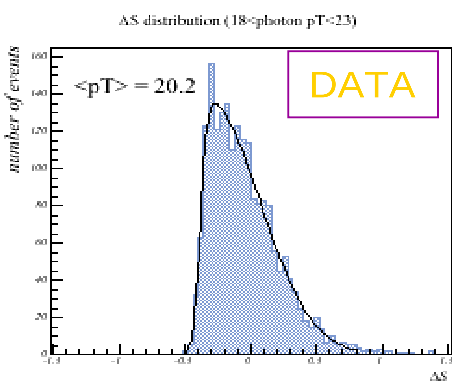
Heinmiller, CALGO, 5/31/05

- **Behaviour at high lumi**
  - the average number of jets increases with vertex multiplicity (Schwartzman, CALGO, 10/5/04)
  - the superposition of many min. bias events could increase dramatically the number of fake jets
  - the effect is probably limited until now but will become very important at high lumi (already seen in Run I)

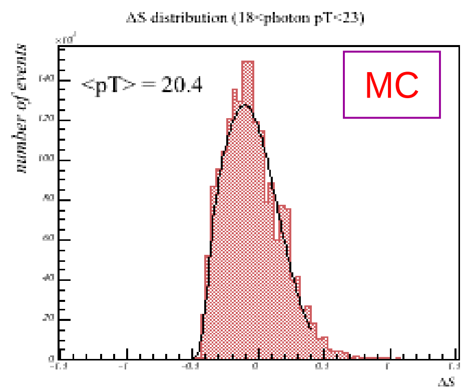
Changes to improve low p_T efficiency (I)

Possible `rcp` changes listed by order of importance

- **Reduce p_T^{min} cut (final cut on p_T^{jet} at the end of algorithm)**
  - reco efficiency is driven by deposited energy...
  - ... but for physics analyses (e.g. cross section measurement), we are usually more interested by efficiency as a function of "true" (particle level) energy
  - given JES corrections + possible residual miscalibration + out-of-cone showering + resolution effects, the effect of p_T^{min} = 8 GeV cut is visible on jets of "true" p_T up to 30 GeV in data



18<pT<23



Makovec & Grivaz
CALGO, 6/21/05
D0-note 4807

$\Delta S = (p_T^{jet} - p_T^{\gamma})/p_T^{\gamma}$ for fixed p_T^{γ} range, fitted by Gaussian x turn- on

  - effect known as low p_T bias in JES group, shown to be removed completely by reducing p_T^{min} down to 3 GeV (D. Gillberg et al. D∅-note 4571)

✓ **probably the main effect to correct for in order to improve efficiency at low p_T**
✓ **impact of reducing p_T^{min} must be studied in great detail (CPU, jet multiplicity,...)**
✓ **if it proved difficult to reduce in general case, reduce it for selected samples only?**

Changes to improve low p_T efficiency (II)

- **Remove p_T^{min} / 2 cut on proto-jets candidates**
  - this cut has no known justification and makes algorithm non "Run-II standard"
  - first study by E. Busato (CALGO, 26 Oct. 2004) shows ~ 6% gain in efficiency for p_T ~ 15 GeV...
  - ...but at a ~ 10% in CPU time increase cost
  - more detailed and systematic studies for different samples (e.g. multi-jet) and lumi conditions needed
  - Important note:
    due to this cut, a change in final p_T^{min} cut changes the result of the algorithm unpredictably:
    using the algorithm with p_T^{min} = 6 GeV then applying a cut on p_T^{jet} > 8 GeV will NOT yield the same jets as running the algorithm with p_T^{min} = 8 GeV
    → good reason of principle to remove this cut anyway
- **Modify preclustering parameters?**
  - p_T^{min} cut (0.5 GeV) on tower seeds for preclustering
    - It was 1 GeV in Run I, so it should be fine
    - Small effect expected but needs to be checked
  - p_T^{min} cut (1 GeV) on precluster seeds for clustering
    - Again probably a small effect, but needs to be checked
- **Remove cut on distance between seeds and found proto-jets** **(ΔR > 0.5 R_{cone})** ?
  - this cut has no known justification and makes algorithm non "Run-II standard"
  - its effect is probably (hopefully!) harmless but needs to be studied
  - it would be best to remove it anyway, unless a good argument is found to keep it,
    e.g. if it really saves much CPU time and has a totally negligible effect on the result.

→ Changes studied by Marine Michaut

Changes for high lumi (I)

Two possible changes in the code (merging/splitting)

- **New p_T cut on proto-jets before merging/splitting**
  - **a cut at 8 GeV was used in RunI and strongly suggested for RunII**
  - this will replace/improve the $p_T^{min}/2$ cut on proto-jets candidates
  - its aim is to avoid forming a fake jet by merging many low p_T proto-jets from min. bias events (helpful/needed at high lumi)
  - a simple cut at a fixed p_T value is maybe not optimal, since it might be wanted to always merge a low p_T proto-jet with a very high p_T one

→ before merging two protojets (1 & 2, with $p_T^1 > p_T^2$) , apply a cut on p_T^2, but only if p_T^1 is below some threshold:
if $p_T^1 > p_T^{min\_leading}$ || $p_T^2 > p_T^{min\_second}$ → do usual merging
otherwise → remove shared cells from 2

- **New cut on the maximum number of merges**
  - since the maximum number of merges is known for good jets at low lumi, it seems justified to limit the number of merges in prevision of high lumi
  - however this maximum number of merges might be not justified for very high p_T proto-jets

→ before merging two protojets (1 & 2, with $p_T^1 > p_T^2$) , apply a cut on the number of merges but only if p_T^1 is below some threshold:
if $p_T^1 > p_T^{min\_merge}$ || $N_{merge} < N_{merge\_max}$ → do usual merging
otherwise → remove shared cells from 2

Changes for high lumi (II)

- The two changes proposed in previous slide have been implemented (in private version) by E. Busato
- 4 new **`rcp`** parameters introduced:
  - **`pT_min_leading_protojet`**, **`pT_min_second_protojet`** for first cut
  - **`merge_max`**, **`pT_min_nomerge`** for second cut
- Changes can be reverted easily using **`rcp`** parameters, so as to give exactly the same result as previous algorithm for comparisons
- Detailed and sytematic studies for different samples and lumi conditions needed to define appropriate values for new **`rcp`** parameters

→ Changes studied by Marc-André Pleier

Summary and outlook

- **Two main problems in D∅ Run II Cone algorithm**
  - reco efficiency at low p_T
  - behaviour at high lumi
- **Two types of solutions identified, one for each problem**
  - **`rcp`** changes (lower the thresholds and remove unwanted cuts) to improve reco efficiency at low p_T (and make the algorithm more conform to RunII)
  - code change (two new cuts) in merging/splitting step to avoid excessive merging of low p_T proto-jets in pile-up events at high lumi

  → **Note that the compatibility of these changes needs to be proven: lowering thresholds to improve reco efficiency at low p_T might give birth to new problems in pile-up events at high lumi**
- **Strategy**
  - optimise changes separately for each problem
  - combine changes and study the effect for each problem
  - if necessary, do a secound round of combined optimisation taking into account both problems at the same time

All comments and suggestions welcome!