Run II Jet Physics

Gerald C. Blazey[a], Jay R. Dittmann[b], Stephen D. Ellis[c], V. Daniel Elvira[b], K. Frame[d], S. Grinstein[e], Robert Hirosky[f], R. Piegaia[e], H. Schellman[g], R. Snihur[g], V. Sorin[e], Dieter Zeppenfeld[h]

[a]Department of Physics, Northern Illinois University, DeKalb, IL 60115, USA

[b]Fermilab, P.O. Box 500, Batavia, IL 60510, USA

[c]Department of Physics, University of Washington, Box 351560, Seattle, WA 98195-1560, USA

[d]Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48824, USA

[e]Depto. de Fisica, FCEyN-Universidad de Buenos Aires, Pab I, Ciudad Universitaria, (1428) Capital Federal, Argentina

[f]Department of Physics, University of Illinois at Chicago, Chicago, IL 60607, USA

[g]Physics Department, Northwestern University, Evanston, IL 60210

[h]Department of Physics, University of Wisconsin at Madison, Madison, WI 53706, USA

The Run II jet physics group includes the Jet Algorithms, Jet Shape/Energy Flow, and Jet Measurements/Correlations subgroups. The main goal of the jet algorithm subgroup was to explore and define standard Run II jet finding procedures for CDF and DØ. The focus of the jet shape/energy flow group was the study of jets as objects and the energy flows around these objects. The jet measurements/correlations subgroup discussed measurements at different beam energies; α_S measurements; and LO, NLO, NNLO, and threshold jet calculations. As a practical matter the algorithm and shape/energy flow groups merged to concentrate on the development of Run II jet algorithms that are both free of theoretical and experimental difficulties and able to reproduce Run I measurements.

Starting from a review of the experience gained during Run I, the group considered a variety of cone algorithms and K_T algorithms. The current understanding of both types of algorithms, including calibration issues, are discussed in this report along with some preliminary experimental results. The jet algorithms group recommends that CDF and DØ employ the *same* version of *both* a cone algorithm and a K_T algorithm during Run II. Proposed versions of each type of algorithm are discussed. The group also recommends the use of full 4-vector kinematic variables whenever possible. The recommended algorithms attempt to minimize the impact of seeds in the case of the cone algorithm and preclustering in the case of the K_T algorithm. Issues regarding precluster definitions and merge/split criteria require further study.

1. Prologue

The Run I jet programs at CDF and DØ made impressive measurements of the inclusive jet cross section, dijet angular and mass distributions, and triple differential cross sections. These measurements were all marked by statistical accuracy equal or superior to current theoretical accuracy [1]. However, the always compelling search for quark compositeness, the quest to improve the calculational accuracy of QCD, and the desire to fully understand the composition of the proton will certainly prompt improvements over these measurements. Without question, with $\sim 2\ fb^{-1}$, the Run II jet physics program will extend the jet measurements of Run I to even higher jet energies.

There are three issues, experimental and theoretical, that currently limit the sensitivity of compositeness searches and QCD tests: limited knowledge of the parton distribution functions (pdfs), systematic uncertainties related to jet energy calibration, and the limited accuracy of fixed order perturbative calculations due to the incomplete nature of the calculations and incomplete specification of jet finding algorithms. Inadequate knowledge of the pdfs and calibration are currently the dominant uncertainties, engendering greater than 50% uncertainties at the largest energies. The reader may refer to the chapter on Parton Distributions for a complete discussion of pdf measurements.

As mentioned, the uncertainty of NLO perturbative calculations is due in part to the inherent incompleteness of fixed order calculations. The initial meeting of the jet physics group included talks on “Leading Order (LO) Multi-jet Calculations” by Michelangelo Mangano, “Next-to-Leading Order (NLO) Multi-jet Calculations” by Bill Kilgore, “Prospects for Next-to-NLO (NNLO) Multi-jet Calculations” by Lance Dixon, “Threshold Resummations for Jet Production” by Nicolas Kidonakis, "Different Beam Energies" by Greg Snow, and "α_S Measurements in Jet Systems" by Christina Mesropian. These attempts to improve the accuracy of perturbative calculations show the vigorous nature of ongoing efforts and should prove fruitful before the arrival of Run II data.

Jet algorithms, the other source of calculation uncertainty, start from a list of "particles" that we take to be calorimeter towers or hadrons at the experimental level, and partons in a perturbative QCD calculation. The role of the algorithm is to associate clusters of these particles into jets such that the kinematic properties of the jets (*e.g.*, momenta) can be related to the corresponding properties of the energetic partons produced in the hard scattering process. Thus the jet algorithm allows us to "see" the partons (or at least their fingerprints) in the hadronic final state.

Differences in the properties of reconstructed jets when going from the parton to the hadron or calorimeter level are a major concern for a good jet algorithm. Each particle i carries a 4-momentum p_i^μ, which we take to be massless. The algorithm selects a set of particles, which are typically emitted close to each other in angle, and combines their momenta to form the momentum of a jet. The selection process is called the "jet algorithm" and the momentum addition rule is called the "recombination scheme". Note that these two steps are logically distinct. One can, for example, use one set of kinematic variables in the jet algorithm to determine the particles in a jet and then construct a separate set of kinematic variables to characterize the jets that have been identified. This point will be important in subsequent discussions.

Historically cone algorithms have been the jet algorithm of choice for hadron-hadron experiments. As envisioned in the Snowmass algorithm [2], a cone jet of radius R consists of all of the particles whose trajectories (assuming no bending by the magnetic field of the detector) lie in an area $A = \pi R^2$ of $\eta \times \phi$ space, where η is the pseudorapidity $\eta = -\ln \tan \theta/2$. It is further required, as explained in detail below, that the axis of the cone coincides with the jet direction as defined by the E_T-weighted centroid of the particles within the cone (where E_T is transverse energy, $E_T = E \sin \theta$). In principle, one simply searches for all such "stable" cones to define the jet content of a given event.

In practice, in order to save computing time, the iterative process of searching for the "stable" cones in experimental data starts with only those cones centered about the most energetic particles in the event (the so-called "seeds"). Usually, the seeds are required to pass a threshold energy of a few hundred MeV in order to minimize computing time. The E_T-weighted centroids are calculated for the particles in each seed cone and then the centroids are used as centers for new cones in $\eta \times \phi$ space. This procedure is iterated for each cone until the cone axis coincides with the centroid. Unfortunately, nothing prevents the final stable cones from overlapping. A single particle may belong to two or more cones. As a result, a procedure must be included in the cone algorithm to specify how to split or merge overlapping cones [3].

At least part of the uncertainty associated with fixed order perturbative calculations of jet cross sections can be attributed to the difficulties encountered when this experimental jet cone algorithm, with both seeds and merging/splitting rules, is applied to theoretical calculations. (See Ref. [1] for a discussion of the CDF and DØ algorithms.) Neither issue was treated by the original Snowmass algorithm [2] that forms the basis of fixed order perturbative cone jet calculations. Current NLO inclusive jet cross section calculations (which describe either two or three final state partons) require the addition of an ad hoc parameter R_{Sep} [4]. This additional parameter is used to regulate the clustering of partons and simulate the role of seeds and merging in the experimentally applied algorithm. In essence, the jet cone algorithm, used so pervasively at hadron-hadron colliders, must be modeled in NLO calculations. This modeling results in 2–5% uncertainties as a function of jet transverse energy E_T in calculated cross sections.

Even worse, with the current cone algorithms, cross sections calculated at NNLO exhibit a marked sensitivity to soft radiation. As an illustration, consider two well-separated partons that will just fit inside, but at opposite sides, of a single cone. With only the two partons, and nothing in between to serve as a seed, the current standard cone algorithms will reconstruct the two partons as two jets. At NNLO a very soft gluon could be radiated between the two well-separated partons and serve as a seed. In this case the single jet solution, with both partons inside, will be identified by the current cone algorithm. Thus the outcome of the current cone algorithm with seeds is manifestly sensitive to soft radiation. Because of the difficulties inherent with typical usage of the cone algorithm, the jet algorithm and jet shape/energy flow subgroups decided to establish an Improved Legacy Cone Algorithm (whimsically dubbed ILCA). Ideally, the ILCA should replicate Run I cross sections within a few percent, but not have the same theoretical difficulties.

Inspired by QCD, a second class of jet algorithms, K_T algorithms, has been developed. These algorithms successively merge pairs of "particles" in order of increasing relative transverse momentum. They typically contain a parameter, D (also called R), that controls termination of merging and characterizes the approximate size of the resulting jets. Since a K_T algorithm fundamentally merges nearby particles, there is a close correspondence of jets reconstructed in a calorimeter to jets reconstructed from individual hadrons, leptons and photons. Furthermore, every particle in an event is assigned to a unique jet. Most importantly, K_T jet algorithms are, by design, infrared and collinear safe to all orders of calculation. The algorithms can be applied in a straightforward way to fixed–order or resummed calculations in QCD, partons or particles from a Monte Carlo event generator, or energy deposited in a detector [5].

However, until recently, a full program for the calibration of K_T algorithms at hadron-hadron colliders had not been developed. This was due mostly to difficulties with the subtraction of energy from spectator fragments and from the pile-up of multiple hadron-hadron interactions. Since the K_T jets have no fixed shape, prescriptions for dealing with the extra energy have been difficult to devise and the use of K_T algorithms at hadron-hadron colliders has been limited. Also, as with the issue of seeds in the case of the cone algorithm, there is a practical question of minimizing the computing time required to apply the K_T algorithm. Typically this is treated in a preclustering step where the number of "particles" is significantly reduced before the K_T algorithm is applied. A successful K_T algorithm must ensure that any preclustering step does not introduce the sort of extra difficulty found with seeds.

Buoyed by the successful use of K_T algorithms at LEP and HERA, eager to benefit from their theoretical preciseness, and reassured by recent success with calibration, the jet physics group decided to specify a standard K_T algorithm for Run II.

2. Attributes of the Ideal Algorithm

Although it provided a good start, the Snowmass algorithm has proved to be incomplete. It does not address either the phenomena of merging and splitting or the role of the seed towers with the related soft gluon sensitivity. Also, jet energy and angle definitions have varied between experiments. To treat these issues, the group began discussions with the following four general criteria:

1. *Fully Specified*: The jet selection process, the jet kinematic variables and the various corrections (*e.g.*, the role of the underlying event) should be clearly and completely defined. If necessary, preclustering, merging, and splitting algorithms must be completely described.

2. *Theoretically Well Behaved*: The algorithm should be infrared and collinear safe with no ad hoc clustering parameters.

3. *Detector Independence*: There should be no dependence on cell type, numbers, or size.

4. *Order Independence*: The algorithms should behave equally at the parton, particle, and detector levels.

The first two criteria should be satisfied by every algorithm; however, the last two can probably never be exactly true, but should be approximately correct.

2.1. Theoretical Attributes of the Ideal Algorithm

The initial efforts of the algorithm working group were focused on extending and illuminating the list of desirable features of an "ideal" jet algorithm. From the "theoretical standpoint" the following features are desirable and, for the most part, necessary:

1. *Infrared safety:* The algorithm should not only be infrared safe, in the sense that any infrared singularities do not appear in the perturbative calculations, but should also find solutions that are insensitive to soft radiation in the event. As illustrated in Fig. 1, algorithms that look for jets only around towers that exhibit some minimum energy activity, called seed towers or just seeds, can be quite sensitive to soft radiation. The experimental cone algorithms employed in previous runs have such seeds.

2. *Collinear safety:* The algorithm should not only be collinear safe, in the sense that collinear singularities do not appear in the perturbative calculations, but should also find jets that are insensitive to any collinear radiation in the event.

   A) Seed-based algorithms will in general break collinear safety until the jets are of sufficiently large E_T that splitting of the seed energy between towers does not affect jet finding (See Fig. 2). This was found to be the case for jets above 20 GeV in the DØ data, where jets were found with 100% efficiency using a seed tower threshold of 1.0 GeV [6]. The collinear dependence introduced via the seed threshold is removed when the jets have sufficient E_T to be reconstructed with 100% efficiency.

   B) Another possible collinear problem can arise if the algorithm is sensitive to the E_T ordering of particles. An example would be an algorithm where a) seeds are treated in order of decreasing

E_T and b) a seed is removed from the seed list when it is within a jet found using a seed that is higher on the list. For such an algorithm consider the configuration illustrated in Fig. 3. The difference between the two situations is that the central (hardest) parton splits into two almost collinear partons. The separation between the two most distant partons is more than R but less than $2R$. Thus all of the partons can fall within a single cone of radius R around the central parton(s). However, if the partons are treated as seeds and analyzed with the candidate algorithm suggested above, different jets will be identified in the two situations. On the left, where the single central parton has the largest E_T, a single jet containing all three partons will be found. In the situation on the right, the splitting of the central parton leaves the right-most parton with the largest E_T. Hence this seed is looked at first and a jet may be found containing only the right-most and two central partons. The left-most parton is a jet by itself. In this case the jet number changes depending on the presence or absence of a collinear splitting. This signals an incomplete cancellation of the divergences in the real and virtual contributions to this configuration and renders the algorithm collinear unsafe. While the algorithm described here is admittedly an extreme case, it is not so different from some schemes used in Run I. Clearly this problem should be avoided by making the selection or ordering of seeds and jet cones independent of the E_T of individual particles.

3. *Invariance under boosts:* The algorithm should find the same solutions independent of boosts in the longitudinal direction. This is particularly important for $p\overline{p}$ collisions where the center-of-mass of the individual parton-parton collisions is typically boosted with respect to the $p\overline{p}$ center-of-mass. This point was emphasized in conversations with the Jet Definition Group Les Houches [7].[1]

4. *Boundary Stability:* It is desirable that the kinematic variables used to describe the jets exhibit kinematic boundaries that are insensitive to the details of the final state. For example, the scalar E_T variable, explained in more detail in the next section, has a boundary that is sensitive to the number of particles present and their relative angle (*i.e.*, the boundary is sensitive to the mass of the jet). The bound $E_T^{max} = \sqrt{s}/2$ applies only for collinear particles and massless jets. In the case of massive jets the boundary for E_T is larger than $\sqrt{s}/2$. Boundary stability is essential in order to perform soft gluon summations.

5. *Order Independence:* The algorithm should find the same jets at parton, particle, and detector level. This feature is clearly desirable from the standpoint of both theory and experiment.

6. *Straightforward Implementation:* The algorithm should be straightforward to implement in perturbative calculations.

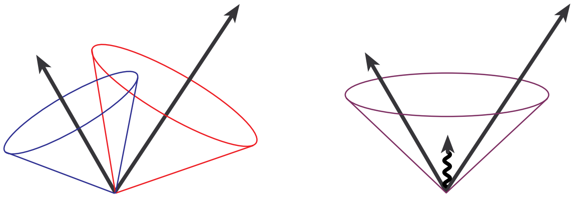

Figure 1. An illustration of infrared sensitivity in cone jet clustering. In this example, jet clustering begins around seed particles, shown here as arrows with length proportional to energy. We illustrate how the presence of soft radiation between two jets may cause a merging of the jets that would not occur in the absence of the soft radiation.

2.2. Experimental Attributes of the Ideal Algorithm

Once jets enter a detector, the effects of particle showering, detector response, noise, and energy from additional hard scatterings from the same beam crossing will subtly affect the performance of even the most ideal algorithm. It is the goal of the experimental groups to correct for such effects in each jet analysis. Ideally the algorithm employed should not cause the corrections to be excessively large. From an "experimental standpoint" we add the following criteria for a desirable jet algorithm:

1. *Detector independence:* The performance of the algorithm should be as independent as possible

[1] The Les Houches group discussed jet algorithms for both the Tevatron and LHC, and they sharpened their algorithm requirements by also requiring boundary stability (the kinematic boundary for the one jet inclusive jet cross section should be at the same place, $E_T = \sqrt{s}/2$, independent of the number of final state particles), suitability for soft gluon summations of the theory, and simplicity and elegance.

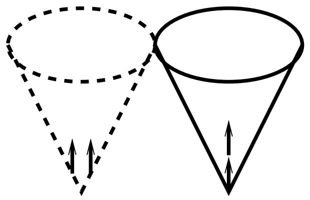

Figure 2. An illustration of collinear sensitivity in jet reconstruction. In this example, the configuration on the left fails to produce a seed because its energy is split among several detector towers. The configuration on the right produces a seed because its energy is more narrowly distributed.

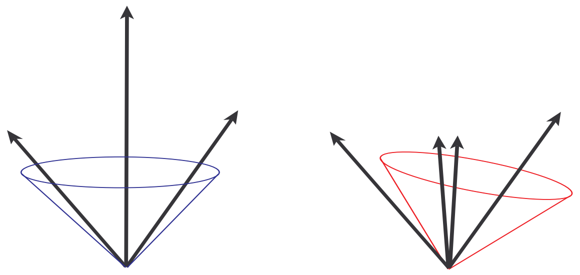

Figure 3. Another collinear problem. In this case we illustrate possible sensitivity to E_T ordering of the particles that act as seeds.

of the detector that provides the data. For example, the algorithm should not be strongly dependent on detector segmentation, energy response, or resolution.

2. *Minimization of resolution smearing and angle biases:* The algorithm should not amplify the inevitable effects of resolution smearing and angle biases.

3. *Stability with luminosity:* Jet finding should not be strongly affected by multiple hard scatterings at high beam luminosities. For example, jets should not grow to excessively large sizes due to additional interactions. Furthermore the jet angular and energy resolutions should not depend strongly on luminosity.

4. *Efficient use of computer resources:* The jet algorithm should provide jet identification with a minimum of computer time. However, changes in the algorithm intended to minimize the necessary computer resources, *e.g.*, the use of seeds and preclustering, can lead to problems in the comparison with theory. In general, it is better to invest in more computer resources instead of distorting the definition of the algorithm.

5. *Maximal reconstruction efficiency:* The jet algorithm should efficiently identify all physically interesting jets (*i.e.*, jets arising from the energetic partons described by perturbative QCD).

6. *Ease of calibration:* The algorithm should not present obstacles to the reliable calibration of the final kinematic properties of the jet.

7. *Ease of use:* The algorithm should be straightforward to implement with typical experimental detectors and data.

8. *Fully specified:* Finally, the algorithm must be **fully specified**. This includes specifications for clustering, energy and angle definition, and all details of jet splitting and merging.

These experimental requirements are primarily a matter of optimization under real-life conditions and will, in general, exhibit complicated sensitivities to running conditions. They have a strong bearing on the ease with which quality physics measurements are achieved. Many of the details necessary to fully implement the jet algorithms have neither been standardized nor widely discussed and this has sometimes led to misunderstandings and confusion. The remainder of this chapter describes the cone and K_T algorithms discussed and recommended by the QCD at Run II Jets Group.

3. Cone Jet Algorithms

3.1. Introduction

This section should serve as a guide for the definition of common cone jet algorithms for the Tevatron and possibly future experiments. Section 3.2 reviews the features of previously employed cone algorithms. Section 3.3 describes a seedless cone algorithm. Section 3.4 gives a description of seed-based cone algorithms and discusses the need for adding midpoints between seeds as alternate starting points for clustering. Finally, in Section 3.5, we offer a detailed proposal for a common cone jet algorithm in Run II analyses.

3.2. Review of Cone Algorithms

Cone algorithms form jets by associating together particles whose trajectories (*i.e.*, towers whose centers) lie within a circle of specific radius R in $\eta \times \phi$ space. This 2-dimensional space is natural in $p\overline{p}$ collisions where the dynamics are spread out in the longitudinal direction. Starting with a trial geometric center (or axis) for a cone in $\eta \times \phi$ space, the energy-weighted centroid is calculated including contributions from all particles within the cone. This new point in $\eta \times \phi$ is then used as the center for a new trial cone. As this calculation is iterated the cone center "flows" until a "stable" solution is found, *i.e.*, until the centroid of the energy depositions within the cone is aligned with the geometric axis of the cone. This leads us to our initial cone algorithm based on the Snowmass scheme [2] of scalar E_T-weighted centers. The particles are specified by massless 4-vectors $(E^i = |\mathbf{p}^i|, \mathbf{p}^i)$ with angles $\left(\phi^i, \theta^i, \eta^i = -\ln\left(\tan(\theta^i/2)\right)\right)$ given by the direction from the interaction point with unit vector $\hat{\mathbf{p}}^i = \mathbf{p}^i/E^i$. The scalar E_T for each particle is $E_T^i = E^i \sin(\theta^i)$. For a specified geometric center for the cone $\left(\eta^C, \phi^C\right)$ the particles i within the cone satisfy

$$i \subset C \quad : \quad \sqrt{\left(\eta^i - \eta^C\right)^2 + \left(\phi^i - \phi^C\right)^2} \leq R. \tag{1}$$

In the Snowmass algorithm a "stable" cone (and potential jet) satisfies the constraints

$$\eta^C = \frac{\sum_{i\subset C} E_T^i \eta^i}{E_T^C}, \quad \phi^C = \frac{\sum_{i\subset C} E_T^i \phi^i}{E_T^C} \tag{2}$$

(*i.e.*, the geometric center of the previous equation is identical to the E_T-weighted centroid) with

$$E_T^C = \sum_{i\subset C} E_T^i \; . \tag{3}$$

Naively we can simply identify these stable cones, and the particles inside, as jets, $J = C$. (We will return to the practical issues of the impact of seeds and of cone overlap below.)

To complete the jet finding process we require a recombination scheme. Various choices for this recombination step include:

1. *Original Snowmass scheme:* Use the stable cone variables:

$$\begin{aligned} E_T^J &= \sum_{i\subset J=C} E_T^i = E_T^C \; , & (4)\\ \eta^J &= \frac{1}{E_T^J} \sum_{i\subset J=C} E_T^i \eta^i \; , & (5)\\ \phi^J &= \frac{1}{E_T^J} \sum_{i\subset J=C} E_T^i \phi^i \; . & (6) \end{aligned}$$

2. *Modified Run I recombination schemes:* After identification of the jet as the contents of the stable cone, construct more 4-vector-like variables:

$$\begin{aligned} E_x^i &= E_T^i \cdot \cos(\phi^i) \; , & (7)\\ E_y^i &= E_T^i \cdot \sin(\phi^i) \; , & (8)\\ E_z^i &= E^i \cdot \cos(\theta^i) \; , & (9)\\ E_{x,y,z}^J &= \sum_{i\subset J=C} E_{x,y,z}^i \; , & (10)\\ \theta^J &= \tan^{-1}\left(\frac{\sqrt{(E_x^J)^2 + (E_y^J)^2}}{E_z^J}\right) . & (11) \end{aligned}$$

A) In Run I, DØ used the scalar E_T^J sum as defined in Eq. 4 but used the following definitions for η^J and ϕ^J:

$$\begin{aligned} \eta^J &= -\ln\left(\tan(\frac{\theta^J}{2})\right) \; , & (12)\\ \phi^J &= \tan^{-1}(\frac{E_y^J}{E_x^J}) \; . & (13) \end{aligned}$$

B) In Run I, CDF used the angular definitions in Eqs. 12–13 and also replaced the Snowmass scheme E_T^J with:

$$E_T^J = E^J \cdot \sin(\theta^J), \quad E^J = \sum_{i\subset J} E^i \; . \tag{14}$$

Note that in the Snowmass scheme the designation of the centroid quantities η^J and ϕ^J of Eqs. 5 and 6 as a pseudorapidity and an azimuthal angle is purely convention. These quantities only approximate the true kinematic properties of the massive cluster that is the jet. They are, however, approximately equal to the "real" quantities, becoming exact in the limit of small jet mass ($M^J << E_T$). Further these quantities transform simply under longitudinal boosts (*i.e.*, η^J boosts additively while ϕ^J is invariant) guaranteeing that the jet structure determined with the Snowmass algorithm is boost invariant. It is also worthwhile noting that the Snowmass η^J is a better estimator of the "true" jet rapidity (y^J) defined below than the "true" jet pseudorapidity defined in Eq. 12. The latter quantity does not boost additively (for $M^J > 0$) and is not a good variable for systematic studies.

While the scalar sum E_T is invariant under longitudinal boosts, it is not a true energy variable. This feature leads to difficulty in resummation calculations: the kinematic boundary of the jet E_T shifts away from

$\sqrt{s}/2$ appropriate for two parton kinematics when additional final state partons are included and the jet acquires a nonzero mass. On the other hand the Snowmass variables have the attractive feature of simplicity, involving only arithmetic rather than transcendental relationships. An alternate choice, which we recommend here, is to use full 4-vector variables for the jets.

3. *E–Scheme, or 4-vector recombination:*

$$p^J = (E^J, \mathbf{p}^J) = \sum_{i \subset J=C} (E^i, p_x^i, p_y^i, p_z^i) , \quad (15)$$

$$p_T^J = \sqrt{(p_x^J)^2 + (p_y^J)^2} , \quad (16)$$

$$y^J = \frac{1}{2} \ln \frac{E^J + p_z^J}{E^J - p_z^J} , \quad \phi^J = \tan^{-1} \frac{p_y^J}{p_x^J} . \quad (17)$$

Note that in this scheme one does *not* use the scalar E_T variable. The 4-vector variables defined above manifestly display the desired Lorentz properties. Phase space boundaries will exhibit the required stability necessary for all-order resummations. While the structure of analytic fixed order perturbative calculations is simpler with the Snowmass variables, NLO cross section calculations are now also possible with Monte Carlo programs [8, 9, 10, 11]. Such programs are fully flexible with respect to the choice of variables and the 4-vector variables pose no practical problems. It is also important to recall that, at least at low orders in perturbation theory, it is not possible for energy to be conserved in detail in going from the parton level to the hadron level. At the parton level the jet will almost surely be a cluster of partons with non-zero color charge. At the hadron level the cluster will be composed of color-singlet hadrons. The transition between the two levels necessarily involves the addition (or subtraction) of at least one colored parton carrying some amount (presumably small) of energy.

One can also employ these true 4-vector variables, rather than the E_T-weighted centroid, in the jet algorithm to find stable cones. While this choice will complicate the analysis, replacing simple arithmetic relationships with transcendental relationships, the group recommends that this possibility be investigated. The goal is to have a uniform set of kinematic variables with appropriate Lorentz properties throughout the jet analysis.

At this point it might seem that a simple and straightforward jet definition would arise from just the choice of a cone size and a recombination scheme. The algorithm would then be used to scan the detector and simply find all stable cones. In practice, this naive algorithm was found to be incomplete. To keep the time for data analysis within reasonable bounds the concept of the seed was introduced. Instead of looking "everywhere" for stable cones, the iteration process started only at the centers of seed towers that passed a minimum energy cut (how could a jet not have sizeable energy deposited near its center?). Additionally, in Run I both CDF and DØ reduced the number of seed towers used as starting points by consolidating adjacent seed towers into single starting points. (The actual clustering was always performed on calorimeter towers.) These types of procedures, however, create the problems illustrated in Figs. 1, 2 and 3, introducing sensitivity to soft emissions and the possibility of collinear sensitivity.

The naive Snowmass algorithm also does not address the question of treating overlapping stable cones. It is quite common for two stable cones to share some subset (but not all) of their particles. While not all particles in the final state need to be assigned to a jet, particles should not be assigned to more than one jet. Hence there must be a step between the stable cone stage and the final jet stage where either the overlapping cones are merged (when there is a good deal of overlap) or the shared particles are split between the cones. Typically cones whose shared energy is larger than a fixed fraction (*e.g.*, $f = 50\%$) of the energy in the lower energy cone are merged. For the cases with shared energy below this cut, the shared particles are typically assigned to the cone that is closer in $\eta \times \phi$ space. As suggested earlier, the detailed properties of the final jets will depend on the merge/split step and it is essential that these details be spelled out in the algorithm. We provide examples in the following sections.

3.3. Cone Jets without Seeds

Since many of the issues outlined in the previous section arise from the use of seed towers to define the starting point in the search for stable cones, it is worthwhile to consider the possibility of a seedless cone algorithm. A seedless algorithm is infrared insensitive. It searches the entire detector and finds all stable cones (or proto-jets[2]), even if these cones do not have a seed tower at their center. Collinear sensitivity is also removed, because the structure of the energy depositions within the cone is unimportant. In this section we present a preliminary study of such an algorithm.

3.3.1. Seedless Jet Clustering

We give an example of a seedless algorithm in the flowchart in Fig. 4. The basic idea [12] follows from the concept of "flowing" cone centers mentioned earlier. The location of a stable cone will act as an at-

[2] At the clustering stage we refer to stable cones as proto-jets. These may be promoted to jets after surviving the splitting and merging stage.

tractor towards which cones will flow during the iteration process. If the process starts close to such a stable center, the flow steps will be small. Starting points further from a stable center will exhibit larger flow steps towards the stable center during the iteration. Starting points outside of the region of attraction will again exhibit small flow steps. The method starts by looping through *all* detector towers[3] in some appropriate fiducial volume. For each tower k, with center $\overrightarrow{k} = (\eta^k, \phi^k)$, we define a cone of size R centered on the tower

$$\begin{aligned} \overrightarrow{C^k} &= \left(\eta^{C^k} = \eta^k, \phi^{C^k} = \phi^k\right), \\ i &\subset C^k : \sqrt{\left(\eta^i - \eta^{C^k}\right)^2 + \left(\phi^i - \phi^{C^k}\right)^2} \leq R. \end{aligned} \tag{18}$$

For each cone we evaluate the E_T-weighted centroid

$$\overrightarrow{\overline{C}^k} = \left(\bar{\eta}^{C^k}, \bar{\phi}^{C^k}\right), \tag{19}$$

$$\bar{\eta}^{C^k} = \frac{\sum_{i\subset C^k} E_T^i \eta^i}{E_T^{C^k}}, \ \bar{\phi}^{C^k} = \frac{\sum_{i\subset C^k} E_T^i \phi^i}{E_T^{C^k}}, \tag{20}$$

$$E_T^{C^k} = \sum_{i\subset C^k} E_T^i . \tag{21}$$

Note that, in general, the centroid $\overrightarrow{\overline{C}^k}$ is not identical to the geometric center $\overrightarrow{C^k}$ and the cone is not stable. While this first step is resource intensive, we simplify the subsequent analysis with the next step. If the calculated centroid of the cone lies outside of the initial tower, further processing of that cone is skipped and the cone is discarded. The specific exclusion distance used in this cut is a somewhat arbitrary parameter and could be adjusted to maximize jet finding efficiency and minimize the CPU demand of the algorithm. All cones that yield a centroid within the original tower become preproto-jets. For these cones the process of calculating a new centroid about the previous centroid is iterated and the cones are allowed to "flow" away from the original towers. This iteration continues until either a stable cone center is found or the centroid migrates out of the fiducial volume. The surviving stable cones constitute the list of proto-jets. Note that the tower content of a cone will vary as its center moves within the area of a single tower. For a cone of radius R and tower dimension Δ (in either η or ϕ) the minimum change in the cone center location for which the tower content in the cone changes by at least one tower is characterized by $\Delta^2/2R$. This distance is of order 0.007 for $\Delta = 0.1$ and $R = 0.7$ (*i.e.*, 10% of a tower width if the diameter of the cone, $2R$, is ten times a tower width).

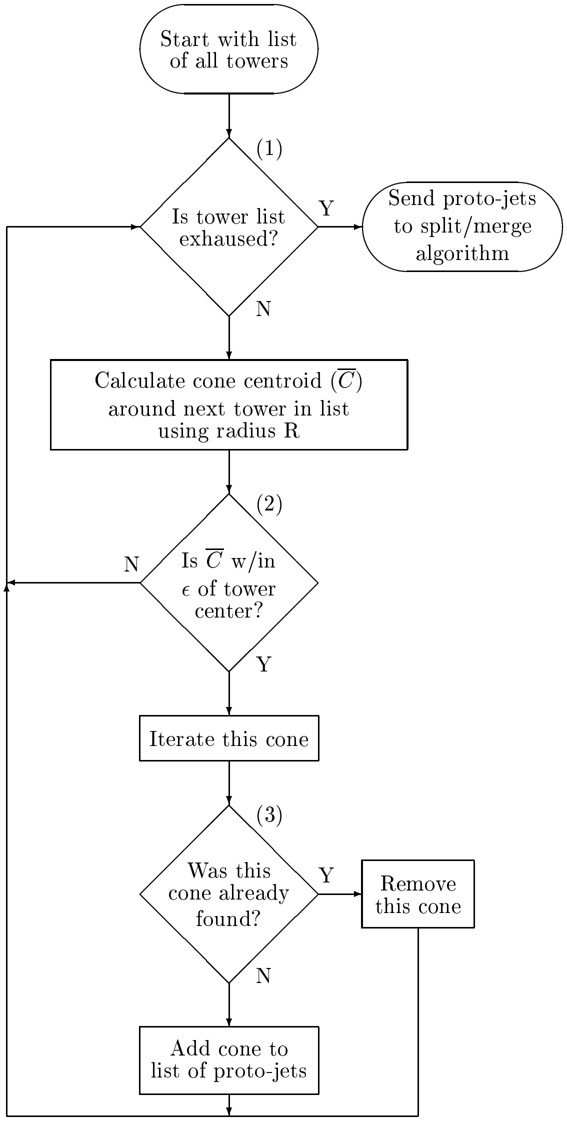
Start with list of all towers
(1)
Is tower list exhaused?
Y
Send proto-jets to split/merge algorithm
N
Calculate cone centroid ($\overline{C}$) around next tower in list using radius R
(2)
N
Is $\overline{C}$ w/in ϵ of tower center?
Y
Iterate this cone
(3)
Was this cone already found?
Y
Remove this cone
N
Add cone to list of proto-jets

Figure 4. A seedless clustering algorithm.

[3] While the algorithm may be run on individual detector cells, we do not believe that cell-level clustering is within the CPU means of current experiments for the largest expected data samples.

An even more streamlined option would be to keep only those cones that yield a stable cone center without leaving the original tower. Since a trial cone is originally placed at the center of every tower, the only distinct stable cone centers missed by this (much!) faster algorithm correspond to very limited regions of attraction (less than the area of a tower). Such situations can arise in only two cases. One possibility is that there are two (or more) stable directions within a single tower. The second possibility is that there is a stable direction within a tower but it is not found starting at the tower center. While both of these scenarios arise in analyses of realistic data, they do not constitute cause for concern. Proto-jets with directions that are nearly collinear (*i.e.*, that lie within a single tower) will have nearly the same tower content and be merged with little impact on the final jet properties. Isolated stable directions with very small regions of attraction (the second case) are most likely fluctuations in the background energy level and not the fingerprints of real emitted partons. In any case the stable cone centers not found by the streamlined algorithm invariably correspond to low E_T proto-jets and are well isolated from large E_T proto-jet directions (otherwise they would be attracted into the larger E_T jet). Thus the leading E_T jets (after merging and splitting) found by either the original seedless algorithm or the streamlined version are nearly identical.

For practical use it may also be necessary to apply some minimum E_T threshold to the list of proto-jets. Ideally such a threshold would be set near the noise level of the detector. However, a higher setting might be warranted to reduce the sensitivity of the algorithm to energy depositions by multiple interactions at high luminosities (see Section 3.3.4 for details of seedless clustering at the detector level).

In general, a number of overlapping cones, where towers are shared by more than one cone, will be found after applying the stable cone finding procedure. As noted earlier, the treatment of proto-jets with overlapping regions can have significant impact on the behavior of the algorithm.

3.3.2. Splitting and Merging Specifications

A well-defined algorithm must include a detailed prescription for the splitting and merging of proto-jets with overlapping cones. We provide an outline of a splitting and merging algorithm in Fig. 5. It is important to note that the splitting and merging process does not begin until all stable cones have been found. Further, the suggested algorithm always works with the highest E_T proto-jet remaining on the list and the ordering of the list is checked after each instance of merging or splitting. If these conditions are not met, it is difficult to predict the behavior of the algorithm for multiply split and/or merged jets and similar lists of proto-jets can lead to distinctly different lists of jets. This undesirable situation does not arise with the well-ordered algorithm in Fig. 5. While there will always be some order dependence in a splitting and merging scheme when treating multiply overlapping jets, we recommend fixing this order by starting with the highest E_T proto-jet and working down in the E_T ordered list. In this way the action of the algorithm is to prefer cones of maximal E_T. Note that, after a merging or splitting event, the E_T ordering on the list of remaining proto-jets can change, since the survivor of merged jets may move up while split jets may move down. Once a proto-jet shares no towers with any of the other proto-jets, it becomes a jet and is not impacted by the subsequent merging and splitting of the remaining proto-jets. As noted earlier and illustrated in Fig. 5, the decision to split or merge a pair of overlapping proto-jets is based on the percentage of transverse energy shared by the lower E_T proto-jet. Proto-jets sharing a fraction greater than f (typically $f = 50\%$) will be merged; others will be split with the shared towers individually assigned to the proto-jet that is closest in $\eta \times \phi$ space. This method will perform predictably even in the case of multiply split and merged jets. Note that there is no requirement that the centroid of the split or merged proto-jet still coincides precisely with its geometric center.

3.3.3. Parton Recombination

The definition of calorimeter towers, *i.e.*, a discretization of (η, ϕ) space, would be cumbersome in a theoretical calculation, and is indeed not necessary. In a theoretical calculation at fixed order, the maximal number of partons, n, is fixed. With specified parton momenta, the only possible positions of stable cones are then given by the partitions of the n parton momenta, *i.e.*, there are at most $2^n - 1$ possible locations of proto-jets. They are given by the positions of individual partons, all pairs of partons, all combinations of three partons, *etc.* In a perturbative calculation, *e.g.* via a NLO Monte Carlo program, the proto-jet selection of the seedless algorithm can then be defined as follows:

1. Make a list of centroids for all possible parton multiplets. These are derived from the coordinates of all parton momenta p_i, of all pairs of parton momenta $p_i + p_j$, of all triplets of parton momenta $p_i + p_j + p_k$, *etc.* For each centroid record which set of partons defines it.

2. Select the next centroid on the list as the center of a trial cone of radius R.

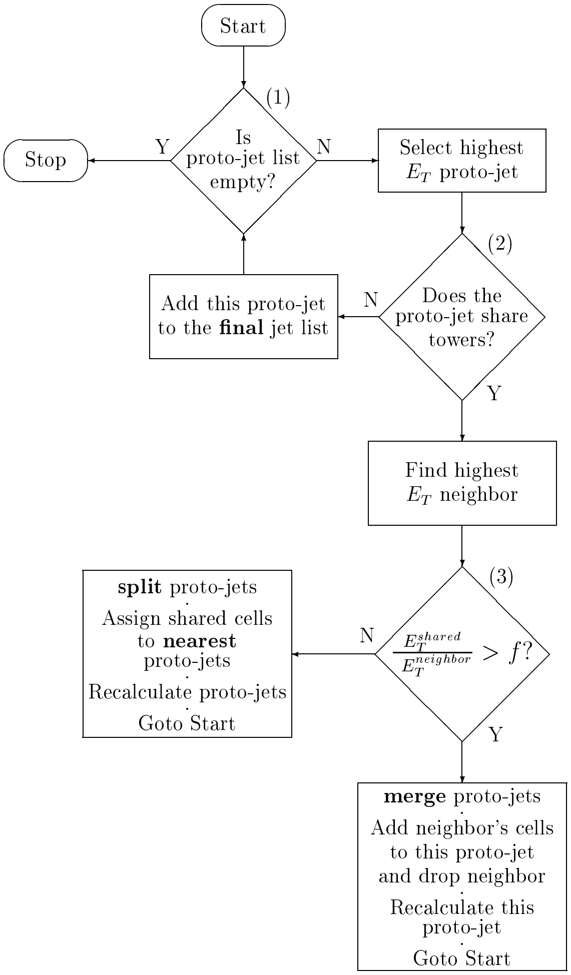
Start
(1)
Is proto-jet list empty?
Y
Stop
N
Select highest E_T proto-jet
(2)
Does the proto-jet share towers?
N
Add this proto-jet to the **final** jet list
Y
Find highest E_T neighbor
(3)
$\frac{E_T^{shared}}{E_T^{neighbor}} > f$?
N
split proto-jets
Assign shared cells to **nearest** proto-jets
Recalculate proto-jets
Goto Start
Y
merge proto-jets
Add neighbor's cells to this proto-jet and drop neighbor
Recalculate this proto-jet
Goto Start

Figure 5. A fully specified splitting and merging algorithm.

Go to the split/merge stage if the list of cone centers is exhausted.

3. Check which partons are inside the trial cone.

4. If the parton list of the centroid and that of the trial cone disagree, discard the trial cone and go to (2). If the lists agree, add the set of partons inside the trial cone as a new entry to the list of proto-jets.

As before, different proto-jets may share partons, *i.e.* they may overlap. The required split/merge step is then identical to the calorimeter-level steps (Fig. 5), with towers replaced by partons as elements of proto-jets.

In the case of analytic evaluations of the NLO perturbative jet cross section [13] the integrations over the multi-parton phase space are divided into various disjoint contributions. For a jet of fixed E_T^J, η^J and ϕ^J we have only the cases where a) one parton is in the jet direction with the jet E_T, and the other partons are excluded from nearby directions where they could fit in a jet cone with the first parton, or b) two partons fit in a single cone with their centroid properties constrained to be the jet values. The questions of overlap, splitting and merging never arise at this order for $R < \pi/3$.

3.3.4. Tests of a Seedless Algorithm

In this section we offer some insight into the performance of the seedless cone algorithm applied to a detector. We begin by examining a simulated large-E_T jet event in the DØ detector (Fig. 6). The event was chosen from a sample generated with PYTHIA [14] using a 160 GeV minimum E_T cut at the parton-level generator. After hadronization, the events were processed through a full simulation of the DØ detector. The towers in the central region ($-3.2 < \eta < 3.2$) are 0.1×0.1 in size. Fig. 6 shows the distribution of calorimeter tower E_T's for the event in the central fiducial volume ($-2.4 < \eta < 2.4$) where cones of $R = 0.7$ can be fully contained in the central region. Three jets clearly dominate the display (along with a less distinctive feature at the large η boundary near $\phi = 4$). Fig. 7 shows the E_T contained in a cone of radius 0.7 centered at each calorimeter tower, displaying the same structure for the event in a slightly different language. We can make this picture even more clear by appealing to the "flow imagery" of Section 3.3.1. We define a flow vector as the 2-dimensional vector difference between the calculated centroid for a cone centered on a tower and the geometric center of the tower ($\overrightarrow{\bar{C}^k} - \overrightarrow{C^k}$ in Eqs. 18 and 19). This vector vanishes for a stable cone. This flow vector is plotted in the corresponding range of $\eta \times \phi$ in Fig. 8 for the same PYTHIA generated event.

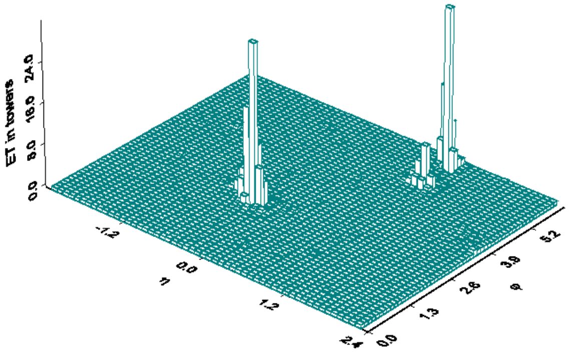

Figure 6. Calorimeter tower E_T lego plot for a simulated large-E_T jet event in the DØ Calorimeter.

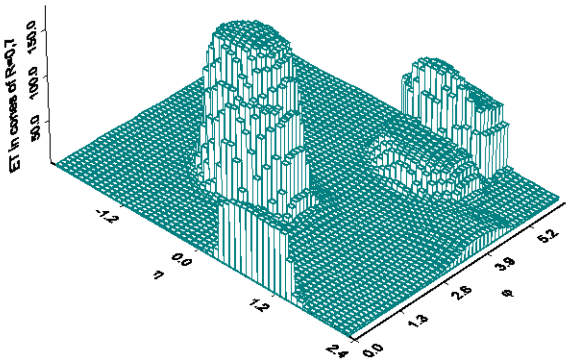

Figure 7. E_T in cones centered on each calorimeter tower (in $|\eta^{tower}| < 2.4$) for the simulated large-E_T jet event of Fig. 6.

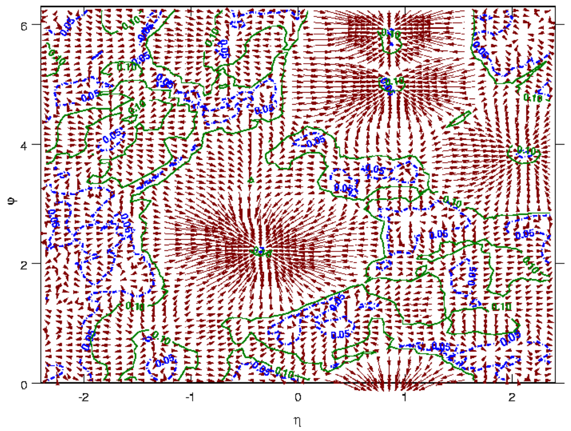

Figure 8. Energy flow for the cones in the large-E_T jet event of Figs. 6–7. The contours bound flow regions with vector magnitude < 0.1 (solid contours) and < 0.05 (dashed contours) in $\eta \times \phi$.

The flow vector clearly points to the four potential jets noted above. Cones that are in the neighborhood of a potential jet exhibit flow vectors of large magnitude pointing towards the jet center. This magnitude will generally be sufficient to cause the cone to fail the second test in Fig. 4, thus preventing further iteration of the cone to define a proto-jet. The contours of Fig. 8 bound regions of flow with magnitude < 0.1 (solid contours) and < 0.05 (dashed contours) in $\eta \times \phi$, within which we expect to find the final jets. It is important to note the size of the detector regions with small flow magnitude. Regions with sufficiently small flow will pass test (2) in the clustering stage and allow the cone to undergo additional iterations. This ultimately increases processing time for clustering and complexity in splitting and merging (due to the production of many additional proto-jets). The flow magnitude cut has a natural size on the order of the detector tower size. For the DØ detector, with a typical towers size of $\eta \times \phi = 0.1 \times 0.1$, the cut would be between the two contours shown above. A too small magnitude cut will cause inefficiencies in jet finding; too large a cut will cause iterations on cones over the whole detector volume.

It is clear from Figs. 6–8 that the region of interest around the jets is much smaller than the area contained within the contours of "stable" cones. There are broad "plains" of low energy deposition where the flow vector is of small magnitude, but also of rapidly varying direction. Stable cones are found in these regions. But these presumably arise simply from local fluctuations yielding local extrema and are not expected to correspond to the fingerprints of underlying (energetic) partons. There are at least two, possibly parallel paths to follow in order to reduce the impact of these regions on the analysis, in terms of both required resources and final results.

As already noted, we can further streamline the analysis by applying the cut on the flow vector at each step in the iteration. Thus we keep only those cones that do not "flow" outside of their original tower before a stable center is reached. Such an algorithm converges rapidly to the stable cones pointed to by the largest magnitude flow vectors in Fig. 8 and efficiently eliminates most of the cones in the "plains". We do lose the stable cones that a full iteration, allowing any amount of flow, finds in the flat regions of the previous figures. However, as already emphasized, these cones do not

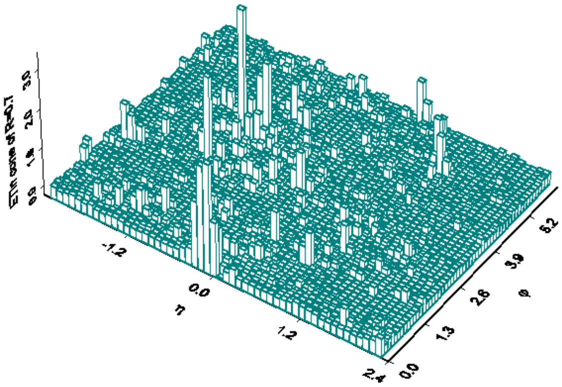

Figure 9. A sample event from data. Tower E_T lego plot for an event passing the DØ $W \rightarrow jets$ trigger.

correspond to the physics we wish to study with jet analyses. With a large savings in analysis time the streamlined algorithm finds the same leading jet properties (*e.g.*, E_T and η^J) as the more complete algorithm to a fraction of a percent. The final jets contain typically 120 to 160 towers. The differences between the leading jets found with the two algorithms arise from differences in tower content of just 1 or 2 towers (at the cone boundary).

One can also reduce the effort and the final event complexity by applying a minimum E_T cut on the cones at the proto-jet stage. An obvious choice for this minimum E_T cut would be to place it above the level of detector noise. As alluded to in Section 3.3.1, a practical cut might be placed slightly higher to reduce sensitivity to varying event pileup with changes in beam luminosity. Unfortunately, this places a rather arbitrary threshold into the algorithm from the standpoint of theoretical calculations, *i.e.* what is the 'noise' level at NLO? Additionally, such cuts will in practice be applied before final jet scale corrections. How does X GeV uncorrected in the experiment compare to X GeV at generator level? Such experiment specific considerations clearly are out of the realm of event generator design! A possible improvement would be to set a minimum cone E_T threshold equal to some fraction of the scalar E_T in the event. In this way such effects will tend to partially cancel between generators and experiments, better relating the cut between the two levels.

We next look at an example of the seedless algorithm tested on actual calorimeter data. Fig. 9 shows the tower E_T lego plot for a DØ event passing a $W \rightarrow jets$ trigger. The trigger required at least two central jets

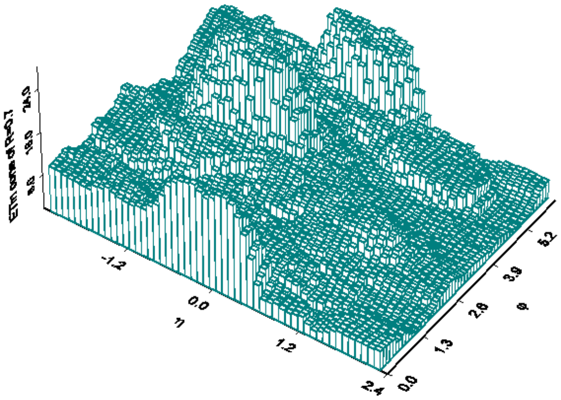

Figure 10. E_T in cones centered on each calorimeter tower (in $|\eta^T| < 2.4$) for the $W \rightarrow jets$ sample event of Fig. 9.

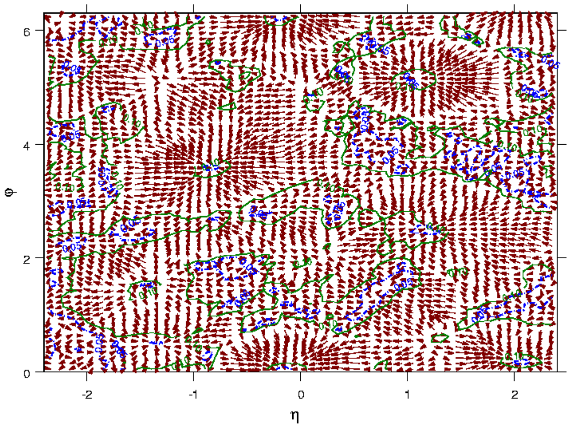

Figure 11. Energy flow for the cones in the $W \rightarrow jets$ event of Figs. 9–10. The contours bound flow regions with vector magnitude < 0.1 (solid contours) and < 0.05 (dashed contours) in $\eta \times \phi$.

with $E_T > 15$ GeV. These data were taken at high luminosity with an average of ∼2.8 interactions per beam crossing. The two leading jets that pass the cut are reasonably obvious (along with, perhaps, two other subleading jets) but overall this event is clearly noisier (more realistic) than the PYTHIA generated event. This point is illustrated also in Figs. 10 and 11, which show the cone energy and flow vectors for this event, analogous to Figs. 7 and 8. In this case the baseline energy subtraction for calorimeter cell energies in the data leads to towers with (small) negative energy deposition.

The increased level of noise and the possibility of negative tower energy results in two new issues for the jet analysis that were not observed in the analysis of the Monte Carlo data. The negative energy cells allow true stability with respect to the iteration process to be replaced by limit cycles. Iteration leads not only to cone center locations for which $\overrightarrow{\bar{C}^j} - \overrightarrow{C^j} = 0$ but also, for example, to doublets of locations for which $\overrightarrow{\bar{C}^1} = \overrightarrow{C^2}$ and $\overrightarrow{\bar{C}^2} = \overrightarrow{C^1}$, or $\overrightarrow{\bar{C}^1} - \overrightarrow{C^1} = -(\overrightarrow{\bar{C}^2} - \overrightarrow{C^2})$. Thus continued iteration simply carries the cone center back and forth between location 1 ($\overrightarrow{C^1}$) and location 2 ($\overrightarrow{C^2}$). (More complex multiplets of locations with sets of 3, or even 6, 2-dimensional flow vectors summing to 0 are also observed.) The good news is that these clusters of cone centers are typically close by each other and yield essentially the same final jets, after merging, independent of where in the limit cycle the iteration process is terminated. This is guaranteed to be true for the streamlined algorithm where the entire cycle must occur within a single tower. (The $(\eta \times \phi)$ distance between two members of such a limiting cycle driven by a negative tower energy of magnitude E_N is approximately $R \cdot E_N/E_C$, where E_C is the total energy in the cone. This can be as small as the minimum distance for a change of one tower in the cone as noted above, *i.e.*, 7% of a tower width.)

The noisy quality of the event leads to an even more troubling phenomenon. There are so many locally stable cone centers found in the now rapidly fluctuating "plain" region that the proto-jet list may exhibit a surprisingly large number of mutually overlapping cones. During the merging phase these can coalesce into jets with large (even leading) E_T. This issue has historically been treated by applying a minimum E_T cut to the proto-jet list before merging and splitting. With the event studied here a cut of 8 GeV (typical of values used by DØ) is not sufficient. If we keep all stable cones with $E_T > 8$ GeV, with no other cuts, as proto-jets, the merging process builds a leading jet by pulling together many cones where there is clearly no real jet. This problem does not arise in the streamlined algorithm where only stable cones that stayed within their original tower are kept. In this case the algorithm identifies the leading jets anticipated intuitively from the above figures.

3.3.5. Comments on the Seedless Clustering

We may summarize the advantages of the seedless clustering described above as follows:

1. Avoids undesirable sensitivity to soft and collinear radiation.

2. Offers increased efficiency for all physically interesting jets.

3. Offers improved treatment of limit cycles and overlapping cones.

4. "Flow cut" method offers more efficient use of computer resources than unrestricted seedless clustering.

We have not investigated further improvements in the optimization of the computational efficiency for this seedless algorithm. However, some improvement may be gained by using the fact that cones centered on adjacent towers are largely overlapping, thus reducing the number of towers to sum for each new center. Other improvements such as region of interest (ROI) clustering may also be explored.

3.4. Cone Jets with Seeds

In an actual experiment the number of calorimeter towers may be very large (order 6000 for tower sizes of $\Delta\eta \times \Delta\phi = 0.1 \times 0.1$ and an η coverage of ± 5 units of pseudorapidity). The above seedless algorithm may then be expensive computationally. The question arises whether an acceptable approximation of the seedless algorithm can be constructed, analogous to the parton-level short cut, while considering primarily those towers which have energy depositions above a minimal seed threshold for finding proto-jets.

Seed-based cone algorithms offer the advantage of being comparatively efficient in CPU time. In a typical application, detector towers are sorted according to descending E_T and only towers passing a seed cut,

$$E_T^{tower} > E_T^{seed} \; , \qquad (22)$$

are used as starting points for the initial jet cones. This greatly reduces the number of cones that need to be evaluated in the initial stage. The seed threshold E_T^{seed} must be chosen low enough so that variations of E_T^{seed} lead to negligible variations in any observable under consideration. The simple seed-based algorithm is sensitive to both infrared or collinear effects. However, sensitivity to the splitting of the seed E_T between multiple towers is greatly reduced for larger E_T jets. As stated above, this is true when the jet reconstruction becomes 100% efficient (*i.e.*, around 20 GeV for jets in DØ). For fully efficient jet algorithms the collinear dependency is reduced to a second-order effect, namely, the effective number of low E_T proto-jets that may engage in splitting and merging. In a typical algorithm a minimum E_T cut may also be applied to each proto-jet to prevent excessive merging of noise and energy not associated with the hard scattering producing the jets.

3.4.1. Addition of Midpoints

The seedless algorithm discussed previously can be approximated by a seed-based algorithm with the addition of 'midpoints' in the list of starting seeds. The idea [15] is to duplicate the parton-level algorithm discussed in Section 3.3.3, but with partons replaced by seeds. By adding a starting point for clustering at the positions given by p_i+p_j, $p_i+p_j+p_k$ etc., the sensitivity of the algorithm to soft radiation as illustrated in Fig. 1 is essentially removed. Since widely separated seeds cannot be clustered to a proto-jet, it is sufficient to only consider those midpoints where all seeds lie within a distance

$$\Delta R < 2.0 \cdot R_{cone} \tag{23}$$

of each other.

With these changes, the resulting algorithm is quite close to those used in Run I of the Tevatron. The main change is the inclusion of midpoints of seeds (the p_i+p_j pairs) and of centers of larger numbers of seeds as additional seed locations for trial cones. Two studies of the effects of adding midpoints were completed during the workshop and are summarized below. The first checks the infrared safety of the midpoint algorithm, also called the Improved Legacy Cone Algorithm (ILCA), in a Monte Carlo study. The second tests the effect of adding midpoints on the performance of the Run I DØ cone algorithm.

3.4.2. Results from a Monte Carlo Study

The request for an infrared and collinear safe jet-algorithm is most important from the viewpoint of perturbative QCD calculations. Unsafe algorithms simply do not permit unambiguous results, once higher order corrections are considered [16, 17]. Instead results will depend on the technical regularization procedure adopted in a specific calculation.

The deficiencies of an unsafe algorithm will only show up at sufficiently high order in the perturbative expansion. For example, the jet merging due to soft gluon radiation as depicted in Fig. 1 will only become a problem when three partons or more can be combined to a single jet. In hadron collider processes this first happens in, for example, the NLO corrections to three-jet production [8], where four-parton final states are included in the real emission contributions. The fourth parton is needed to provide the necessary recoil transverse momentum to the other three partons which may or may not form a single jet. The NLO three-jet Monte Carlo is very CPU intensive, however, making it a cumbersome tool to investigate jet algorithms, at present. A much faster probe is provided by the existing NLO dijet Monte Carlos in DIS [10, 11].

In $ep \to ejjX$, the electron provides the necessary recoil p_T to the final-state partons. The real emission QCD corrections at $\mathcal{O}(\alpha_s^2)$ thus contain three partons which can be close together. Their merging to a single jet, with the concomitant loss of two-jet cross section, is a probe of the infrared safety of the two-jet vs. one-jet classification of partonic events. A second probe is provided by the E_T flow inside a jet, which has recently been modeled with up to three partons in a single jet, for the current jets in DIS [18].

We have investigated these issues with the MEPJET Monte Carlo [10], which calculates dijet production in DIS at NLO. The program was run in a kinematical range typical for HERA, ep collisions at $\sqrt{s} = 300$ GeV with $Q^2 > 100$ GeV^2. Reconstructed jets were required to satisfy

$$E_T > 10 \text{ GeV}, \qquad -1 < y < 2, \qquad R_{jj} < 2, \tag{24}$$

where E-scheme recombination is used. Here R_{jj} is the separation of reconstructed jets in the legoplot. Following HERA practice, we use a cone size $R = 1$. Considering jets with a maximal separation of twice the cone size enhances the statistical significance of any splitting/merging effects in the Monte Carlo calculation.

With these settings two cone algorithms are considered to investigate the importance of extra midpoints in the perturbative results. The first is the seedless algorithm in its parton-level implementation as described in Section 3.3.3, which we here call the "midpoint" algorithm. In order to test the analog of tower threshold effects, only partons with $E_{T,i} > E_T^{seed}$ are considered for centers of trial cones, *i.e.*, trial cone centroids are the directions of these partons and their midpoints $p_i + p_j$ and $p_i + p_j + p_k$. The second algorithm, dubbed "no center seed" is identical, except that the midpoints are left out as trial cone centers. For both algorithms, the final splitting/merging decision is made with an E_T-fraction of $f = 0.75$ of the lower E_T proto-jet as the dividing line.

The MEPJET program is based on the phase space slicing method, with a parameter s_{min} defining the separation between three-parton final states on the one hand, and the virtual contributions plus soft and collinear real emission processes (which cancel the divergences of the virtual graphs) on the other. This dividing line is completely arbitrary and observables

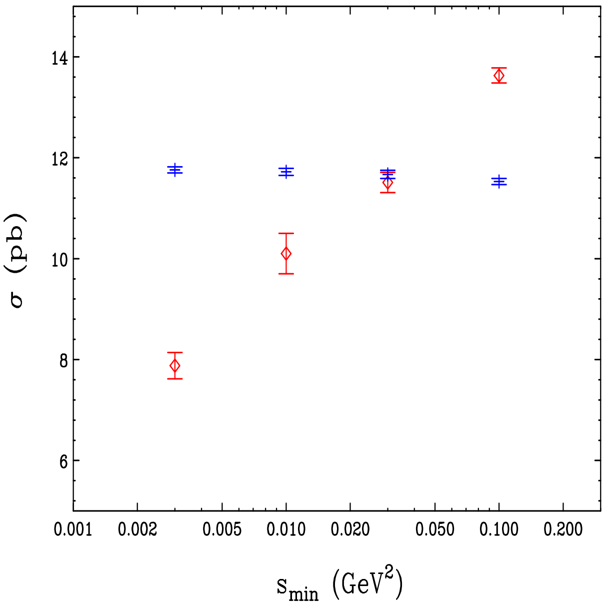

Figure 12. Dependence of the DIS dijet cross section on s_{min} for the ILCA algorithm with midpoints (plain symbols) and for the "no center seed" algorithm (diamonds).

should not depend on it. A test of this requirement is shown in Fig. 12 where the dijet cross section within the cuts of Eq. 24 is shown as a function of s_{min}. Whereas the midpoint algorithm shows s_{min}-independence within the statistical errors of the Monte Carlo (plain symbols), leaving out the midpoints between partons leads to a pronounced decrease of the cross section as s_{min} becomes smaller. Smaller s_{min} implies that more events are generated as explicit three-parton final states. The additional soft gluons act as extra seeds that tend to merge the two jets, leaving the event classified as a one-jet event, which does not contribute to the plotted dijet cross section. The s_{min} dependence of the "no center seed" algorithm means that no perturbative prediction is possible for this algorithm: as s_{min} approaches zero, the dijet cross section diverges logarithmically as $\log s_{min}/Q^2$.

Even when fixing s_{min} to some typical soft QCD scale, like $s_{min} = 0.03$ GeV^2, the "no center seed" algorithm has fatal defects. This is demonstrated in Fig. 13 where the variation of the dijet cross section within the cuts of Eq. 24 is shown as a function of "tower threshold" transverse energy E_T^{seed}. The midpoint algorithm is almost independent of this threshold, as long as E_T^{seed} is less than about 10% of the jet transverse energy. The "no center seed" algorithm, on the other hand, shows a pronounced threshold dependence, raising the specter of substantial dependence of jet cross sections on detector thresholds, detector response to soft particles and nonperturbative effects. These effects have been discussed previously for three-jet events at the Tevatron [8, 16].

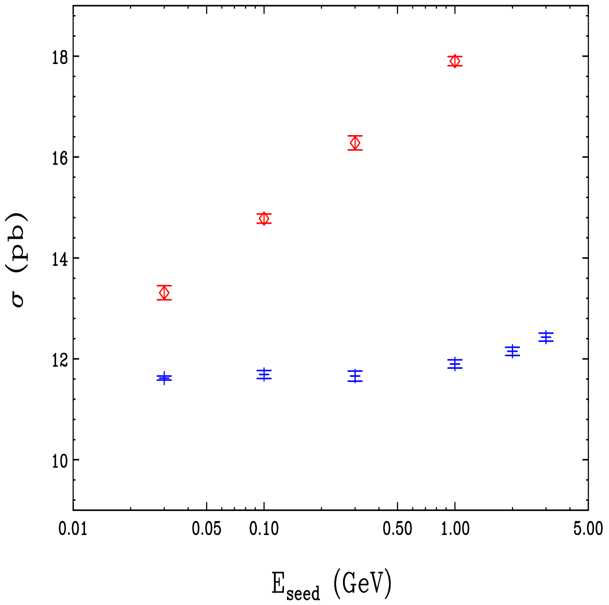

Figure 13. Dependence of the DIS dijet cross section on the seed threshold E_T^{seed} of Eq. 22. Results are shown for ILCA, with midpoints (plain symbols) and for a "no center seed" variant (diamonds).

Discarding the "no center seed" algorithm we turn to internal E_T flow inside a single jet as another measure of the performance of jet algorithms. The differential jet shape, $\rho(r)$, is defined as $1/\Delta r$ times the average E_T fraction of a jet in a narrow ring of width Δr, a distance r from the jet axis. In Fig. 14 the differential jet shape is shown for current jets at HERA, in the phase space region

$$E_T > 14 \text{ GeV}, \qquad -1 < \eta < 2 \tag{25}$$

for DIS events with $Q^2 > 100$ GeV^2. Results are shown for the midpoint (ILCA) and the K_T algorithm (to be described later) at NLO ($\mathcal{O}(\alpha_s^2)$). The midpoint algorithm produces wider jets than the K_T algorithm with

$D = R$, as is to be expected since two partons with a separation slightly less than $2R$ can be clustered by the midpoint, but not the K_T algorithm. NLO corrections are quite small for the midpoint algorithm. We have also checked that the jet shapes in the midpoint algorithm exhibit good scale dependence at NLO, similar to the K_T algorithm [18].

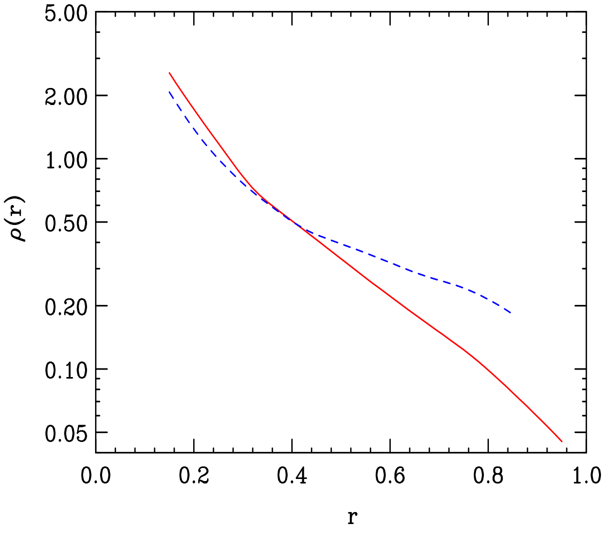

Figure 14. Jets shapes in ILCA (dashed line) compared to K_T (solid line).

3.4.3. Results from Data Study

A midpoint algorithm has previously been employed by the OPAL Collaboration [19]. We now report a study performed using the DØ data. The data were acquired from a two-jet trigger sample with an average of 2.8 interactions per beam crossing. The goal of the data-based study was to test the sensitivity of DØ's Run I cone algorithm to the addition of midpoints. To facilitate a direct comparison of Run II jet results with the current data it is desirable that algorithms supported[4] for the new data produce similar results.

Details in the DØ Run I jet algorithm forced the splitting and merging of jets to occur as they are found. In effect this defines an order dependence based on the seed E_T of the jets. It was possible to test two orderings in the jet clustering. In the first case, jets were initially found around all seed towers above a 1 GeV threshold, then around all midpoints. In the second case they were first found around all midpoints between seed towers, then around the seed towers themselves. Fig. 15 shows the E_T distributions for three trials, the legacy seed, seed + midpoint, and midpoint + seed trials. Also shown are the ratios of the E_T spectra. A cone radius of 0.7 was used.

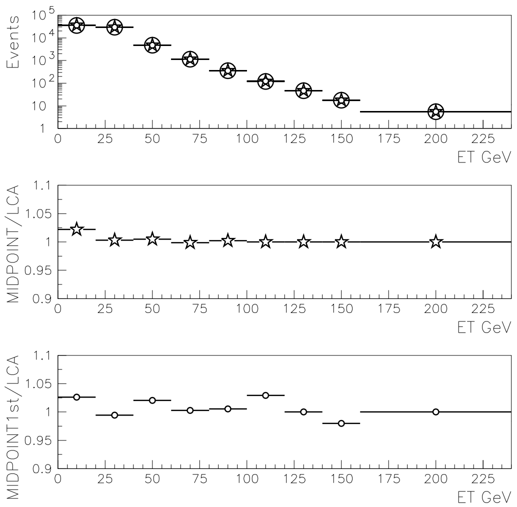

Figure 15. Jet E_T distributions and ratios. Top: Jet E_T distributions for the three algorithms overlayed. Legacy seeds (large circles), seeds + midpoints (stars), midpoints + seeds (small circles). Middle: Seeds + midpoint distribution divided by the legacy distribution. Bottom: Midpoint + seeds distribution divided by the legacy distribution.

There are two effects to observe in Fig. 15. First, the addition of midpoints tends to cause an increase in the number of low E_T jets. This is because the midpoints are effectively zero threshold seeds, therefore very soft jets that tend to fail reconstruction by falling short of the seed requirement may sometimes be reconstructed around a midpoint. Second, the results are different depending on the order in which the seeds + midpoints are used. However, we can safely conclude that the addition of midpoints has little more than a few percent effect on the experimental jet E_T distribution.

Fig. 16 shows the ratio of the leading jet for the legacy seed and midpoint + seed algorithms. Since

[4] While any number of jet algorithms may in principle be included in an offline analysis stream, in practice only a few algorithms will typically be fully supported by detailed energy scale, resolution, and efficiency corrections.

a meaningful test requires the comparison of the same jets, the jets were also required to be matched within a radius of 0.2 (in $\Delta\eta \times \Delta\phi$) to prevent accidental comparisons of unrelated jets due to ‘flipping’ of the jet order between algorithms. Fig. 17 shows the fractions of isolated, merged, split, and multiply split/merged jets for the legacy seed and midpoint + seed algorithms. In each case only small variations are observed between the two algorithms, indicating that a legacy cone algorithm augmented by midpoints is an acceptable choice for comparisons to Run I physics results. In fact, Figs. 15 and 16 represent extreme deviations in jet E_T, since E_T differences are expected to be reduced after application of jet energy corrections appropriate to each algorithm.

3.5. Proposals for Common Run II Cone Jet Algorithms

The cone algorithm starts with a cone defined in E-scheme variables as

$$i \subset C \quad : \quad \sqrt{\left(y^i - y^C\right)^2 + \left(\phi^i - \phi^C\right)^2} \leq R. \tag{26}$$

where for massless towers, particles, or partons $y^i = \eta^i$. The E-scheme centroid corresponding to this cone is given by

$$p^C = (E^C, \mathbf{p}^C) = \sum_{i \subset C} (E^i, p^i_x, p^i_y, p^i_z) \, , \tag{27}$$

$$\bar{y}^C = \frac{1}{2} \ln \frac{E^C + p^C_z}{E^C - p^C_z} \, , \quad \bar{\phi}^C = \tan^{-1} \frac{p^C_y}{p^C_x} \, . \tag{28}$$

A jet arises from a “stable” cone, for which $\bar{y}^C = y^C = y^J$ and $\bar{\phi}^C = \phi^C = \phi^J$, and the jet has kinematic properties

$$p^J = (E^J, \mathbf{p}^J) = \sum_{i \subset J = C} (E^i, p^i_x, p^i_y, p^i_z) \, , \tag{29}$$

$$p^J_T = \sqrt{(p^J_x)^2 + (p^J_y)^2} \, , \tag{30}$$

$$y^J = \frac{1}{2} \ln \frac{E^J + p^J_z}{E^J - p^J_z} \, , \quad \phi^J = \tan^{-1} \frac{p^J_y}{p^J_x} \, . \tag{31}$$

Seedless algorithm. For a seedless algorithm we recommend the streamlined jet algorithm defined in Section 3.3.1 that includes the flow cut for computational efficiency improvement and reduction of soft proto-jet construction. The clustering or jet finding should be done in terms of E-scheme variables.

Seed–based algorithm or ILCA. Backwards compatibility is important here as well as common specifications between experiments. For the Run II algorithm we recommend that jet clustering commence on each seed tower (rather than consolidated seeds as in Run I), for simplicity of the algorithm and to reduce dependencies on detector segmentation. Since the finding

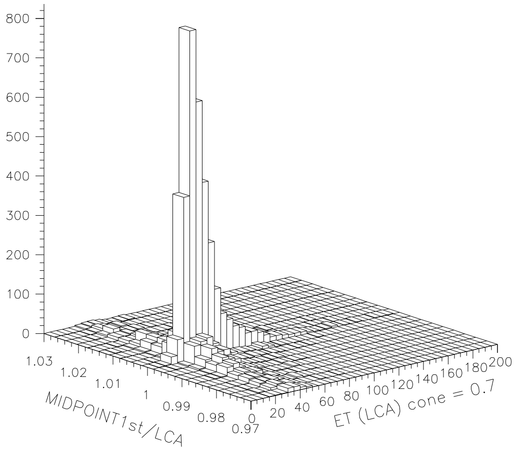

Figure 16. E_T ratios for leading jets. The ratio of leading jet E_T in the midpoint algorithm is plotted as a function of the legacy cone jet's E_T.

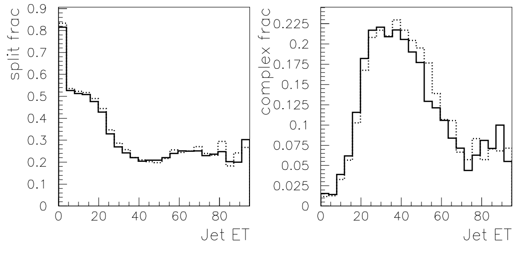

Figure 17. A view of splitting and merging fractions in the legacy seed (solid) and midpoint + seed algorithms (dotted).

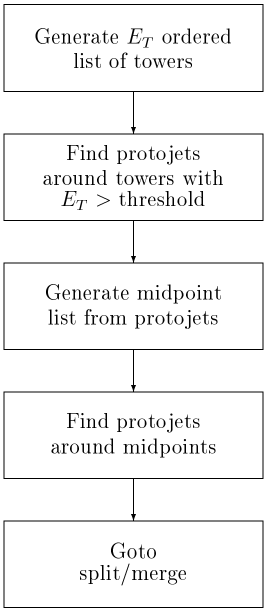
Generate E_T ordered list of towers
Find protojets around towers with $E_T >$ threshold
Generate midpoint list from protojets
Find protojets around midpoints
Goto split/merge

Figure 18. Method for addition of midpoints.

of proto-jets is determined by the seed threshold, it is reasonable to determine the midpoints based on the positions of the proto-jets rather than the seed list itself, as illustrated in Fig. 18. This would reduce the number of midpoints to be calculated due to the large combinatorics caused by adjacent seed towers within jet cones.

Specifications Summary We list here the precise specifications of the jet algorithms and variables:

1. R_{cone}: 0.7
2. p_T^{seed}: 1.0 GeV
3. Recombination: E-scheme
4. Midpoints: Added after cone clustering
5. Split/Merge: p_T ordered, threshold = 50% of lower p_T jet
6. Reported kinematic variables: E-scheme, either directly as $(E^J, \mathbf{p}^J)$ or as $(m^J, {p_T}^J, y^J, \phi^J)$, where m^J is the mass of the jet $(m^J = \sqrt{{E^J}^2 - {\mathbf{p}^J}^2})$.

4. K_T Jet Algorithms

4.1. Introduction

This section provides a guide for the definition of K_T jet algorithms for the Tevatron. Section 4.2 describes the recommended algorithm in detail. Section 4.3 discusses preclustering of particles, cells, or towers for both the CDF and DØ experiments. Sections 4.4 and 4.5 outline momentum calibration of the K_T algorithm and briefly describe jet resolution. Finally, in Section 4.6, we provide a few examples of the versatility of the K_T algorithm.

4.2. The Run II K_T Algorithm

In this section we propose a standard K_T jet algorithm for Run II at the Fermilab Tevatron. This proposal, based on studies of the K_T algorithm by several groups [20, 21, 22], establishes a common algorithm that satisfies the general criteria presented in Section 1.

The K_T jet algorithm starts with a list of *preclusters* which are formed from calorimeter cells, particles, or partons.[5] Initially, each precluster is assigned a vector

$$(E, \mathbf{p}) = E\,(1,\, \cos\phi\sin\theta,\, \sin\phi\sin\theta,\, \cos\theta) \tag{32}$$

where E is the energy associated with the precluster, ϕ is the azimuthal angle, and θ is the polar angle with respect to the beam axis. For each precluster, we calculate the square of the transverse momentum, p_T^2, using

$$p_T^2 = p_x^2 + p_y^2 \tag{33}$$

and the rapidity, y, using[6]

$$y = \frac{1}{2}\ln\frac{E + p_z}{E - p_z}. \tag{34}$$

A flowchart of the K_T algorithm is shown in Fig. 19. Starting with a list of preclusters and an empty list of jets, the steps of the algorithm are as follows:

1. For each precluster i in the list, define

   $$d_i = p_{T,i}^2\,. \tag{35}$$

   For each pair (i, j) of preclusters $(i \neq j)$, define

   $$\begin{aligned} d_{ij} &= \min\left(p_{T,i}^2, p_{T,j}^2\right)\frac{\Delta\mathcal{R}_{ij}^2}{D^2} \\ &= \min\left(p_{T,i}^2, p_{T,j}^2\right)\frac{(y_i - y_j)^2 + (\phi_i - \phi_j)^2}{D^2} \end{aligned} \tag{36}$$

   where $D \approx 1$ is a parameter of the jet algorithm. For $D = 1$ and $\Delta\mathcal{R}_{ij} \ll 1$, d_{ij} is the minimal relative transverse momentum $k_\perp$ (squared) of one vector with respect to the other.

[5] Preclustering is discussed in detail in Section 4.3.

[6] To avoid differences in the behavior of the algorithm due to computational precision when $|y|$ is large, we assign $y = \pm 10$ if $|y| > 10$.

2. Find the minimum of all the d_i and d_{ij} and label it d_{min}.

3. If d_{min} is a d_{ij}, remove preclusters i and j from the list and replace them with a new, merged precluster $(E_{ij}, \mathbf{p}_{ij})$ given by

$$E_{ij} = E_i + E_j \ , \tag{37}$$

$$\mathbf{p}_{ij} = \mathbf{p}_i + \mathbf{p}_j \ . \tag{38}$$

4. If d_{min} is a d_i, the corresponding precluster i is "not mergable." Remove it from the list of preclusters and add it to the list of jets.

5. If any preclusters remain, go to step 1.

The algorithm produces a list of jets, each separated by $\Delta\mathcal{R} > D$. Fig. 20 illustrates how the K_T algorithm successively merges the preclusters in a simplified diagram of a hadron collision.

The K_T algorithm presented above is based on several slightly different K_T jet clustering algorithms for hadron colliders [20, 21, 22]. The main differences have to do with (1) the recombination scheme and (2) the method of terminating the clustering. The choices in the proposal above are discussed in the following paragraphs.

The recombination scheme was investigated by Catani *et al.* [20]. We elect to use the covariant E-scheme (Eqs. 37–38), which corresponds to vector addition of four-momenta, because our goals are

1. conceptual simplicity,

2. correspondence to the scheme used in the K_T algorithm for e^+e^- collisions [23],

3. absence of an energy defect [24], and

4. optimum suitability for the calibration method described in Section 4.4. [25]

The prescription of Catani, *et al.* [20, 21] introduces a stopping parameter, d_{cut}, that defines the hard scale of the physics process and separates the event into a hard scattering part and a low-p_T part ("beam jets"). Catani *et al.* suggest two ways to use the d_{cut} parameter. First, d_{cut} can be set to a constant value *a priori*, and when $d_{min} > d_{cut}$ the algorithm stops. At this point, all previously identified jets with $p_T^2 < d_{cut}$ are classified as beam jets, and all remaining preclusters with $p_{T,i}^2 > d_{cut}$ are retained as hard final-state jets. Alternatively, an effective d_{cut} can be identified on an event-by-event basis so that clustering continues until a given number of final-state jets are reconstructed.

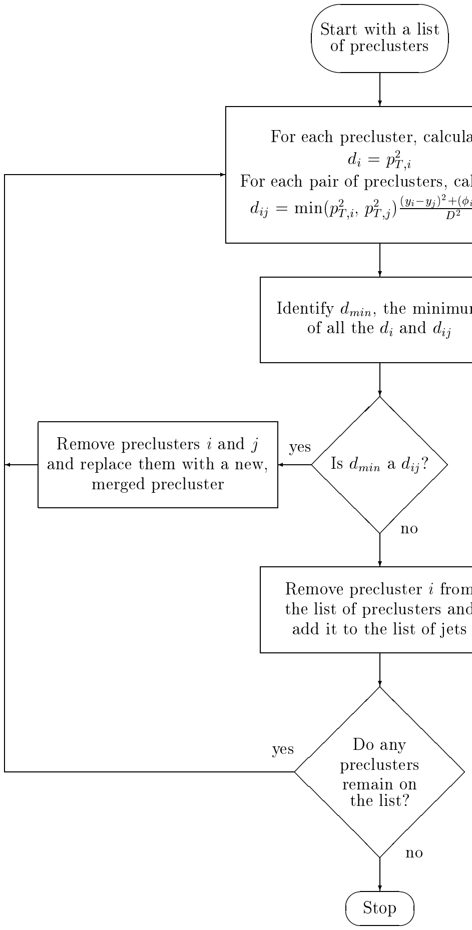
Start with a list of preclusters
For each precluster, calculate
$d_i = p_{T,i}^2$
For each pair of preclusters, calculate
$d_{ij} = \min(p_{T,i}^2, p_{T,j}^2)\frac{(y_i-y_j)^2+(\phi_i-\phi_j)^2}{D^2}$
Identify d_{min}, the minimum of all the d_i and d_{ij}
Is d_{min} a d_{ij}?
yes
Remove preclusters i and j and replace them with a new, merged precluster
no
Remove precluster i from the list of preclusters and add it to the list of jets
Do any preclusters remain on the list?
yes
no
Stop

Figure 19. The K_T jet algorithm.

Unlike Catani, *et al.*, the algorithm proposed by Ellis and Soper [22] continues to merge preclusters until all jets are separated by $\Delta\mathcal{R} > D$. We have adopted this choice. Besides its simplicity, this method maintains a similarity with cone algorithms in hadron collisions. Whereas the use of d_{cut} is well suited for defining an *exclusive* jet cross section (typical of e^+e^- collisions), we desire an algorithm that defines *inclusive* jet cross sections in terms of a single angular resolution parameter D, which is similar to R for cone algorithms.

4.3. Preclustering

As described in the previous section, the input to the K_T jet algorithm is a list of vectors, or preclusters. Ideally, one should be able to apply the K_T algorithm equally at the parton, particle, and detector levels, with no dependence on detector cell type, number of cells, or size. The goal of *preclustering* is to strive for order independence and detector independence by employing well-defined procedures to remove (or reduce) the detector-dependent aspects of

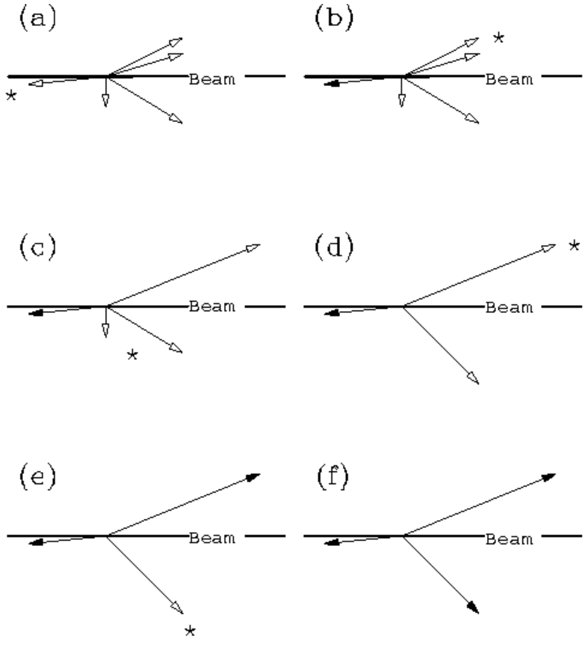

Figure 20. A simplified example of the final state of a hadron collision. The open arrows represent preclusters in the event, and the solid arrows represent the final jets reconstructed by the K_T algorithm. The six diagrams show successive iterations of the algorithm. In each diagram, either a jet is defined (when it is well separated from all other preclusters), or two preclusters are merged (when they have small relative $k_\perp$). The asterisk labels the relevant precluster(s) at each step.

jet clustering. Practically, however, this independence is very difficult to achieve. For example, if a single particle strikes the boundary between two calorimeter towers, two clusters of energy may be measured. Conversely, two collinear particles may shower in a single calorimeter tower so that only one vector is measured experimentally. Preclustering all vectors within a radius larger than the calorimeter tower size removes this problem.

At the parton and particle levels, the simplest possible preclustering scheme is to identify each parton or particle four-vector as a precluster. Experimentally, differences between the geometries of the CDF and DØ calorimeters necessitate different preclustering schemes. In particular, the DØ discussion describes how the preclustering scheme can be used to control the number of preclusters passed to the K_T algorithm in order to keep the jet analysis computationally feasible. It can also be used to ensure that the preclusters all exhibit positive energy. Candidate schemes to achieve these goals are described in detail in the following sections. However, it is important that the preclustering scheme does not introduce the sort of problems with infrared or collinear sensitivities that we earlier discussed for the case of seeds.

4.3.1. CDF Preclustering

The CDF calorimeter system for Run II [26] consists of 1,536 towers. Each tower consists of an electromagnetic (EM) component and a hadronic (HAD) component. In order to form preclusters for input to the K_T algorithm, we propose the following:

1. Measure the amount of EM energy deposited into each calorimeter tower, E_{EM}, and form the vector $(E_{EM}, \mathbf{p}_{EM})$ where

$$p_{x,EM} = E_{EM} \cos\phi \sin\theta_{EM} \ , \tag{39}$$
$$p_{y,EM} = E_{EM} \sin\phi \sin\theta_{EM} \ , \tag{40}$$
$$p_{z,EM} = E_{EM} \cos\theta_{EM} \ . \tag{41}$$

Likewise, measure the amount of HAD energy deposited into each calorimeter tower, E_{HAD}, and form the vector $(E_{HAD}, \mathbf{p}_{HAD})$ where

$$p_{x,HAD} = E_{HAD} \cos\phi \sin\theta_{HAD} \ , \tag{42}$$
$$p_{y,HAD} = E_{HAD} \sin\phi \sin\theta_{HAD} \ , \tag{43}$$
$$p_{z,HAD} = E_{HAD} \cos\theta_{HAD} \ . \tag{44}$$

The angles θ_{EM}, θ_{HAD} and ϕ specify the position of the calorimeter tower components with respect to the interaction point. Note that θ_{EM} and θ_{HAD} may take on slightly different values when calculated using different interaction points along the beam axis (see Fig. 21).

2. For each calorimeter tower, calculate a vector $(E, \mathbf{p})$ by summing the vectors for the EM and HAD components:

$$(E, \mathbf{p}) = (E_{EM} + E_{HAD}, \, \mathbf{p}_{EM} + \mathbf{p}_{HAD}) \tag{45}$$

3. For each calorimeter tower, calculate the p_T from its associated vector using

$$\begin{aligned} p_T &= \sqrt{p_x^2 + p_y^2} \\ &= E_{EM} \sin\theta_{EM} + E_{HAD} \sin\theta_{HAD} \ . \end{aligned} \tag{46}$$

4. Assemble a list of tower vectors for which

$$p_T > p_T^{min}, \tag{47}$$

where $p_T^{min} \approx 100$ MeV.[7] These are the preclusters for the K_T algorithm.

In designing the CDF preclustering scheme, the primary goal was simplicity. We made every attempt to maintain a close relationship between the physical calorimeter towers and the input preclusters for the K_T algorithm.

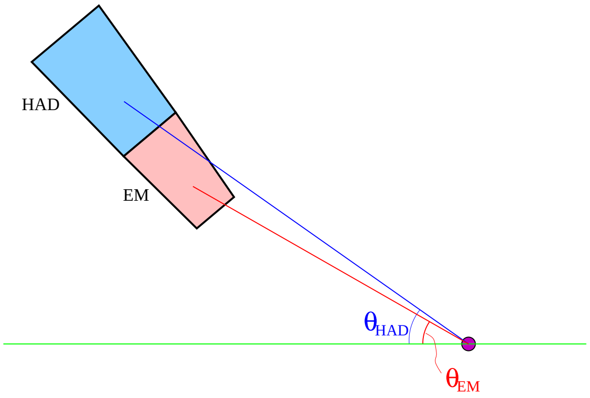

Figure 21. Schematic of a single CDF calorimeter tower.

4.3.2. DØ Preclustering

The K_T jet algorithm is an $\mathcal{O}(n^3)$ algorithm, where n is the number of vectors in the event [20]. Limited computer processing time does not allow this algorithm to run on the ∼45000 cells or even the ∼6000 towers of the DØ calorimeter. Therefore, we employ a preclustering algorithm to reduce the number of vectors input to the algorithm. Essentially, towers are merged if they are close together in $\eta \times \phi$ space, or if they have small p_T (or negative p_T, as explained below). The preclustering algorithm described below was used by the DØ experiment in Run I. We examine the effects of the Run I preclustering algorithm, and discuss possible alternatives for Run II. Although the effects of preclustering on jet observables should be small, this is analysis and detector dependent. A Monte Carlo study of preclustering effects on the jet p_T and on jet structure is also presented.

In Run I, one use of preclustering was to account for negative energy calorimeter towers [27] which can cause difficulties for the K_T algorithm. In the DØ calorimeter, we measured the difference in voltage between two readings (peak minus base), as illustrated in Fig. 22. To first order, this online baseline subtraction technique removes the effect of luminosity-dependent noise in the calorimeter, on a tower-by-tower basis. Residual fluctuations in each reading, however, sometimes lead to measured energies which are negative. One can imagine at least four ways to deal with these negative energy towers.

1. Absorb the negative energy into a precluster of towers such that the overall precluster energy is positive, as will be discussed here.

2. Add an offset to all tower energies so that there are none with negative energy. The offset could then be removed later in the analysis.

3. Ignore all towers with negative energy, *i.e.*, remove them from the jet analysis.

4. Proceed with the K_T algorithm analysis including the negative energy towers, assuming that their impact is negligible. Recall that in the cone algorithm case the negative energy towers are the source of the observed limit cycles for quasi-stable cones, which does not seem to be a serious problem.

Clearly, further studies of this issue are required. The precluster scheme can also be used to absorb low p_T towers similarly to what is done for negative energy towers.

The Run I preclustering algorithm, which is employed in the following studies, has six steps:

1. Identify each calorimeter cell with a 4-vector $(E, \mathbf{p}) = E\,(1, \cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta)$ where E is the measured energy in the cell. For each cell, define

$$p_T = \sqrt{p_x^2 + p_y^2} = E\sin\theta \qquad (48)$$

and

$$\eta = -\ln\left(\tan\frac{\theta}{2}\right) \ . \qquad (49)$$

2. Remove any calorimeter cells with $p_T < -500$ MeV. Cells with slightly negative p_T are allowed due to pileup effects in the calorimeter, but cells with highly negative p_T are very rarely observed in minimum bias events and are thus considered spurious, so they are removed.

[7] This p_T cut is designed to retain towers with energies well above the level of electronic noise. The exact value for this p_T cut will depend on measurements of calorimeter performance.

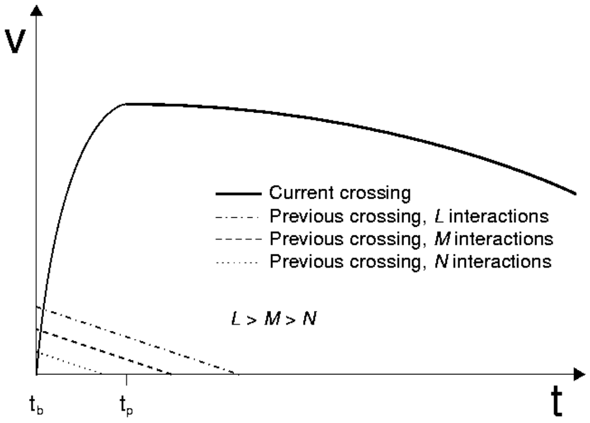
V
Current crossing
Previous crossing, L interactions
Previous crossing, M interactions
Previous crossing, N interactions
$L > M > N$
t_b
t_p
t

Figure 22. Schematic of voltage in a calorimeter cell as a function of time. The solid line shows the contribution for a given event (the current crossing). The cell is sampled once at t_b, just before a $p\bar{p}$ bunch crossing, to establish a base voltage. The voltage rises during the time it takes electrons to drift in the liquid argon gap (~500 ns), and reaches a peak value at $t_p \approx 2\,\mu$s, which is set by pulse-shaping amplifiers in the signal path. The cell is sampled again at t_p, and the voltage difference $\Delta V = V(t_p) - V(t_b)$ is proportional to the raw energy in the cell. Because the decay time of the signal $\tau \approx 30\,\mu$s is much larger than the accelerator bunch crossing time $t_x = 3.5\,\mu$s, $V(t_b)$ may have a contribution from a previous bunch crossing. The size of this contribution is related to the number of $p\bar{p}$ interactions in the previous crossing, which depends on the beam luminosity. The dashed lines show an example contribution from a previous bunch crossing containing three different numbers of $p\bar{p}$ interactions. The figure is not drawn to scale.

3. For each calorimeter tower, sum the transverse momenta of cells within that tower:

$$p_T^{tower} = \sum_{cell\,\in\,tower} p_T^{cell}\ . \tag{50}$$

4. Merge towers if they are close together in $\eta \times \phi$ space:

    (a) Form an η-ordered (from most negative to most positive) list of towers; towers with equal η are ordered from $\phi = 0$ to $\phi = 2\pi$.

    (b) Remove the first tower in the list and call it a precluster.

    (c) From the remainder of the list, find the closest tower to the precluster.

    (d) If they are within $\Delta\mathcal{R}_p = \sqrt{\Delta\eta^2 + \Delta\phi^2} = 0.2$, remove the closest tower from the list, and combine it with the existing precluster, forming a new precluster; go to 4c.

    (e) If any towers remain, go to 4b.

5. Preclusters which have negative transverse momentum $p_T = p_{T-} < 0$ are redistributed to neighboring preclusters. Given a negative p_T precluster with (p_{T-}, η_-, ϕ_-), we define a search square of size $(\eta_- \pm 0.1) \times (\phi_- \pm 0.1)$. If the vector sum of positive p_T in the search square is greater than $|p_{T-}|$, then p_{T-} is redistributed to the positive p_T preclusters in the search square. Otherwise, the search square is increased in steps of $\Delta\eta = \pm 0.1$ and $\Delta\phi = \pm 0.1$, and redistribution is again attempted. If redistribution still fails with a search square of size $(\eta_- \pm 0.7) \times (\phi_- \pm 0.7)$, the p_T of the negative momentum precluster is set to zero.

6. Preclusters which have $p_T < p_T^p = 200$ MeV are redistributed to neighboring preclusters, as in step 5. We make the additional requirement that the search square have at least three positive p_T preclusters, to reduce the overall number of preclusters. The threshold p_T^p was tuned to produce $\sim$200 preclusters/event, as shown in Fig. 23, to fit processing time constraints. Next, jets are reconstructed from the preclusters.

In steps 4–6, the combination followed a Snowmass style prescription:

$$p_T = p_{T,i} + p_{T,j}\ , \tag{51}$$

$$\eta = \frac{p_{T,i}\eta_i + p_{T,j}\eta_j}{p_{T,i} + p_{T,j}}\ , \tag{52}$$

$$\phi = \frac{p_{T,i}\phi_i + p_{T,j}\phi_j}{p_{T,i} + p_{T,j}}\ . \tag{53}$$

As a minimal change to the Run I preclustering algorithm, a possible Run II preclustering proposal should instead use vector addition of four-momenta. The Run II preclustering algorithm should also use y (as defined by Eq. 34) instead of η and a true 2-vector p_T rather than the scalar p_T of Eq. 51. Generally, the definitions of variables and recombination scheme in the preclustering algorithm should match the choices used in the proposed K_T jet algorithm. All of the results presented here used the Run I preclustering algorithm.

The preclustering radius $\Delta\mathcal{R}_p$ in step 4 of the algorithm above can be used to test the sensitivity of jets to the calorimeter segmentation size, $\Delta\phi \times \Delta\eta = 0.1 \times 0.1$ (or smaller) in the DØ calorimeter. Preclustering with $\Delta\mathcal{R}_p = 0.2 > \Delta\eta$ or $\Delta\phi$ in step 4 of the algorithm mimics a coarser calorimeter. This effect was studied

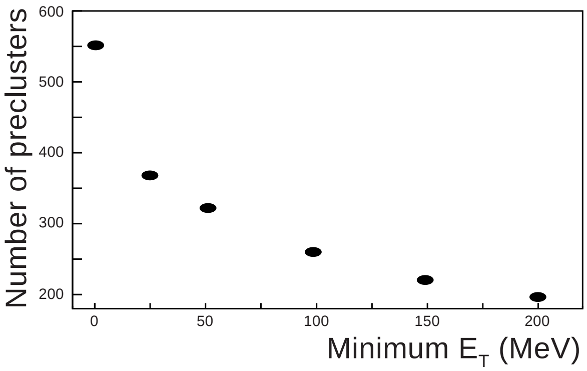

Figure 23. The number of preclusters per event, as a function of minimum precluster transverse energy E_T^p. The DØ data were preclustered with the choice $E_T^p = 200$ MeV, which produced ~200 preclusters per event. With the preclusters treated as massless, E_T is the same as p_T. This identification is certainly appropriate for individual calorimeter towers.

in a sample of HERWIG Monte Carlo QCD jet events. The jets in the hard 2 → 2 scattering were generated with $p_T > 50$ GeV, and at least one of the two leading order partons was required to be central ($|\eta| < 0.9$). The events were passed through a full simulation (including luminosity $\mathcal{L} \approx 5 \times 10^{30}\mathrm{cm}^{-2}\mathrm{s}^{-1}$) of the DØ detector. The MC sample is described in more detail in Section 4.4.1. Fig. 24 shows the number of preclusters with $\Delta\mathcal{R}_p = 0.2$ is ~180, reduced by 37% from that obtained with $\Delta\mathcal{R}_p = 0$. Fig. 25 shows that preclustering is necessary even at the particle level in the Monte Carlo, reducing the number preclusters by 24%. Comparing Figs. 24 and 25, the number of preclusters in the detailed detector simulation is a factor 2.4 higher than at the particle level for $\Delta\mathcal{R}_p = 0$. Most of the additional preclusters are reconstructed near the beampipe and some are due to localized deposits of low energy. With $\Delta\mathcal{R}_p = 0.2$, the number of preclusters increases only by a factor 2.0.

The effect of the preclustering radius $\Delta\mathcal{R}_p$ on jets and jet structure was examined next. Fig. 26 shows the comparison of the leading jet p_T with $\Delta\mathcal{R}_p = 0.2$ to that with $\Delta\mathcal{R}_p = 0$. The jets were reconstructed with the K_T jet algorithm $D = 0.5$. The preclustering radius $\Delta\mathcal{R}_p = 0.2$ (step 4 of the preclustering algorithm) reduces the mean jet p_T by 0.7 GeV. Evidently, the preclustering algorithm assigns energy differently than the K_T algorithm. It is difficult to track exactly which towers end up in each jet, in part because of the redistribution of energy in steps 5 and 6 of the preclustering algorithm. The net effect is that some energy belonging to the leading jet when $\Delta\mathcal{R}_p = 0$ is transferred to non-leading jets when $\Delta\mathcal{R}_p = 0.2$. The shift in the leading jet p_T spectrum is visible in the top panel of Fig. 26, and the ratio in the bottom panel suggests some dependence on the jet p_T. Such a shift may need to be corrected for in the Run II experimental data, but will be different due to the change in calorimeter electronics. In Run I, a correction was not explicitly applied to the experimental data for this effect. Instead, the theoretical predictions included the identical preclustering algorithm used in experimental data. Fortunately, the particle-level result for leading jet p_T is not as sensitive to $\Delta\mathcal{R}_p$. This is shown in Fig. 27. Note that even the particles in the Monte Carlo were projected into a calorimeter-like grid ($\Delta\phi \times \Delta\eta = 0.1 \times 0.1$) by the preclustering algorithm. If this were not the case, then we would expect an even larger effect than illustrated in Fig. 27.

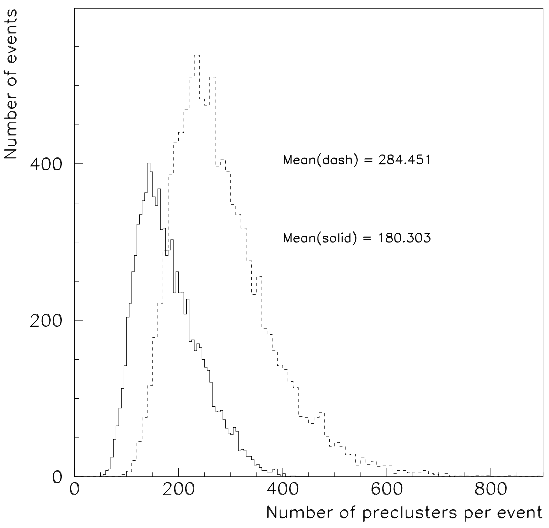

Figure 24. Distribution of the number of preclusters per event, with $\Delta\mathcal{R}_p = 0.2$ (solid), and with $\Delta\mathcal{R}_p = 0$ (dash). Taken from a sample of QCD jet events from MC data. The jets were reconstructed using the calorimeter simulation, including the luminosity simulation. The preclustering radius $\Delta\mathcal{R}_p = 0.2$ reduces the mean number of preclusters per event by 37%.

The jet structure, however, is more sensitive to the preclustering radius $\Delta\mathcal{R}_p$. Fig. 28 shows the average subjet multiplicity, as a function of y_{cut} (see Section 4.6.1), in particle-level jets. There are more subjets in jets when $\Delta\mathcal{R}_p = 0$, compared to when $\Delta\mathcal{R}_p = 0.2$. Requiring preclusters to be separated by $\Delta\mathcal{R}_p$ affects

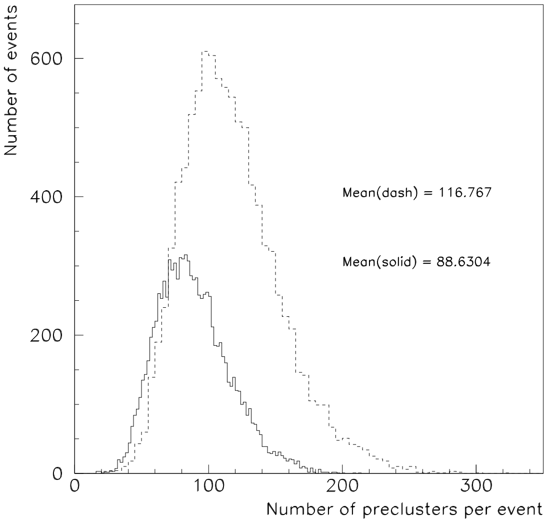

Figure 25. Same as in Fig. 24, except the jets were reconstructed in MC data at the particle level, with no calorimeter or luminosity simulation. The same preclustering radius $\Delta\mathcal{R}_p = 0.2$ reduces the mean number of preclusters per event by 24%
.

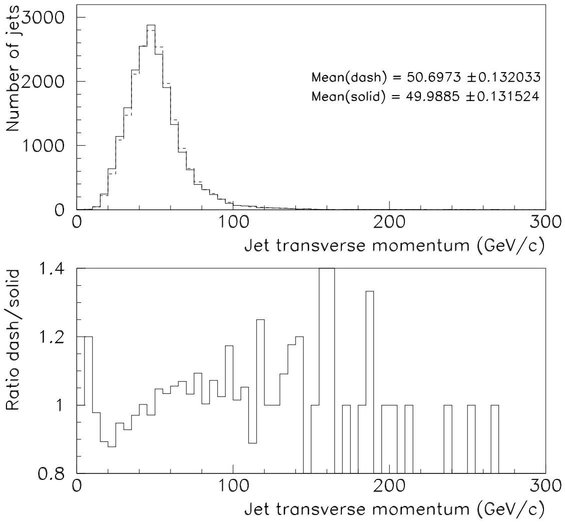

Figure 26. The top panel shows the distribution of the leading jet p_T with $\Delta\mathcal{R}_p = 0.2$ (solid), and with $\Delta\mathcal{R}_p = 0$ (dash). Measured in a sample of QCD jet events from MC data. The sample was generated with minimum parton transverse momentum $p_T^{min} =$ 50 GeV. The K_T jets were reconstructed with $D = 0.5$ in the calorimeter simulation, including the luminosity simulation. The preclustering radius $\Delta\mathcal{R}_p = 0.2$ reduces the mean of the leading jet p_T by 0.7 GeV. The bottom panel shows the ratio of the histograms in the top panel.

the subjet structure below

$$\begin{aligned} y_{cut} &< \left(\frac{\Delta\mathcal{R}_p}{2D}\right)^2 \\ &< 10^{-1.4}. \end{aligned} \tag{54}$$

Again, the subjet multiplicity is increased even further when particles in the Monte Carlo are not projected into a calorimeter-like grid ($\Delta\phi \times \Delta\eta = 0.1 \times 0.1$). This underscores the fact that the same preclustering algorithm, as well as the same jet algorithm, must be used in any comparisons of theoretical predictions to experimental data which are sensitive to internal jet structure at the level of the detector granularity.

4.4. Momentum Calibration of K_T Jets at DØ

Jet production is the dominant process in $p\overline{p}$ collisions at $\sqrt{s} = 1.8$ TeV, and almost every physics measurement at the Tevatron involves events with jets. A precise calibration of measured jet momentum and energy, therefore, is of fundamental importance. Although the use of a K_T algorithm is well defined theoretically, questions have recently arisen regarding the performance of the algorithm in a high luminosity hadron collider environment.

The DØ Collaboration developed a method to calibrate K_T jets to a high level of accuracy. The details are discussed thoroughly in Ref. [28, 29]. Here, we briefly summarize this work by the DØ Collaboration to illustrate instrumentation effects on the K_T algorithm, as well as its behavior in a high luminosity hadron collider. The K_T jets momentum scale correction is largely based on the calibration of cone jets, extensively discussed in a recent article [27]. The derivation of the momentum scale correction is performed for K_T jets with $D = 1$. The measured jet momentum, p_{jet}^{meas}, is corrected to that of the final-state particle-level jet, p_{jet}^{ptcl}, using the following relation:

$$p_{jet}^{ptcl} = \frac{p_{jet}^{meas} - p_O}{R_{jet}}\,, \tag{55}$$

where p_O denotes a momentum offset correction for underlying event, uranium noise, pile-up, and additional $p\overline{p}$ interactions. R_{jet} is the calorimeter momentum response to jets. Note that the equation is missing the out-of-cone showering loss factor. In cone jets, this fac-

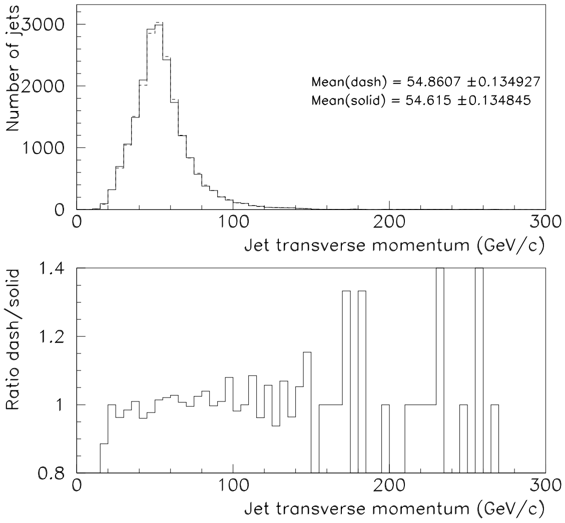

Figure 27. Same as in Fig. 26, except the jets were reconstructed in MC data at the particle level, with no calorimeter or luminosity simulation. The same preclustering radius $\Delta\mathcal{R}_p = 0.2$ reduces the mean of the leading jet p_T by 0.25 GeV. The bottom panel shows the ratio of the histograms in the top panel.

tor corrects for the fraction of the energy of the final-state hadrons which is lost outside the cone boundaries due to calorimeter showering. This is an instrumentation effect completely unrelated to parton showering losses outside the cone. There is no correction for the latter. Note that the important issue here is not so much that p_O be small or that R_{jet} be near unity, but rather that these parameters can be determined with precision. This is the question to be addressed when comparing jet algorithms.

The DØ uranium-liquid argon sampling calorimeters [30] are shown in Fig. 29–30. They constitute the primary system used to identify e, γ, jets and missing transverse energy ($\vec{\not\!\!E}_T$). $\vec{\not\!\!E}_T$ is defined as the negative of the vector sum of the calorimeter cell transverse energies (E_T's). The Central (CC) and End (EC) Calorimeters contain approximately seven and nine interaction lengths of material respectively, ensuring containment of nearly all particles except high p_T muons and neutrinos. The intercryostat region (IC), between the CC and the EC calorimeters, is covered by a scintillator based intercryostat detector (ICD) and massless gaps (MG) [30]. The segmentation is $\Delta\phi \times \Delta\eta = 0.1 \times 0.1$ (or smaller).

The fractional energy resolution, σ_E/E, characterizes the suitability of the DØ calorimeter system for *in-situ* momentum calibration techniques. It is parameterized with a $\sqrt{S^2/E + C^2}$ functional form. For electrons, the sampling term, S, is 14.8 (15.7)% in the CC (EC), and the constant term, C, is 0.3% in both the CC and EC. For pions, the sampling term is 47.0 (44.6)%, and the constant term is 4.5 (3.9)% in the CC (EC). The energy response is linear to within 0.5% for electrons above 10 GeV and for pions above 20 GeV. The DØ calorimeters are nearly compensating, with an $\frac{e}{\pi}$ ratio less than 1.05 above 30 GeV. Due to the hermiticity and linearity of the DØ calorimeters their response function is well described by a Gaussian distribution. These properties indicate that the DØ calorimeter system is well suited for jet and $\not\!\!E_T$ measurements and are the basis of the *in-situ* calibration method described here.

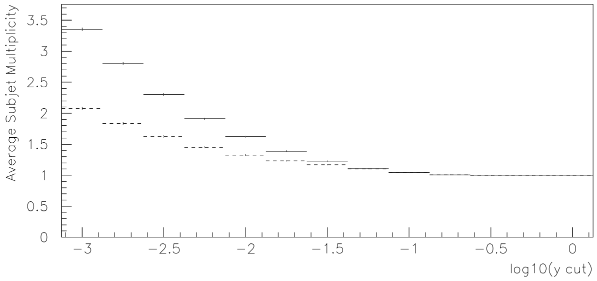

Figure 28. The average subjet multiplicity, as a function of y_{cut}, in a sample of jets reconstructed in MC data at the particle level, with no calorimeter or luminosity simulation. The solid curve shows the results with $\Delta\mathcal{R}_p = 0$, and the dashed curve shows the results with $\Delta\mathcal{R}_p = 0.2$. The preclustering radius $\Delta\mathcal{R}_p = 0.2$ reduces the average subjet multiplicity for $y_{cut} < 10^{-1.4}$.

4.4.1. Offset Correction

The total offset correction is measured in transverse momentum and expressed as $p_{T,O} = O_{ue} + O_{zb}$. The first term is the contribution of the underlying event (energy associated with the spectator partons in a high p_T event). The second term accounts for uranium noise, pile-up and energy from additional $p\overline{p}$ interactions in the same crossing. Pile-up is the residual energy from previous $p\overline{p}$ crossings as a result of the long shaping times associated with the preamplification stage in calorimeter readout cells.

To simulate the contribution of O_{zb} to jets, DØ Run I collider data taken in a random $p\overline{p}$ crossing (no trigger requirements) was overlayed on high p_T jet HERWIG [31] Monte Carlo events. Jets were matched in

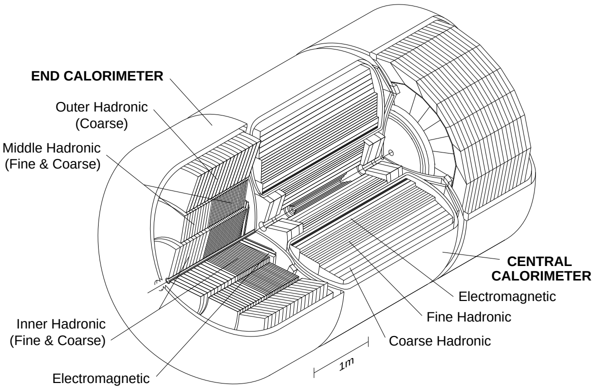

Figure 29. The DØ liquid argon calorimeter is divided physically into three cryostats, defining the central calorimeter and two end calorimeters. Plates of absorber material are immersed in the liquid argon contained by the cryostats. Each cryostat is divided into an electromagnetic, fine hadronic, and coarse hadronic section.

this sample to jets in the sample with no overlay. The contribution of uranium noise, pile-up, and multiple interactions was determined by taking the difference in p_T between matched pairs. O_{ue} was extracted in the same way from the overlap of low luminosity minimum bias data (a crossing with an inelastic collision) on Monte Carlo events. O_{ue} and O_{zb} for jets with $p_T = 30 - 50$ GeV are shown in Figs. 31 and 32. The offset is derived in the central calorimeter and extrapolated to higher η regions.

4.4.2. Response: The Missing E_T Projection Fraction Method

DØ makes a direct measurement of the jet momentum response using conservation of p_T in Run I photon-jet (γ-jet) collider events [27]. Previously, the photon energy/momentum scale was determined from the DØ $Z \to e^+e^-$, J/ψ and π° data samples, using the masses of these known resonances. In the case of a γ-jet two body process, the jet momentum response can be measured as:

$$R_{jet} = 1 + \frac{\vec{\not{E}}_T \cdot \hat{n}_{T\gamma}}{p_{T\gamma}}\,, \tag{56}$$

where $p_{T\gamma}$ and $\hat{n}$ are the transverse momentum and direction of the photon. To avoid resolution and trigger biases, R_{jet} is binned in terms of $E' = p_{T\gamma}^{meas} \cdot \cosh(\eta_{jet})$ and then mapped onto p_{jet}^{meas}. E' depends only on photon variables and jet pseudorapidity, which are quantities measured with very good precision. R_{jet} and E' depend only on the jet position, which has little dependence on the type of jet algorithm employed.

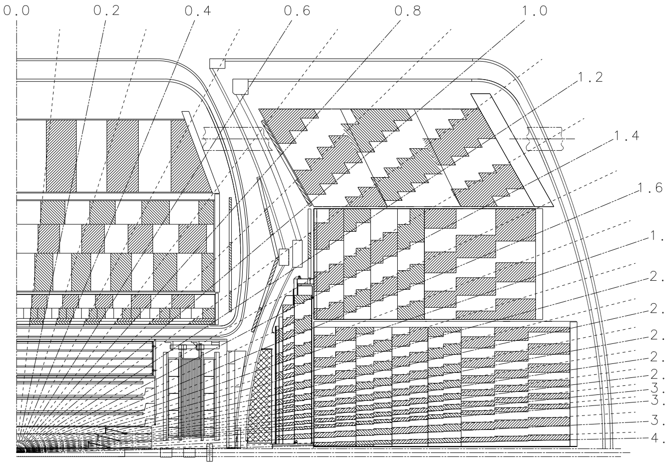

Figure 30. One quadrant of the DØ calorimeter and drift chamber, projected in the $x-z$ plane. Radial lines illustrate the detector pseudorapidity and the pseudoprojective geometry of the calorimeter towers. Each tower is of size $\Delta\eta \times \Delta\phi = 0.1 \times 0.1$.

R_{jet} as a function of p_{jet}^{meas} (p_{Kt}) is shown in Fig. 33. The data is fit with the functional form $R_{jet}(P) = a + b \cdot \ln(P) + c \cdot \ln(P)^2$. R_{jet} for cone ($R = 0.7$) [27] and K_T ($D = 1$) jets are different by about 0.05. This difference does not have any physical meaning; it arises from different voltage-to-energy conversion factors at the cell level before reconstruction.

4.4.3. Tests of the Method

The accuracy of the K_T jet momentum scale correction was verified using a HERWIG γ-jet sample and a fast version (SHOWERLIB) [32] of the Run I detector simulation using GEANT [33]. A Monte Carlo jet momentum scale was derived and the corrected jet momentum compared directly to the momentum of the associated particle jet. Figure 34 shows the ratio of calorimeter and particle jet momentum before and after the jet scale correction in the CC. The vertical bars are statistical errors. Systematic errors (not shown) are of the order of 0.01–0.02. After the jet correction is applied, the ratio versus particle jet p_T is consistent with unity within the total uncertainty.

4.4.4. Summary

DØ improved the method introduced in Ref. [27] for estimating the effects of underlying event, uranium noise, pile-up, and additional $p\overline{p}$ interactions. The offset correction is larger for K_T jets with $D = 1$ than for cone jets with $R = 0.7$ by approximately 20–30%. The uncertainty (∼0.1 GeV for underlying event, and ∼0.2 GeV for the second offset term in the CC), however, is slightly smaller. A K_T ($D = 1$) algorithm reconstructs more energy from uranium noise, pile-up, underlying event, and multiple $p\overline{p}$ interactions than a

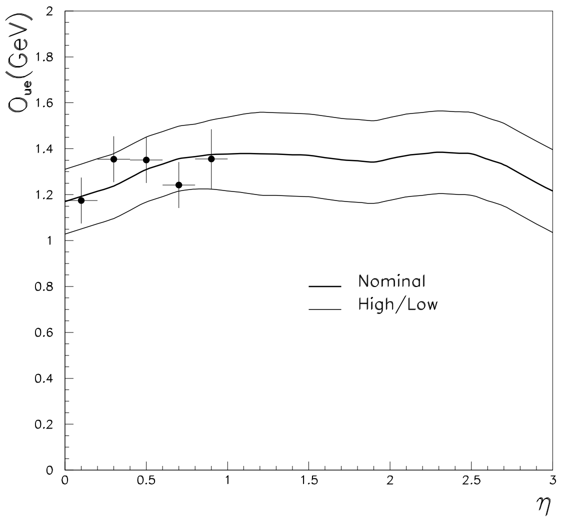

Figure 31. Physics underlying event offset O_{ue} versus η. Above $\eta = 0.9$, the result is an extrapolation.

cone algorithm ($R = 0.7$). The accuracy of the associated corrections are, however, on the same order. The missing E_T projection fraction method is well suited to calibrate K_T jets [34]. The uncertainty in R_{jet} for K_T and cone jets is about the same: (0.5–1.6%) for jet p_T = 50–450 GeV in the CC.

Overall, it may be possible to determine the jet momentum scale more accurately for K_T jets than the energy scale for cone jets, given the absence of a cone boundary in the former. The difference in precision could be large in the low p_T and high pseudorapidity range, where the cone showering correction is larger and more inaccurately determined. (The showering correction uncertainty contributes 1–3% [34] to the total error for $R = 0.7$ cone jets.)

4.5. Jet Momentum Resolutions of K_T Jets

One of the largest sources of uncertainty in jet measurements (besides the jet momentum scale) is the effect of a finite calorimeter jet momentum resolution. *A priori*, due to the absence of cone boundaries, K_T jets should be affected little by jet-to-jet fluctuations in the shower development. The jets will, of course, still be subjected to the effects of hadronization.

We compared jet energy resolutions for cone jets ($R = 0.7$) and momentum resolutions for K_T jets ($D = 1$) derived from a DØ Monte Carlo simulation using the HERWIG event generator plus the GEANT simulation of the DØ detector (Run I). The test was performed for an inclusive jet sample with average p_T = 60 GeV and 80 GeV in $|\eta| < 0.5$. Within statistical errors, σ_{p_T}/p_T for K_T jets and σ_{E_T}/E_T for cone jets are the same: 0.109 ± 0.009 and 0.105 ± 0.006 for K_T ($D = 1$) and cone ($R = 0.7$) jets at 60 GeV, and 0.10 ± 0.01 for both at 80 GeV. Preliminary measurements of K_T jet momentum resolutions and cone jet energy resolutions using Run I collider data support the previous statement. Note, however, that resolutions depend on the algorithm parameters R and D. Resolution studies for different (smaller) R and D parameters should be performed, as well as for different type of samples, for example quark or gluon enriched samples. These studies will make more clear how energy/momentum resolutions compare for cone and K_T jets.

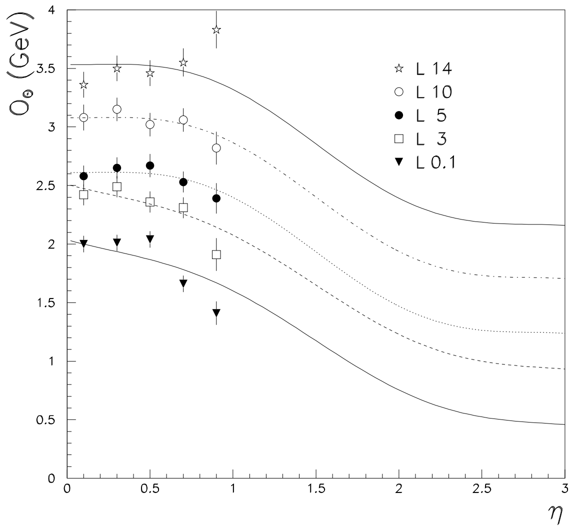

Figure 32. Offset due to uranium noise, pile-up and multiple interactions, O_{zb} versus η for different luminosities in units of 10^{30} cm^{-2}sec^{-1}. Above η = 0.9, the result is an extrapolation.

4.6. Testing QCD with the K_T Jet Algorithm

4.6.1. Jet Structure

The subjet multiplicity is a natural observable of a K_T jet [35, 36]. Subjets are defined by re-running the K_T algorithm starting with a list of preclusters in a jet. Pairs of objects with the smallest d_{ij} are merged successively until all remaining d_{ij} are larger than $y_{cut}E_T^2(jet)$, where $0 < y_{cut} < 1$ is a resolution parameter. The resolved objects are called subjets, and the number of subjets within the jet is the subjet multiplicity M. For $y_{cut} = 1$, every jet consists of a single subjet ($M = 1$). As y_{cut} decreases, the subjet multiplicity increases until every precluster becomes resolved as a separate subjet. At this level of detail the specific preclustering algorithm used clearly influences the result. A measurement of M for quark and gluon jets is a test of QCD, and may eventually be used

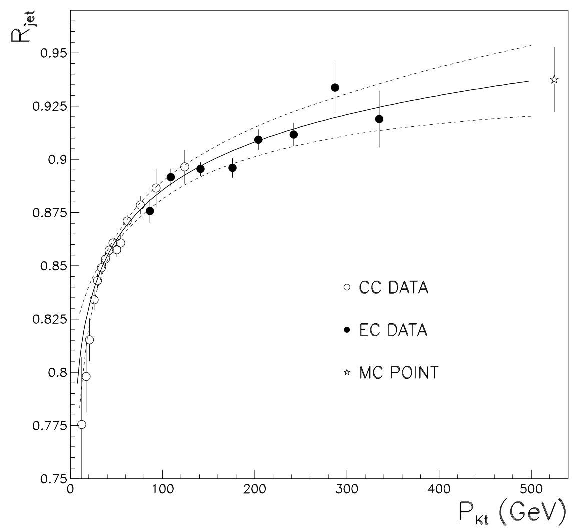

Figure 33. R_{jet} versus K_T jet momentum. The solid lines are the fit and the dashed band the error of the fit. (The three lowest points have nearly fully correlated uncertainties and are excluded from the fit.)

in Run II as a discriminant variable to tag quark jets in the final state. Fig. 35 shows a preliminary measurement of M by DØ [37], using Run I data (K_T algorithm with $D = 0.5$ and $y_{cut} = 0.001$). The ratio $R = \frac{\langle M_g \rangle - 1}{\langle M_q \rangle - 1}$ is $1.91 \pm 0.04(stat) \pm 0.23(sys)$. It is well described by the HERWIG Monte Carlo, and illustrates the fact that gluons radiate more than quarks.

4.6.2. Jet Production

Jet cross section measurements have been extensively used by both Fermilab Tevatron collaborations during Run I to test perturbative (NLO) QCD predictions, to test the available parton distribution functions at the x and Q^2 ranges covered by the Tevatron, and to search for quark compositeness [38, 39, 40, 41, 42, 43, 44, 45, 46, 47]. The higher center-of-mass energy and the larger data sample will allow the Tevatron experiments to extend the energy reach and precision of jet cross sections in Run II. The largest source of uncertainty in a jet cross section measurement is the jet energy (or momentum) scale. As an example, a 1% uncertainty in the jet energy calibration translates into a 5–6% (10–15%) uncertainty in the $|\eta| < 0.5$ inclusive jet cross section at 100 GeV (450 GeV). As a function of η, the jet cross section falls more quickly with transverse energy, and the cross section error is even larger.

The K_T jet algorithm may provide experimental advantages for jet production measurements. At DØ, the jet scale uncertainty for cone jets in the high E_T range is dominated by the contributions from the response and out-of-cone showering corrections. In Run II, the availability of more high E_T photon data and a more accurate determination of the position of the interaction vertex promise a reduction in the response uncertainty. Furthermore, the absence of out-of-cone showering losses in K_T jets will likely lead to improved jet cross section measurements in the forward η regions. Most of the Run I cross section results by CDF and DØ use jet energy measurements restricted to central regions ($|\eta| < 1$). A couple of exceptions to the rule are the DØ measurements of the pseudorapidity dependence of the jet cross section [45], and the test of BFKL dynamics in dijet cross sections at large pseudorapidity intervals [48].

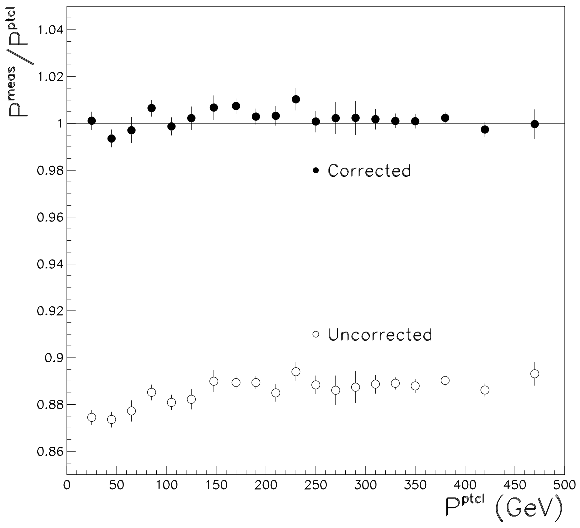

Figure 34. Monte Carlo verification test. The vertical bars are statistical errors. Systematic errors (not shown) are of the order of 0.01–0.02. The corrected $p_{jet}^{meas}/p_{jet}^{ptcl}$ ratio is consistent with unity within errors.

4.6.3. Event Shapes

Event shape variables in e^+e^- and ep interactions have attracted considerable interest over the last few years [49, 50, 51]. Little attention has been paid to measurements or calculations of event shape variables at hadron colliders. An important example is thrust which is defined as:

$$T = \max_{\hat{n}} \frac{\sum_i |\vec{p_i} \cdot \hat{n}|}{\sum_i |\vec{p_i}|}\,, \qquad (57)$$

where the sum is over all parton, or particle momenta.

A LO jet rate calculation with two partons in the final state yields $T = 1$. A NLO calculation, with three partons in the final state would produce a deviation from $T = 1$ (LO in thrust). A NNLO prediction with four partons in the final state would then give a NLO prediction of thrust. At all orders, thrust would take values from 0.5 to unity. In other words, thrust mea-

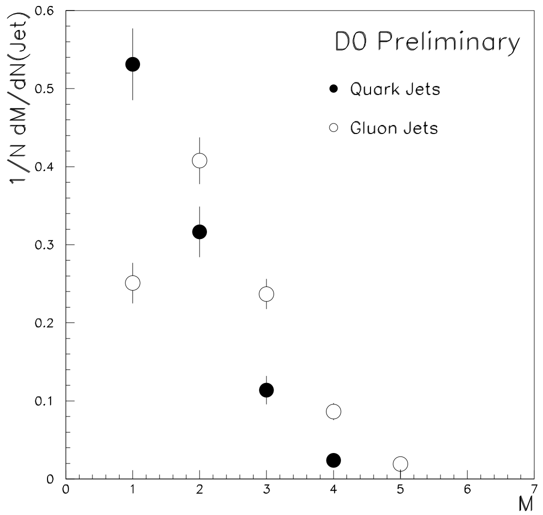

Figure 35. Subjet multiplicity for quark and gluon jets at DØ.

sures the pencil-likeness of the event: $T \to 1$ for back-to-back events, and $T < 1$ as more radiation is present. The low scales introduced by soft and collinear emission in events with $T \lesssim 1$ could be the reason for the observed discrepancy between LO and NLO calculations and experimental e^+e^- data [49]. Resummation of higher-order perturbative terms could lead to a better understanding of the problem.

The simplest measurements of thrust we can perform are the thrust distributions in jet events, changing the definition of thrust to sum over all the jets in the event. In order to be able to resum logarithms of the jet resolution scale, jets must be defined using an algorithm such as the K_T algorithm [52]. The contribution of the underlying event, and multiple $p\overline{p}$ interactions in hadron colliders, introduce an experimental difficulty not present in lepton colliders. It is possible, however, to minimize these systematics by choosing carefully the variable to measure.

We can also define transverse thrust, T_T, by replacing particle momenta by transverse momenta in Eq. 57. T_T is Lorentz invariant for boosts along the beam axis, an advantage in the case of hadron colliders.

Figs. 36–38 show the difference between T_T calculated from particle-level jets (reconstructed from final-state hadrons) and T_T from calorimeter-level jets (reconstructed from cells). HERWIG was used as the generator, and SHOWERLIB [32] (a fast version of GEANT) simulated the Run I detector. In all cases jets are reconstructed with the K_T jet algorithm ($D = 1$). Fig. 36 shows a T_T distribution for events with $H_T = 90$–150 GeV, where H_T is the scalar sum p_T of all jets above 8 GeV. H_T was chosen instead of Q^2 as an estimator of the hard scattering energy scale of the event. All jets with $p_T > 8$ GeV contribute to T_T. The full circles are the particle-level or "true" distribution. The triangles are the distribution as seen in the calorimeter in an ideal environment with no offset (underlying event, multiple $p\overline{p}$ interactions, pile-up, or noise). The open circles are a calorimeter-level distribution which includes a random collider crossing event at a luminosity of $5 \times 10^{30}\text{cm}^{-1}\text{sec}^{-1}$. While the effect of calorimeter momentum response, resolution, and showering is minimal, the offset distorts the distribution to a large extent.

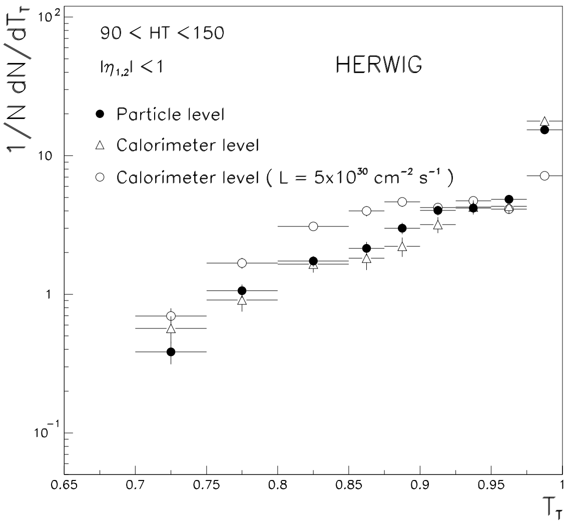

Figure 36. All jets with $p_T > 8$ GeV contribute to T_T. The full circles are the particle-level or "true" distribution. The triangles are the distribution as seen in the calorimeter in an ideal environment with no offset (underlying event, multiple $p\overline{p}$ interactions, pile-up, or noise). The open circles are a calorimeter-level distribution which includes a random crossing collider event at a luminosity of $5 \times 10^{30}\text{cm}^{-2}\text{sec}^{-1}$.

In Fig. 37, the thrust definition was modified to allow only the three leading jets (above 8 GeV) to contribute to the thrust (T_{T3}) and to H_T (now H_{T3}). The difference between the true and the fully-simulated calorimeter distribution is now much smaller. Finally, in Fig. 38, only the two leading jets contribute to the thrust (T_{T2}) for events with $H_{T3} = 90$–150 GeV. Now the calorimeter distribution is even closer to the true distribution. Although T_{T3} and T_{T2} are not calculated from all final-state particles (to reduce contamination), they implicitly include the information about the whole radiation pattern through the p_T and $\eta - \phi$ position of the first few leading jets.

Event shape variables, like a modified version of thrust, can be studied with precision at the Tevatron. The use of the K_T algorithm, infrared safe at all orders

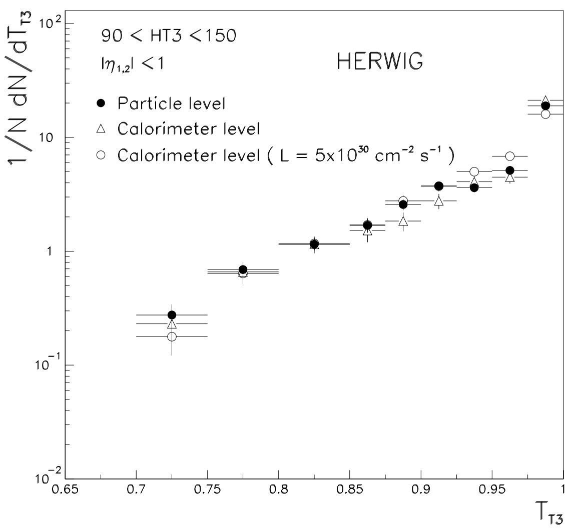

Figure 37. Same as Fig. 36 but only the three leading jets contribute to T_T, now T_{T3}. H_{T3} is the scalar sum p_T of the three leading jets in the event.

in perturbation theory, provides a test of the newly available hadronic three jet production calculations at NLO [8, 53]. In the QCD calculation of the thrust variables defined in this section, there are no large logarithms in the $T \to 1$ limit. Then, it is neither possible nor necessary to resum them. However, if we redefine thrust in terms of subjets or tracks, the measurement is more interesting and resummation becomes an issue [54]. The availability of the contributions of higher-order terms through a resummation calculation would be desirable, in that case, to improve the understanding of the range $T \lesssim 1$. In Run II, both the CDF and the DØ detectors will have upgraded tracking systems. This will allow both experiments to implement improved techniques for the identification of hadrons using both the calorimeter and the tracking detectors.

The H_T dependence of $\langle 1 - T \rangle$, in the range where resummation and hadronization effects are small, could also provide a measurement of α_s.

5. Conclusions

Jet algorithms present a challenge to experimentalists and theorists alike. Although everyone "knows a jet when they see it," precise definitions are elusive and detailed. The jet working group has attempted to provide guidelines and recommendations for jet algorithm development. The end product of the year-long effort has been standardized jet cone and K_T algorithms, and the recommendation to use 4-vector, E-scheme kinematic variables. A legacy algorithm or ILCA has been suggested which will bridge the gap between past results and improved theoretical calculations. This document has addressed concerns about the use and calibration of K_T jets.

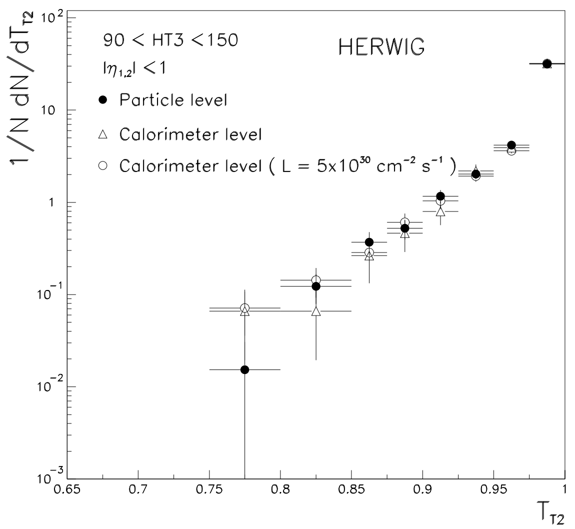

Figure 38. Same as Fig. 36 but only the two leading jets contribute to T_T, now T_{T2}. H_{T3} is the scalar sum p_T of the three leading jets in the event.

We strongly recommend that both CDF and DØ adopt standard algorithms for Run II. Since continued development is probably inevitable, we encourage continued dialogue. The usefulness of standardized algorithms, which can replicate past results and meet experimental and theoretical requirements, makes continued coordination well worth the effort.

6. Acknowledgments

We would like to thank Stefano Catani, Dave Soper and the other members of the Les Houches QCD working group for stimulating discussions which lead to the final definition of ILCA. We are also pleased to thank Rick Field, David Stuart, and numerous members of the CDF and DØ Collaborations for many helpful discussions. Financial support by the Department of Energy and National Science Foundation (USA) and CONICET and UBACyT (Argentina) is gratefully acknowledged. Stephen D. Ellis would also like to thank the University of Washington Office of Research for partial support and Fermilab for a Fermilab Frontier Fellowship through the Theoretical Physics Department.

REFERENCES

1. G.C. Blazey and B.L. Flaugher, Ann. Rev. of Nucl. and Part. Sci., Vol. 49 (1999), FERMILAB-Pub-99/038-E.
2. J.E. Huth *et al.* in *Proceedings of Research Directions For The Decade: Snowmass 1990*, July, 1990,

edited by E.L. Berger (World Scientific, Singapore, 1992) p. 134.

3. B. Abbott *et al.* (DØ Collaboration), Fermilab-Pub-97-242-E (1997).
4. S.D. Ellis, Z. Kunszt, D. Soper, Phys. Rev. Lett. **69** 3615 (1992) and S. Ellis CERN-TH-6861-93, Jun 1993. 13pp. Presented at 28th Rencontres de Moriond: QCD and High Energy Hadronic Interactions, Les Arcs, France, 20-27 Mar 1993. Published in Moriond 1993: Hadronic: 235–244; hep-ph/930628. B. Abbott *et al.*, Fermilab-Pub-97-242-E (1997).
5. Y. L. Dokshitzer, G. D. Leder, S. Moretti and B. R. Webber, JHEP **9708**, 001 (1997) [hep-ph/9707323]; A. T. Pierce and B. R. Webber, Phys. Rev. **D59**, 034014 (1999) [hep-ph/9807532]; M. H. Seymour, Nucl. Phys. **B513**, 269 (1998) [hep-ph/9707338].
6. Z. Casilum and R. Hirosky, *Jet Reconstruction Efficiency*, DØ Note 3324, unpublished.
7. S. Catani et al., *Report of the QCD and standard model working group* for the Workshop *Physics at TeV Colliders*, Les Houches, France, 8-18 June 1999.
8. Z. Bern, L. Dixon, and D.A. Kosower, Ann. Rev. Nucl. Part. Sci. **46**, 109 (1996) [hep-ph/9602280]; W.B. Kilgore and W.T. Giele, Phys. Rev. **D55**, 7183 (1997); W.B. Kilgore and W.T. Giele, *Hadronic three jet production at next-to-leading order*, e-Print hep-ph/9903361.
9. H. Baer, J. Ohnemus and J. F. Owens, Phys. Lett. **B234**, 127 (1990); J. Ohnemus, Phys. Rev. **D50**, 1931 (1994) [hep-ph/9403331]; R. K. Ellis and S. Veseli, Phys. Rev. **D60**, 011501 (1999) [hep-ph/9810489]; J. M. Campbell and R. K. Ellis, Phys. Rev. **D60**, 113006 (1999) [hep-ph/9905386].
10. E. Mirkes and D. Zeppenfeld, Phys. Lett. **B380**, 205 (1996); E. Mirkes, preprint TTP-97-39 (1997) [hep-ph/9711224].
11. S. Catani and M. H. Seymour, Nucl. Phys. **B485**, 291 (1997) [hep-ph/9605323]; D. Graudenz, hep-ph/9710244; B. Potter, Comput. Phys. Commun. **119**, 45 (1999) [hep-ph/9806437].
12. Private communication from S.D. Ellis and D.E. Soper.
13. S.D. Ellis, Z. Kunszt, and D.E. Soper, Phys. Rev.Lett. **69**, 291 (1992); *ibid.* 1496 (1992); Phys. Rev. Lett. **62**, 2121 (1990).
14. T. Sjostrand, "Pythia 5.7 and jetset 7.4 physics and manual," CERN-TH-7112-93.
15. This idea was originally suggested by S.D. Ellis, D.E. Soper and H.-C. Yang to the OPAL Collaboration. See the discussion in references [16] and [19].
16. M.H. Seymour, Nucl. Phys. **B513**, 269 (1998).
17. B. Potter and M.H. Seymour, J. Phys. **G25**, 1473 (1999).
18. N. Kauer, L. Reina, J. Repond and D. Zeppenfeld, Phys. Lett. **B460**, 189 (1999) hep-ph/9904500.
19. OPAL Collaboration, Z. Phys., C63, 197–211 (1994).
20. S. Catani, Yu.L. Dokshitzer, M.H. Seymour, and B.R. Webber, Nucl. Phys. B **406** 187 (1993).
21. S. Catani, Yu.L. Dokshitzer, and B.R. Webber, Phys. Lett. B **285** 291 (1992).
22. S.D. Ellis and D.E. Soper, Phys. Rev. **D48** 3160 (1993).
23. D. Buskulic *et al.* (ALEPH Collaboration), Phys. Lett. B **346**, 389 (1995).
24. E.W.N. Glover and D.A. Kosower, Phys. Lett. B **367** 369 (1996).
25. K.C. Frame in Proceedings of the VIII International Conference on Calorimetry in High Energy Physics, Lisbon, 1999 (to be published); Ph.D. thesis, Michigan State University 1999 (unpublished).
26. R. Blair *et al.* (CDF Collaboration), The CDF II Detector: Technical Design Report, FERMILAB-PUB-96/390-E (1996).
27. B. Abbott *et al.*, Nucl. Instr. and Meth. A (1999) 352.
28. K. Frame (DØ Collaboration). To be published in the proceedings of the Seventh International Conference on Calorimetry in High Energy Physics.
29. To be submitted to Nucl. Instr. and Meth. (in preparation).
30. S. Abachi *et al.*, Nucl. Instr. and Meth. A (1994) 185.
31. G. Marchesini and B. Webber, Nucl. Phys. B 310 (1988).
32. J. Womersley for the DØ Collaboration, presented at the International Conference on High Energy Physics, Dallas, USA (1992), FERMILAB-CONF-92-306 (unpublished).
33. DØ detector simulation packaged based on GEANT. GEANT by R. Brun and F. Carminati, CERN Program Library Long Writeup W5013 (1993).
34. A. Goussiou for the DØ Collab., FERMILAB-PUB-99/264-E (1995).
35. M. Seymour, Nucl. Phys. B **421** 545 (1994).
36. J. Forshaw and M. Seymour, JHEP **9909** 009 (1999).
37. R. Snihur (DØ Collaboration), 7th International Workshop on Deep Inelastic Scattering and QCD DESY Zeuthen, Germany 19-23 Apr 1999. Nuclear Physics B (Proc. Suppl.) 79, 494–496 (1999).
38. B. Abbott *et al.* (DØ Collaboration), Phys. Rev. Lett. **82**, 2451–2456 (1999).
39. F. Abe *et al.* (CDF Collaboration), Phys. Rev.

Lett. **77**, 438 (1996).

40. B. Abbott *et al.* (DØ Collaboration), Phys. Rev. Lett. **80**, 666–671 (1997).
41. F. Abe *et al.* (CDF Collaboration), Phys. Rev. Lett. **77**, 5336–5341 (1996).
42. B. Abbott *et al.* (DØ Collaboration), Phys. Rev. Lett. **82**, 2457–2462 (1999).
43. T. Affolder *et al.* (CDF Collaboration). Submitted to Phys. Rev. D (1999).
44. J. Krane (CDF and DØ Collaborations), Iowa State University, Proceedings of DPF99.
45. L. Babukhadia (DØ Collaboration), QCD Multiparticle Production, XXIX International Symposium on Multiparticle Dynamics (ISMD99), August 9–13, 1999, Brown University, Providence. Proceedings in preparation.
46. H. Schellman (DØ Collaboration), 7th International Workshop on Deep Inelastic Scattering and QCD (DIS 99), Zeuthen, Germany, 19–23 Apr 1999. FERMILAB-CONF-99-170-E, Jun 1999.
47. F. Chlebana (CDF Collaboration), 7th International Workshop on Deep Inelastic Scattering and QCD DESY Zeuthen, Germany 19–23 Apr 1999.
48. B. Abbott *et al.* (DØ Collaboration). Submitted to Phys. Rev. Lett. (1999).
49. E. Gardi, G. Grunberg (Ecole Polytechnique). High Energy Physics International Euroconference on Quantum Chromo-Dynamics (QCD99), Montpellier, France, 7–13 Jul 1999 (hep-ph/9909226).
50. V. Antonelli, M. Dasgupta, G. Salam, BICOCCA-FT-99-32, Dec 1999. 25pp. (hep-ph/9912488).
51. G. Korchemsky, G. Sterman, Nucl. Phys. **B555**, 335–351, 1999 (hep-ph/9902341).
52. M. Seymour, CERN-TH/95-225 (hep-ph/9603281).
53. Private communication with W. Giele.
54. Private communication with M. Seymour.