Jet Algorithms at D-Zero

Run 1 jet algorithms at D-Zero:

- Cone Jets
  - Starting points, Split/merge criteria, cone sizes
  - ‘D-Zero’ angle definition
  - Snowmass angle definition
  - Modified Snowmass, R_{sep}
- NN or Nearest Neighbor
- K_T Jets
  - preclustering choices

Cone Jets at DØ

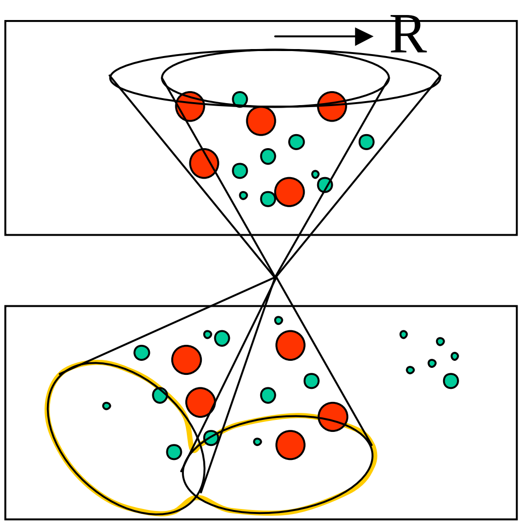
Cone jets are defined by a number of algorithm parameters

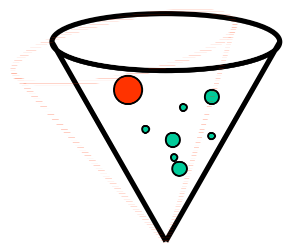
- Cone Size (ie radius, R = 0.3,0.5,0.7 in η x ϕ space)
- Seed or starting point for iterations (DØ uses 1 GeV E_T towers)

Calorimeter E_T

Jet Seeds

- Minimum ET requirement = 8 GeV





•Clustering begins w/ seed tower > 1 GeV

•Preclusters are formed by combining seed towers w/ their neighbors (reduces # of jet computations)

•Draw cone around seed/precluster, find ET weighted centroid, recalculate jet centroid, repeat until stable

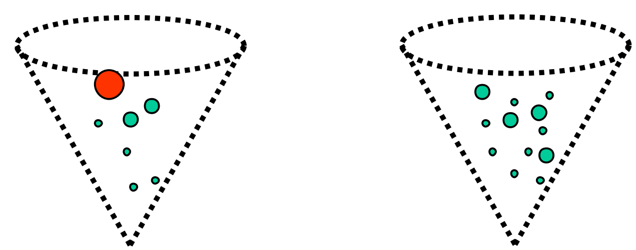
Standard Snowmass definitions

$$\eta_{jet} = \frac{\sum_i E_T^i \eta^i}{\sum_i E_T^i}$$

$$\phi_{jet} = \frac{\sum_i E_T^i \phi^i}{\sum_i E_T^i}$$

$$E_T = \sum_i E_T^i = \sum_i E_i \sin(\theta_i)$$

Lost jets



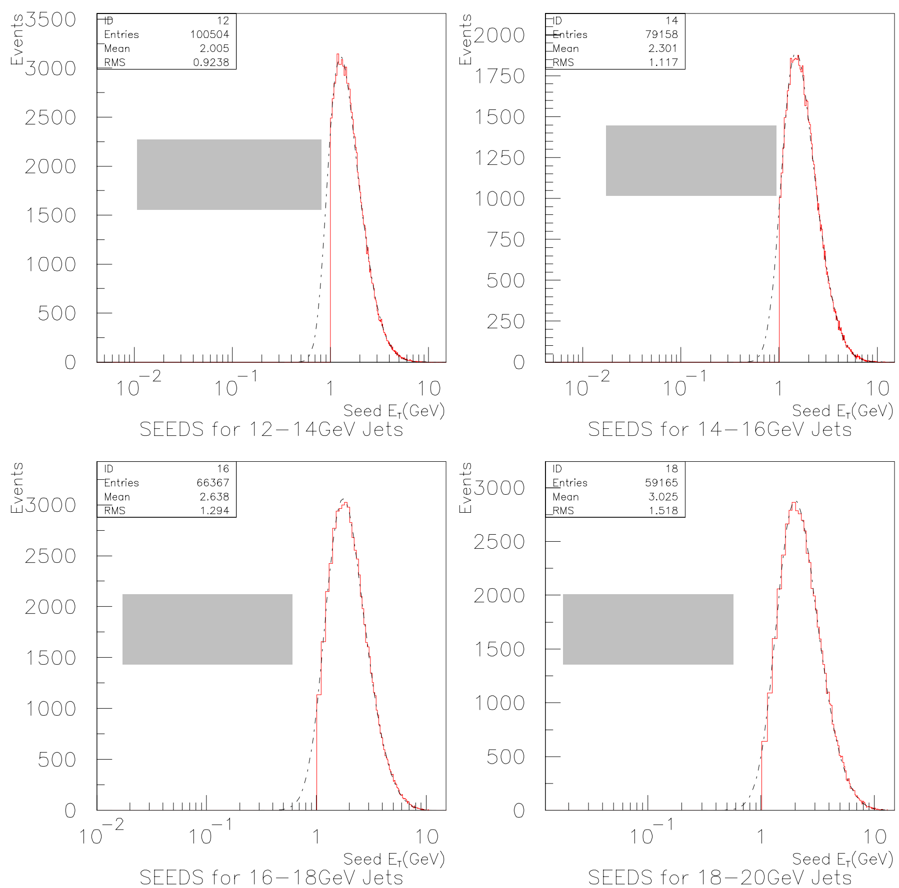
Seed, but
sum(ET) < 8
no jet found

Sum(ET) >8,
but no seed
no jet found

Seeds in Data

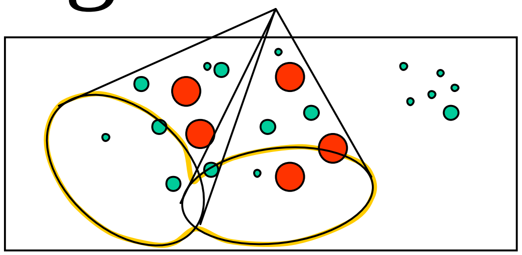
Seed distribution

Extrapolation



Splitting or Merging

Fixed cones often lead to ambiguous jet definitions:
• Which energy clusters go in the jet?
• Do I have one or two jets?
•Need split/merge prescription….



At the calorimeter level jets spread out and are more likely to overlap due to showering effects. Since the ability to resolve individual jets is not perfect, some compromise is made to merge jets that are close together.

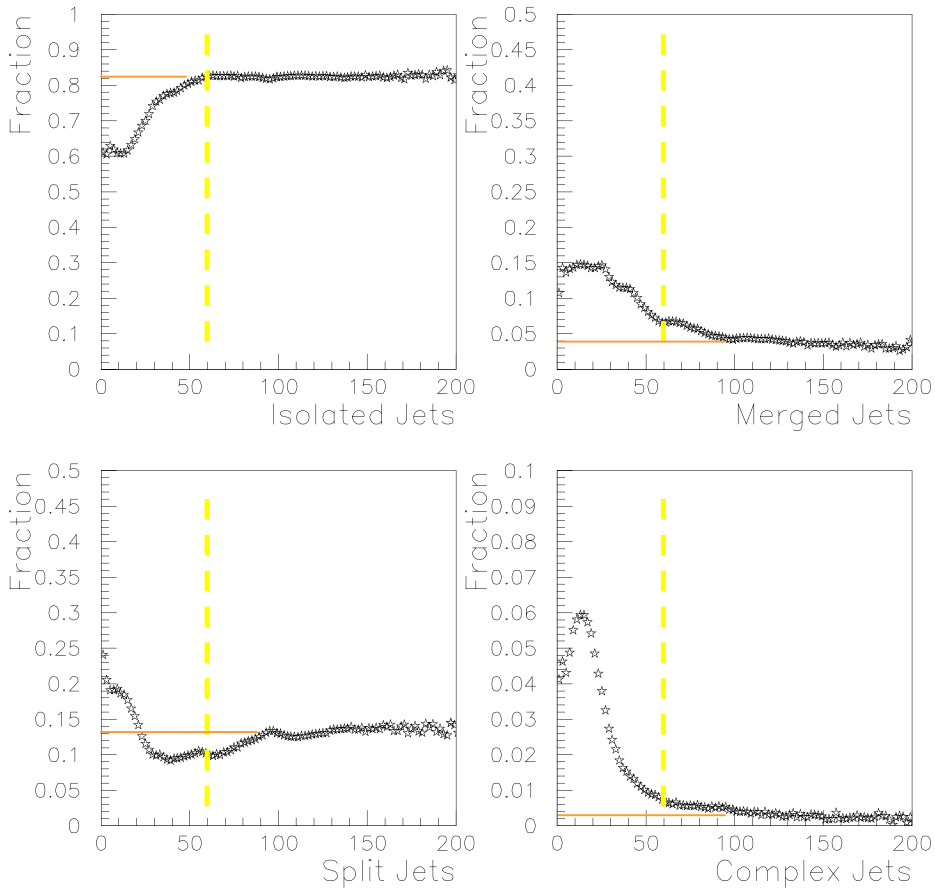
DØ choices for overlapping cone jets:

• If 50% of the ET of the non-leading jet is shared, jets are merged

• otherwise they are split - each shared cell is assigned to the nearest jet

• 'Complicated' jets also can result from multiple splitting and merging

Splitting/Merging in DØ Data



Splitting/Merging of jets varies strongly below ~60GeV

Constant values approached at large E_T's

Angle Definitions

Snowmass:

$$\eta_{jet} = \frac{\sum_i E_T^i \eta^i}{\sum_i E_T^i}$$

$$\phi_{jet} = \frac{\sum_i E_T^i \phi^i}{\sum_i E_T^i}$$

•Much better agreement with 4-vector parton addition for clustering at NLO

D-Zero original choice:

$$\theta_{jet} = \tan^{-1}\left(\frac{\sqrt{(\sum_i E_x^i)^2 + (\sum_i E_y^i)^2}}{\sum_i E_z^i}\right)$$

$$\phi_{jet} = \tan^{-1}\left(\frac{\sum_i E_y^i}{\sum_i E_x^i}\right)$$

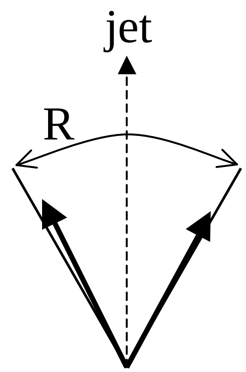
$$\eta_{jet} = -\ln(\tan(\theta_{jet}/2))$$

•Shows slightly less η-bias in reconstructed jets than Snowmass

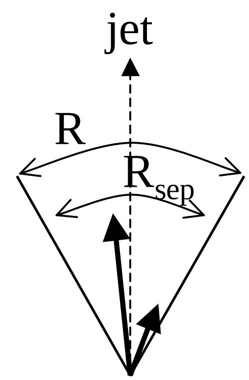
At calorimeter level there is no significant difference between the two, except for a small, Δη ~ 0.05 shift for far forward jets

Modified Snowmass, R_{sep}

Snowmass jets at NLO



Modified Snowmass at NLO



Phenomenological parameter introduced to accommodate difference in NLO cone algorithm and calorimeter algorithm

Rsep = 1.3 R determined from jet merging studies in data, in essence jets within a distance of 1.3 R are merged, farther jets are split.

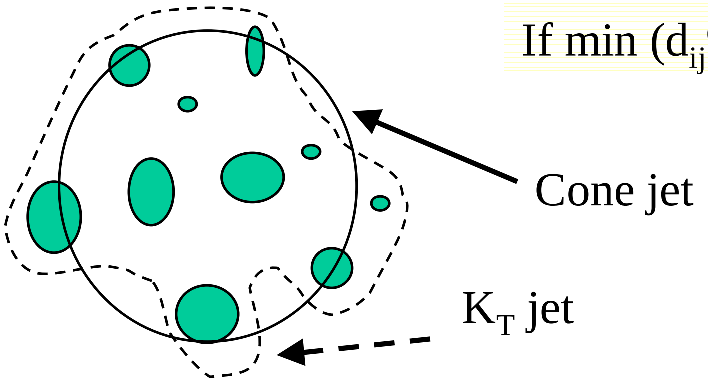
KT Algorithm

Define jets, not by geometric cones, but by more "organic" standards:

Cal. towers clustered into singles: $d_i^{(n)} = E_{Ti}^2$

pairs: $d_{ij}^{(n)} = \min (E_{Ti}^2, E_{Tj}^2) \times \Delta R_{ij}^2 / D$

If $\min (d_{ij}^{(n)} , d_i^{(n)}) = d_{ij}^{(n)}$ merge clusters, else keep separate



Repeat iterations until only separate objects remain

$E_T < f \times E_T^{max}$ - 'f'cut to drop low E_T/beam jets

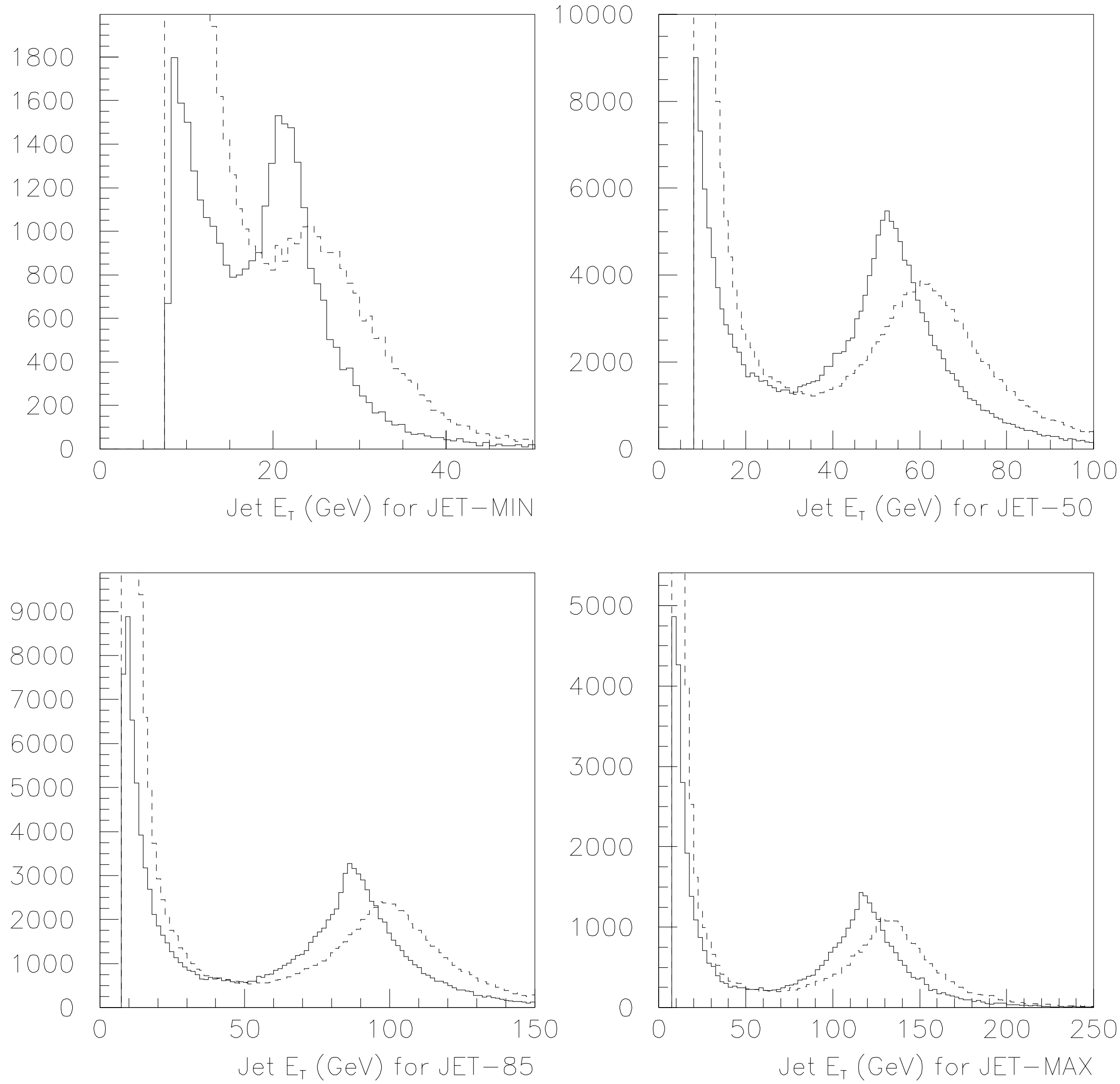
Conceptually:

- All clusters w/in radius, D, are merged (like cone algorithm)
- Clusters >>D can be merged if ΔE_T>>0

Shapes are more natural

- no arbitrary spl/mer param @ calorim.
- no Rsep param @ parton level

same algorithm @ all levels



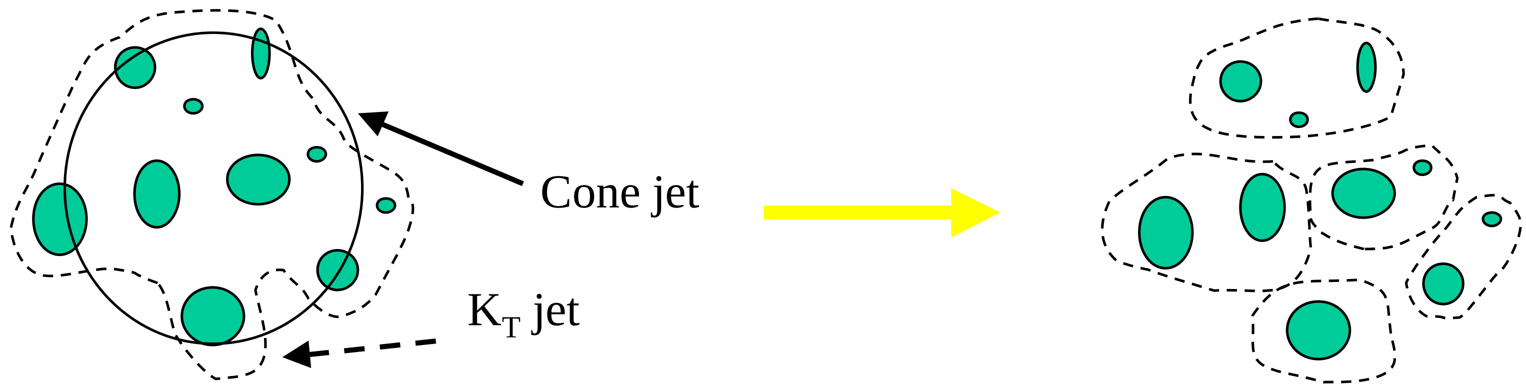
Cone jets (solid)

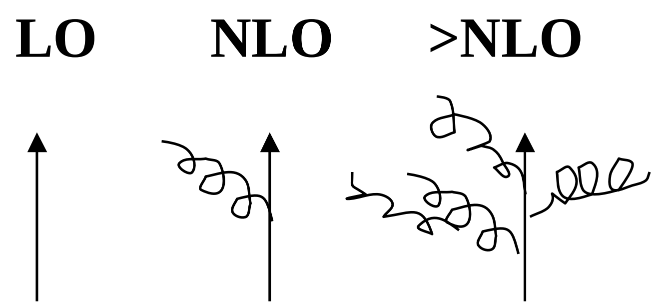
K_T jets (dashed)

‘seedless’ K_T algorithm finds many more low E_T jets

scale difference from 0.7 cone jets even at large E_T

Adjusting merge criteria gives powerful tool for studying jet structure.





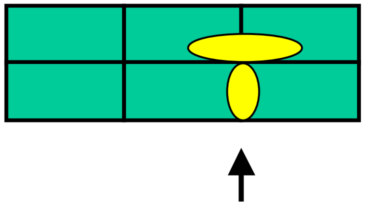
K_T Algorithm Parameters

Clearly D,f parameters control which jets are found but, what other choices affect the behavior of K_T algorithm?

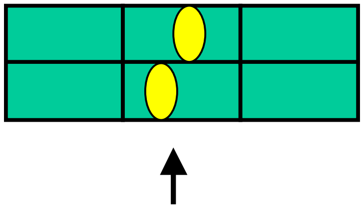
Preclustering - detector granularity limits resolution for particles, preclustering at BOTH detector and particle level necessary for physics comparisons

•Effective lower limit on useable `D' imposed by calorimeter

•Preclusters speed final jet finding, save much CPU when many `D's are studied



One particle hits multiple cells/towers



Two particles hit one cell/tower

Toward Common Algorithms

Common algorithms between experiments are desirable to avoid confusion, every subtle change in a jet algorithm produces subtle changes in the physics measurement, making comparison of experiments more difficult

Points to consider for Cone Algorithms:

- cone size
- values of seed thresholds, or `seedless' algorithm
- Snowmass angle definition for clustering iterations and final angles!
- Merge criteria, 60%, 50%, 40%? Do detector designs affect this choice?
- Split prescription - how is E_T in overlap region shared?
- When to apply merge? Before or after min ET cut? And what is this cut?
-
-

Points to consider for KT Algorithms:

- Preclustering - many ways to implement
  - run KT algorithm w/ minimal `D' for detector
  - apply a basic NN algorithm
    - where to start such clusters, highest ET tower?
  - how to deal w/ negative energies?
- `Clustering scheme', Snowmass or 4-vectors?
-

Consistent algorithm choices are crucial BEFORE data begins

Large data samples prevent the easy reapplication of jet clustering someplace `down the road'. Also new jet definitions require completely new derivations or energy scale corrections!

Remember a jet is what your algorithm finds - there's a lot of room for creativity here, but in this case let's at least use the same crayons!