An Investigation into the Natural Width of the Semi-Classical Jet Reclustering Algorithm using Monte-Carlo Simulation

Candidate Number: XXXXXXXX

April 16, 2014

Abstract

Using Monte-Carlo simulations of $Z' \to u + \bar{u}$, I investigated the widths of jets reconstructed by the Semi-Classical sequential recombination algorithm in comparison to other existing jet reclustering algorithms. I found that the Semi-Classical algorithm is effective at reconstructing the 60% jet width for a jet width parameter, $R = 1.5 - 2.0$; where existing algorithms are effective for $R = 0.5 - 1.0$.

Introduction

In high energy physics, jets are narrow beams of propagating high energy hadrons. The production of jets is a common process in high energy elementary particle collisions. Jets are typically produced by an interaction that contains free quarks or gluons in its final state, such as the decay of a photon into two quarks. When the free quarks and gluons seperate to a distance of order 1 fm the strong force will become large, which will decelerate the coloured objects rapidly. This deceleration will cause them to radiate hadrons (mostly light π-mesons); similar to how a decelerated charge will radiate photons through bremsstrahlung. As the original quark or gluon will have been boosted, the radiated hadrons will be collimated due to the relativistic headlight effect, forming the narrow beam of hadrons that we call a jet. [1]

Jets produced at high energy collider experiments are detected when the light hadron constituents of the jets deposit energy into the hadronic calorimeter. To identify a jet from the calorimeter data, one must apply a reclustering algorithm to reconstruct the jet. One set of reclustering algorithms are the sequential recombination algorithms, which selectively add together the calorimeter energy deposits to form the jet. In this report I consider a new sequential recombination algorithm, the Semi-Classical(SC) jet algorithm [2], in comparison to the three most commonly used sequential recombination algorithms; the k_t, the anti-k_t and the Cambridge-Aachen(CA) algorithms [3].

The latter three algorithms begin with a set of 4-vectors (clusters) which represent the energy deposits in the calorimeter. One then introduces an inter-jet distance between clusters i and j defined as,

$$d_{ij} = min[(p_{Ti})^a, (p_{Tj})^a] \left(\frac{\Delta R_{ij}}{R}\right)^2, \qquad \Delta R_{ij} = \sqrt{(y_i - y_j)^2 + (\phi_i - \phi_j)^2} \qquad (1)$$

and a particle-beam distance for cluster i defined as,

$$d_{iB} = (p_{Ti})^a \tag{2}$$

where $y = \tanh^{-1}(v/c)$ is rapidity, ϕ is azimuthal angle and p_T is transverse momentum (momentum perpindicular to the beam axis of the colliding particles) of the cluster. R is the jet width parameter, which is a free parameter of the algorithm. I will discuss the significance of this parameter below.

The inter-jet and particle-beam distances are not physical distances as such, but should instead be thought of as dimensionful measures of how likely it is that clusters i and j represent energy deposits caused by hadrons from the same jet. If the inter-jet distance for a pair of clusters is smaller than the particle-beam distances for the two clusters ($d_{ij} < d_{iB}$) then it is highly likely that the two clusters are from the same jet. In contrast, if $d_{ij} > d_{iB}$ then it is unlikely that the two clusters are from the same jet.

The algorithm then follows the steps

1. Calculate d_{ij} and d_{iB} for all combinations of clusters.
2. Find the minimum of the d_{ij} and d_{iB}.
3. If the minimum is a d_{ij} combine cluster i and j to form a new cluster and return to step 1.
4. If the minimum is a d_{iB} then call cluster i a final-state jet, remove it from the set and return to step 1.
5. Stop when all clusters have been declared as jets.
6. Remove all jets that have $p_T < 100$ GeV. [1]

The parameter a in Eq. (1) and (2) takes the value $a = 2$ for the k_t algorithm, $a = -2$ for the anti-k_t algorithm and $a = 0$ for the Cambridge-Aachen algorithm [3].

The Semi-Classical jet algorithm [2] is a modified sequential recombination algorithm constructed by considering the classical theory of radiation by a moving charge. This algorithm follows the same steps as outlined above but, instead of using Eqns. (1) and (2), we use the re-defined inter-jet and particle beam distances

$$d_{ij} = \frac{1}{4}(m_{Ti} + m_{Tj})^2 \left(\frac{\Delta R_{ij}}{R}\right)^3 \tag{3}$$

$$d_{iB} = m_{Ti}^2 \tag{4}$$

where $m_{Ti} = \sqrt{p_{Ti}^2 + m_i^2}$ is the transverse mass of cluster i, and m_i is its mass. The algorithm is formed by considering the angular distribution of radiation being classically emmitted by a moving charge, which contains a relativistic boost factor $\gamma \sim E_i + E_j$. This boost factor is then modified so it contains only longitudinaly invariant quantities, which

[1] We are using natural units where c = 1, hence momentum can be expressed in units of energy.

are quantities that are invariant under lorentz boost transforms along the beam axis of the colliding particles, thus the boost factor becomes $\gamma \sim m_{Ti} + m_{Tj}$. The algorithm has been shown to effectively subtract background as it reconstructs the jet [2], in a manner that is similar to a process called pruning, which is a background subtraction technique often applied to reconstructed jets formed by sequential recombination algorithms [4]. However, this pruning nature of the SC jet algorithm can also remove genuine jet constituents as well.

In this report I will investigate how the sequential recombination algorithms listed above reconstruct the width of jets. This is important as in the inter-jet distances, defined in Eqns. (1) and (3), the jet width parameter, R, effectively gives the maximum width of a reconstructed jet. This is because when $\Delta R_{ij} > R$ for a pair of clusters, we will find that $d_{iB} < d_{ij}$ for the cluster with the smaller value of p_T or m_T. Hence, the algorithm will never merge these two clusters as it will always skip to step 4 of the algorithm and remove one of the clusters as a jet. R is a free parameter of the algorithms, which means that the user sets the R value. Choosing an appropriate value for R is important in reconstructing jets correctly with minimal background. An R value that is too large may allow too much background to be included in the reconstructed jet, whilst an R that is too small will mean that we may lose significant amounts of the actual jet in the reconstruction. By better understanding how the algorithms reconstruct the width of the jet for different R parameters, we will be able to better select the value of the R parameter to reconstruct the natural width of the jet.

The event I used to produce the jets was $Z' \to u + \bar{u}$, where the Z' boson is a hypothetical gauge boson that is predicted by various beyond Standard Model theories [5, 6, 7]. This particular decay was chosen as it produces two high energy free quarks that will form jets with a low background contribution from the underlying event, because there is nothing else in the final state. The two quarks also have ϕ values differing by π because the Z' created would have zero tranverse momentum, hence the resulting particles of the decay must also have zero total transverse momentum. The opposite ϕ values meant that there was a clear separation between the two jets which was used to differentiate between them.

To conduct these investigations I used Monte-Carlo Simulations to produce the data sets that the jet reconstructing algorithms were applied to. Monte-Carlo Simulations are a broad set of computering techniques that use random numbers to produce numerical results based on probabilistic distributions. Simulations were used to give us the opportunity to apply the algorithms to data sets that are independant of detector effects and where we know the true 4-momentum of the original quark that caused the jet. However, it must be noted that, whilst simulation is a useful tool in analysing the peformance of jet clustering algorithms, they should not replace testing with experimental data.

The first section of this report will outline the details of the simulation and the computational tools used to simulate and analyse the data. The second section will present the analysis of the data in three subsections; firstly I will characterise the distribution of

the jet energy from the simulation, then I will consider the width of jets reconstructed using the SC jet algorithm and lastly I will compare the width reconstruction peformance of the SC jet algorithm to the k_T, anti-k_T and CA algorithms. In the final section I will present the conclusions of the analysis and outline areas for futher research.

1 Simulation

To produce jets for analysis I used the simulation package Pythia, version 8.176, which is a standard tool for creating Monte-Carlo simulations of high-energy particle collisions [8]. In the simulated $Z' \to u + \bar{u}$ events, the Z' was produced in a proton-proton collision with a centre of mass energy of 7 TeV. As the mass of the Z' is not a fixed parameter, the value of $m_{Z'}$ could be varied to produce jets with a range of initial quark energies, E_q. I generated events for $m_{Z'}$ ranging from 200 to 2000 GeV in steps of 200 GeV, with 100,000 events for each mass. As $E_q \sim \frac{1}{2} m_{Z'}$, this range of masses gave me a large sample size of jets with E_q in the range 100-1000 GeV. As step 6 of the reclustering algorithms oulined above applies a p_T cut of 100 GeV, jets with $E_q < 200$ GeV are not considered as they will not be consitently reconstructed by the algorithms, so cannot be used to investigate the algorithms.

To analyse these events I used Rivet, which is a Monte-Carlo validation package [9]. Using Rivet I built a programme to analyse the simulations on an event-by-event basis. The output of the Pythia simulation is transfered into the Rivet analysis programme using a FIFO (First In First Out) pipe that prevents excessively large temporary data files being required, which may have limited the size of the simulation. The Rivet analysis programme is built using various "projections" that are objects in Rivet that allow the user to extract required information from the simulation. My analysis programme used the visible final state projection, to output the 4-momentum of the final-state particles (excluding neutrinos), and projections from the FastJets package, to apply the various jet clustering algorithms to the final state particles [10]. I was also able to extract the 4-momentum of the inital quarks from the Pythia event record within the analysis.

A problem that I encountered whilst using Rivet was that its internal plotting package, YODA, did not have the capabilities to produce many of the data plots required, in particular 2D histograms. To overcome this problem I was able to interface the Rivet analysis package with ROOT, which is the standard analysis and plotting package used in high energy physics analyses [11].

2 Analysis

2.1 Chacterising the Jet Width

In this subsection I will characterise the true width of the jets by considering the final state particles, as selected by the visible final state projection. This characterisation I will call the final state jet widths, which will be used as benchmark to compare the width reconstruction of jet reclustering algortithms to. Figure 1 shows the positional distribution of final-state particle energies alongside the position of the initial quarks from the Z' decay for a single event. The x-axis represents pseudorapidity, defined as $\eta = -\ln[\tan(\theta/2)]$, where θ is the polar angle from the beam-pipe of the colliding particles. Pseudorapidity is used as it is a quantity that is mathematically similar to rapidity, y, but calculable from just the polar angle. In Figure 1 we can see that most of the final state energy has been deposited in the jets (the clusters of energy around the initial quark postion). We can also see that much of the background energy from the underlying event is at high absolute values of η. Hence, a pseudorapidity cut of $\eta < |2|$ was applied to reduce background.

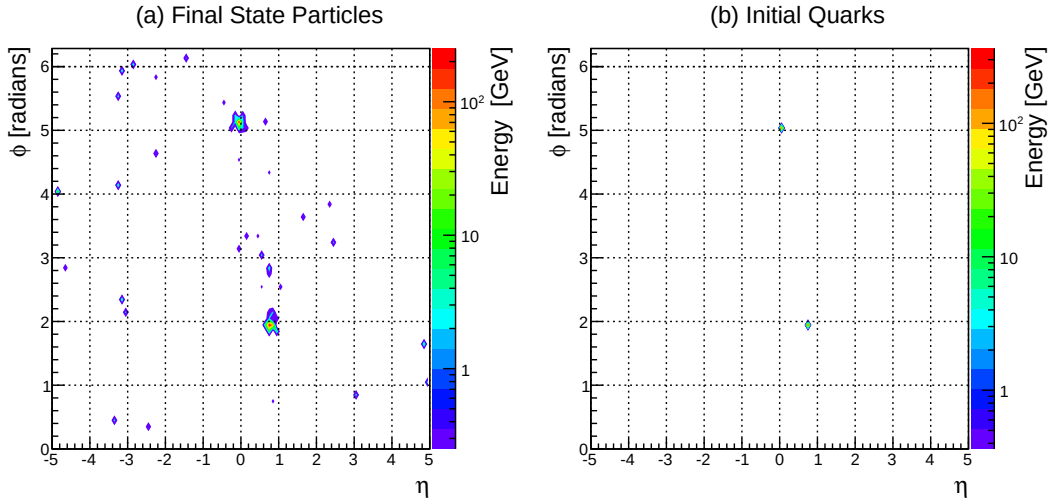

Figure 1: A single event energy distribution of (a) the final state particles (b) the initial quarks on an η-ϕ grid with a resolution of 0.1. The energy axis is a log-scale for both plots.

As the two quarks are clearly seperated in ϕ-space, analysis of the the width of the jets was done using ϕ to prevent a contribution from the opposite jet. Figure 2 is a 2D histogram, containing final state particles from all events, that shows the ϕ distribution of the energy of the final state particles about the initial quark axis, for different values of quark energy. This distribution has been normalised by the quark energy. This effectively gives the azimuthal energy profile of the average jet produced. The distribution shows that jets becomes thinner at higher energies and their widths are on the scale 0.01-0.1 radians.

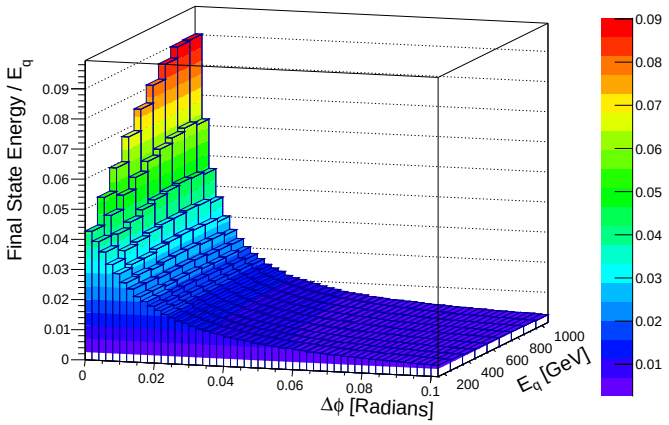

Figure 2: The normalised ϕ distribution of final state energy about the quark axis for different values of quark energy, E_q. The distributions were normalised by E_q

Figure 2 gives us an idea of the shape of the final state jets in ϕ-space. However, it is required that information about the width of these jets is quantised, to be able to compare to the widths of the jets reconstructed by the reclustering algorithms later. This is done by finding the value of $\Delta\phi$ that contains 60% and 95% of the intial quark energy in the final state as a function E_q. I will call these values r_{60} and r_{95} respectively.

The technique used to calculate the r_{60} and r_{95} values goes as follows; first sum the energies of final state particles close to the quark axis in ϕ to find the values of r_{60} and r_{95} for each individual jet. The distributions of the individual r_{60} and r_{95} are calculated for quark energies 200-1000 GeV with bins of width 100 GeV. Figure 3 shows these distributions for the quark energies of 200, 600 and 1000 GeV. We can see that the distributions consist of a peak with a long tail; I found that this long tail distorted the means of the distributions causing the means to be unrepresentatively large. Instead a fifth-order polynomial was used to fit these distributions to identify the maxima; the fitting is shown in Figure 3 by the red-line. The values of r_{60} and r_{95} at the maxima are what I shall use to represent the final state jet widths.

Figure 4 shows the final state jet widths r_{60} and r_{95} as a function of the quark energy. This shows that the widths of the jets are of order 0.1-0.01 radians and become narrower at higher energies. These distributions appear consistent with the jet shape in Figure 2.

However, one must exercise some caution in declaring the final state jet widths, calculated above, as the absolute true widths of the jets. An effect which could cause the widths to be be misrepresentative is that if the free quark loses energy through processes apart from hadronisation, for example gluon radiation [12], then the summation process in the

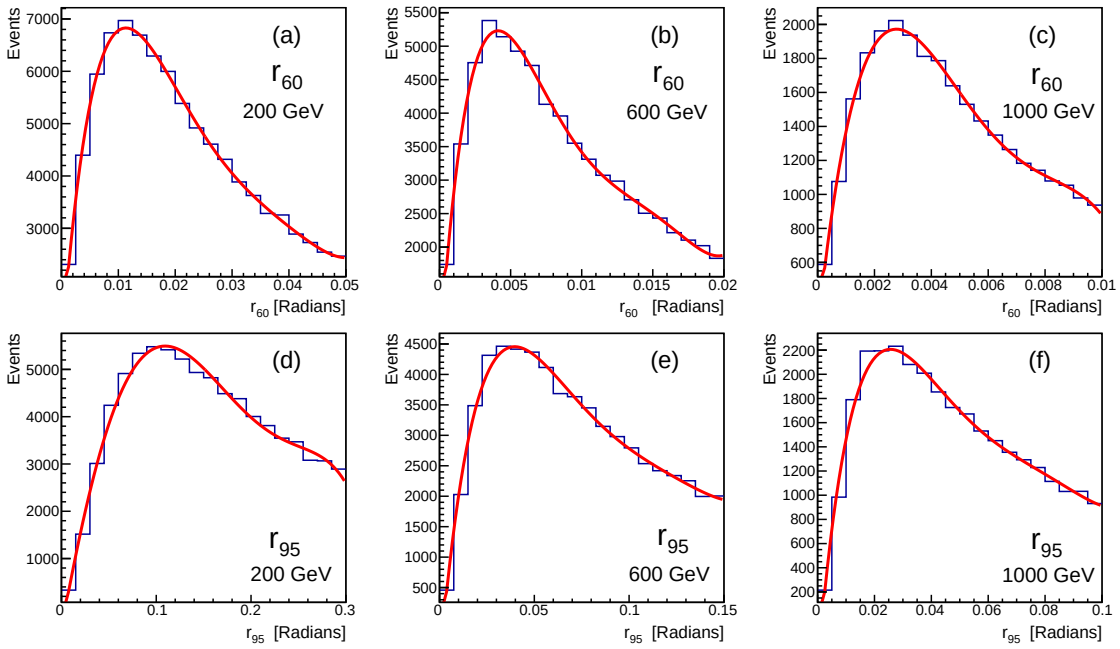

Figure 3: Histograms to show the final state distributions of r_{60} for inital quark energy, E_q = 200, 600 and 1000 GeV [(a), (b) and (c)] and r_{95} for E_q = 200, 600 and 1000 GeV [(d), (e) and (f)]. A fifth-order polynomial has been fitted to these to identify the maxima of the distributions.

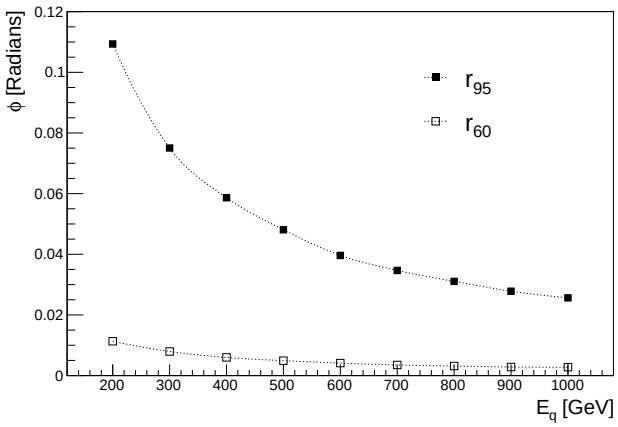

Figure 4: The 95% and 60% final state jet widths as a function of inital quark energy.

width characterisation will be forced to include some wider angle background to account for the lost quark energy. This will cause the r_{95} and r_{60} values to be slightly wider than they should be. In particular the r_{95} value will be sensitive to this effect as it is already at the edge of the jet distribution. Although some protection is provided against this widening effect by using the maxima of the fitted distribution, hence excluding the tail where most of the anomlous events will lie, there may still be some effect.

2.2 Applying the Semi-Classical jet algorithm

Now that I have developed a measure of the final state jet width, I can apply the SC jet algorithm to the final state particles and analyse the width of the reconstructed jets in comparison to the final state jets. For the purposes of this discussion I will define an SC jet to be a jet that has been reconstructed by the SC jet algorithm.

The first step is to see how accurately the SC jet algorithm is reconstructing the 4-momentum of the jet by comparing the axis and energy of the SC jets to the inital quarks. In Figure 5 we can see that SC jets for a jet width parameter of $R = 1$, closely match the ϕ position of the intial quark, showing that the SC jet algorithm peforms well in reconstructing the direction of the jet. Figure 6 shows that the SC jet algorithm reconstructs energies predominantly accurately with some loss of energy, mostly within 50 GeV for lower quark energies and within 100 GeV for larger quark energies. The loss of energy is because the background subratcting nature of the SC jet algorithm, as described in the introduction, will mean that the algorithm removes some of the actual jet constituents in some cases. This inaccuracy in reconstructing the energy does not completely undermine the integrity of the SC jets because it is small for the majority of events.

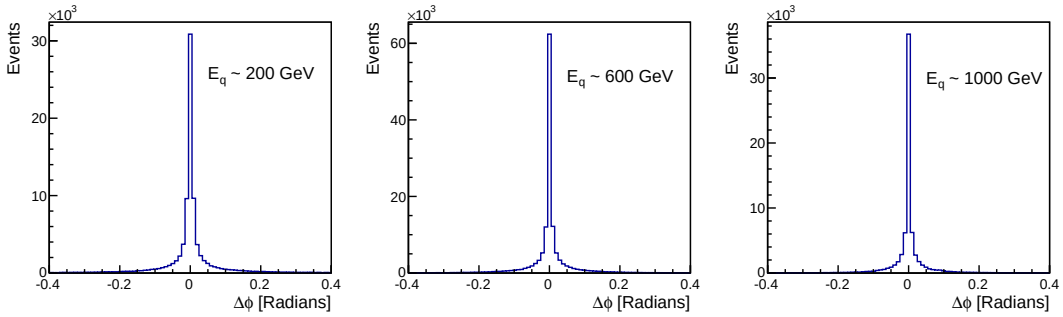

Figure 5: Histograms to show the difference in ϕ between the quark and the reconstructed SC jet axis with $R = 1$, for different quark energies. ($\Delta\phi = \phi_q - \phi_{jet}$)

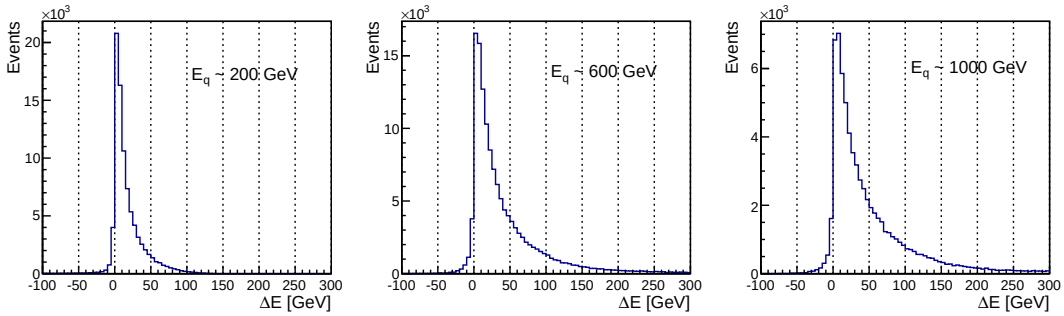

Figure 6: Histograms to show the difference in E between the quark and the reconstructed SC jet with $R = 1$, for different quark energies. ($\Delta E = E_q - E_{jet}$)

The next step is to calculate r_{60} and r_{95} of the SC jet, using the same technique employed for the final state jets in subsection 2.1. However, as this is profiling the width of the SC jet, one must use the SC jet constituents instead of the final state particles, sum around the SC jet axis and use the SC jet's energy to calculate the 60% and 95% thresholds, instead of using the initial quark axis and energy as before. The distributions of the individual r_{95} and r_{60} have a peak and a long tail; which is a similar shape to those for the final state jets in Figure 3. Hence, fitting with a fifth-order polynomial could be used again to interpolate the values for r_{60} and r_{95} at the maxima.

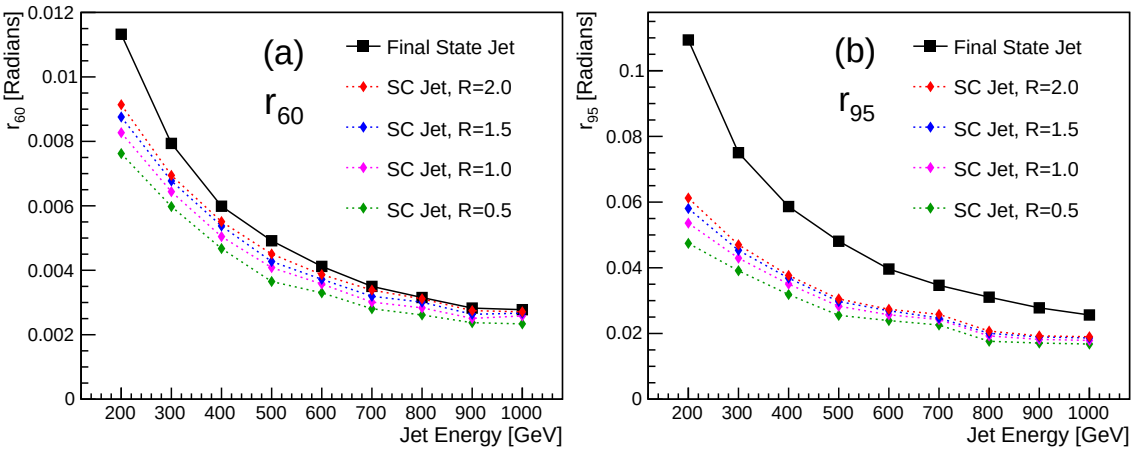

Figure 7: The jet widths, (a) r_{60} and (b) r_{95}, for the final state jets compared to the jets reconstucted using the SC jet algorithm with varying jet width parameter, R.

Figure 7 shows the r_{95} and r_{60} values for jets reconstructed by the SC jet algorithm with different R values, compared to the final state jet widths calculated in subsection 2.1. The figure shows that the SC jet algorithm reconstructs the r_{60} of the jet with reasonable accuracy, particularly at high energy. The reconstructed jets are slightly thinner than the final state jets, which is due to the pruning nature of the SC jet algorithm removing some of the true jet constituents. However, the SC jet algorithm reconstructs the r_{95} thinner by a factor of approximately 1.5-2.5. The thinner r_{95} values may be due to the jet algorithm cutting off significant amounts of the true jet at higher angles. However, this seems unlikely as we are selecting values of R up to 2.0, which represents the effective maximum width of a reconstructed jet. Therefore, one must also consider that it could be the final state energy jet r_{95} construction that is too wide, due the effect of missing quark energy discussed in the last paragraph of subsection 2.1. The cause of the seemingly anomolous bump at 700 GeV of the r_{95} is not understood and requires further analysis. However, it seems unlikely to me that it has any physical importance.

Figure 7 also shows that the jets reconstructed using the higher values of R have a more accurate jet width than those of lower R. This is because the larger R value suppresses the inter-jet distances d_{ij} as desribed in Eq. (3), meaning that more cluster mergers are

allowed. This observation indicates that for a free quark jet, produced by a clean channel such as $Z' \rightarrow u + \bar{u}$, a large value of R parameter ($\sim$1.5-2.0) should be chosen for the SC jet algorithm to optomise jet width reconstruction. However, there are other cases where there may be more background; such as events with more underlying event, two jets produced near each other in the final state, or pile-up (which is when many events occur at roughly same time [13]). In these cases it is highly likely that a large R parameter would cause the reconstructed jet to include some of these non-jet events and hence the resconstructed widths would become artificialy large. Futher investigation is required for these cases.

2.3 Comparison to Current Sequential Recombination Algorithms

In this subsection I will compare the width of jets reconstructed by the SC jet algorithm to that of the anti-k_T, k_T and Cambridge-Aachen (CA) algorithms. For the purposes of this discussion, I shall collectively call anti-k_T, k_T and Cambridge-Aachen algorithms as the current algorithms and I shall also define an anti-k_T jet, a k_T jet and a CA jet to be a jet that has been reconstructed by its respective algorithm.

Just like in subsection 2.2, I will first compare the position and energies of the reconstruced jets to the initial quarks. The distribution of $\Delta\phi$ between anti-k_T, k_T and CA jets and the inital quarks is very similar to that of the SC jets, which is shown in Figure 5. This shows that all of the algorithms peform well in reconstructing the direction of the jets.

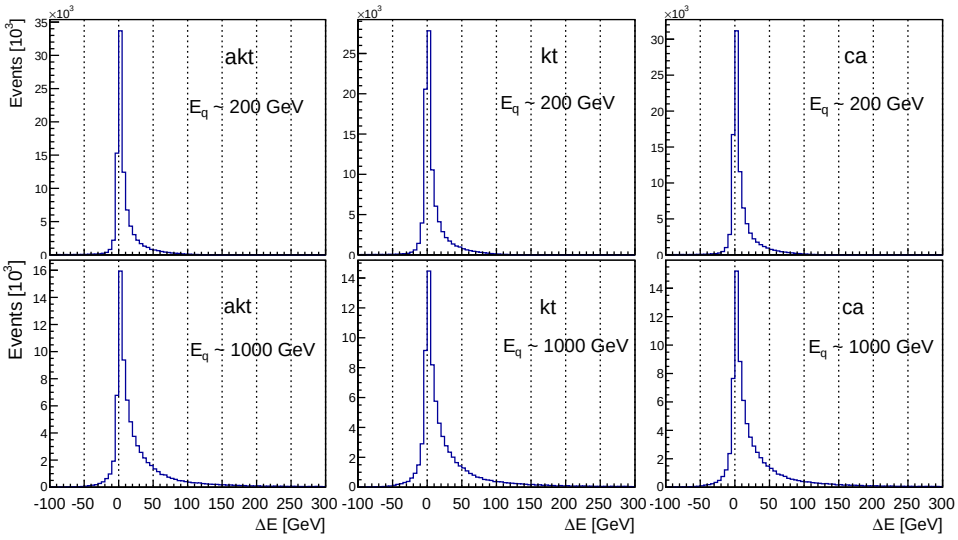

Figure 8: Histograms to show the difference in E between the quark and the reconstructed jet for different quark energies. The algorithms used were anti-k_T (akt), k_T (kt) and Cambridge-Aachen (CA) algorithms, with R =1.

Figure 8 shows the difference in energy, ΔE, between the intial quark and anti-k_T, k_T and CA jets at 200 and 1000 GeV. This shows that the energy tagging peformance of the current algorithms are similar in comparison to each other, and improved, particularly at high energy, in comparison to the SC jet algorithm, which is shown in Figure 6. One also should note that the current algorithms have considerably higher occurrence of the reconstructed jet energy being larger than the quark energy. The observation of the improved energy tagging peformance shows us that the current algorithms include more of the true jet constituents than the SC jet algorithm does, as they do not have the pruning effect of the SC jet algorithm. However, this also means that they are more susceptible to including background in their reconstructions, which is shown by the observation of the increased chance of $E_{jet} > E_q$.

The r_{95} and r_{60} for the anti-k_T, k_T and CA jets were then calculated using the same technique as used previously. Again the distributions of the individual events' r_{95} and r_{60} consist of a peak with a long tail; similar in shape to those shown in Figure 3. Hence the use of a fifth-order polynomial fitting to find the maxima is justified.

Figure 9 shows the r_{60} and r_{95} for anti-k_T, k_T and CA jets compared to the final state jet widths. The current algorithms reconstruct the jet widths very similarly to each other, which indicates that, for this case, the width of the reconstructed jets is determined by their positional properties, as opposed to their transverse momentum properties. For r_{60} [(a), (c) and (e)] we can see that the jet width is being constructed wider than the final state jet width for the the current algorithms when a larger value of R is used. This is in contrast to SC jet widths shown in Figure 7. This shows that, without the pruning property of the SC jet algorithm, higher R values will allow too much background energy to be included in the reconstruction of the jet for the current algorithms, forcing the reconstructed jets to be wider. Hence a lower value of R should be chosen, to help exclude background from the reconstruction. To optimise r_{60} width reconstruction for the current algorithms; a value of $R \sim 1.0$ should be chosen for jets with $E < 400$ GeV, and a width parameter of $R \sim 0.5$ should be chosen for jets with $E > 400$ GeV.

Figure 9 also shows that for the r_{95} jet widths, one must choose a large R value ($R \sim 2.0$) to closely reconstruct the width of the final state jet. However this is a very suprising result as Figure 3. of *Tseng and Evans* (2013) [2] shows that the current algorithms over-construct the mass of a jet for large R values. Hence, we would expect that for large values of R we would get artificially wide jets, as found for the r_{60} widths. The observation that large parameters are required would indicate that the final-state r_{95} may not be a good measure of the true jet width. The widenening to the final state jet's r_{95} could be due to the inital quark losing energy through gluon radiation, as outlined in the final paragraph of subsection 2.1.

One important area of further investigation required is the examination of the width properties of jets reconstructuted by the current algorithms combined with one of the common background subtraction techniques, such as pruning. This combination of algorithm followed by pruning is commonly used in jet studies [14] [15], hence it would be important to compare to the perfomance of the SC jet algorithm to these combinations.

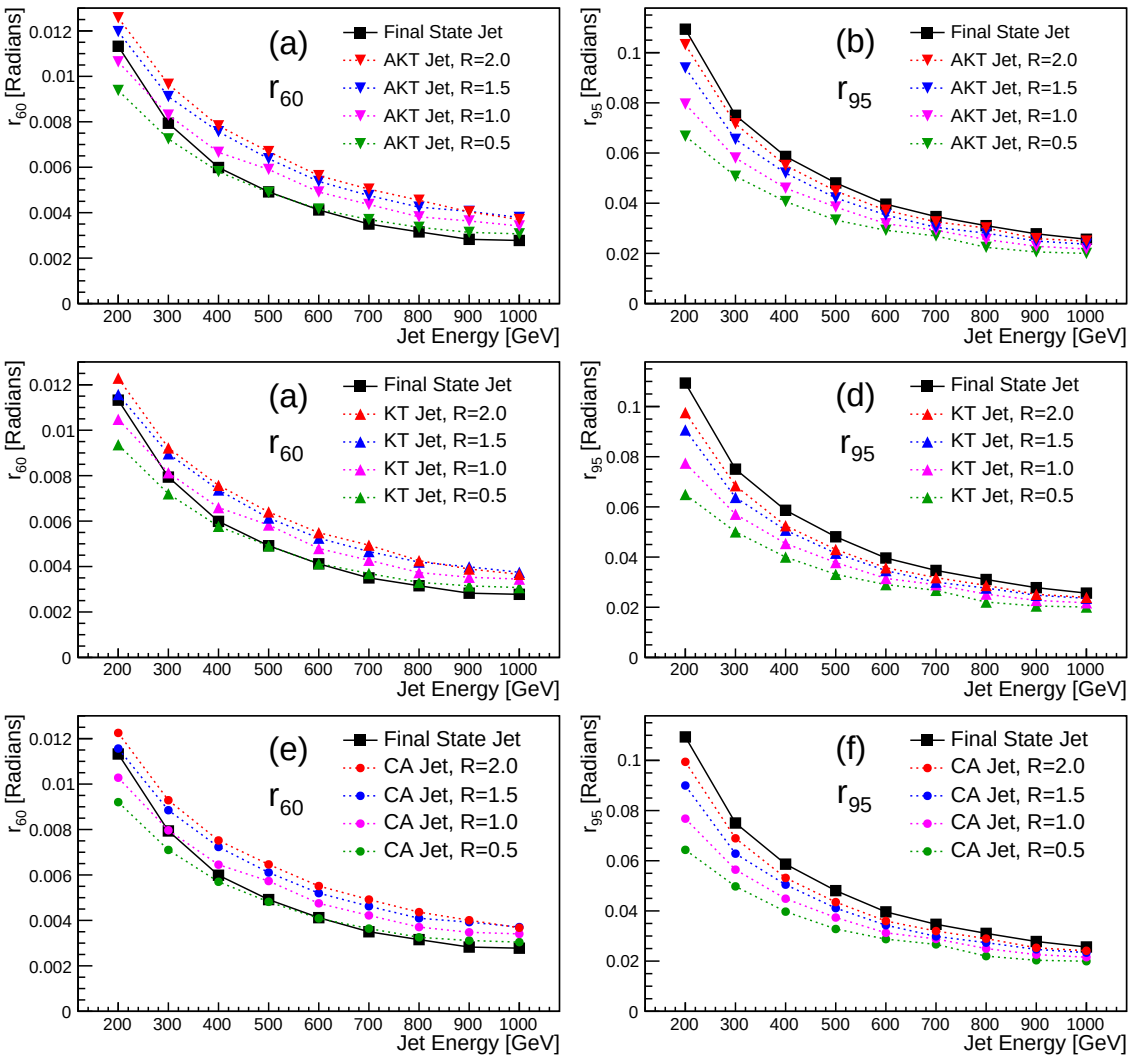

Figure 9: The jet widths, r_{60} and r_{95} respectively, for the final state jets compared to the jets reconstucted using the anti-k_T [(a) and (b)], k_T [(c) and (d)] and Cambridge-Aachen [(e) and (f)] jet algorithm with varying jet width parameter, R.

3 Conclusions

In this report I have investigated the reconstruction of jet widths by the Semi-Classical jet algorithm compared to the other sequential recombination algorithms currently in use for a range of values of jet width parameter, R. Monte-Carlo simulations of the event $Z' \to u + \bar{u}$ were used to carry out these investigations.

It has been shown that for the 60% jet width, r_{60}, the Semi-Classical jet algorithm is effective in reconstructing jet width when $R = 1.5 - 2.0$. In contrast, the anti-k_T, k_T and Cambridge-Aachen algorithms are effective in reconstructing jet width when $R = 0.5 - 1.0$.

Futher studies are required to investigate the width reconstruction on events with pile-up and more background from the underlying event. It is also required that the jet width reconstruction of the anti-k_T, k_T and Cambridge-Aachen algorithms with pruning applied is studied for a further comparison to the Semi-Classical jet algorithm.

Another possible area of study is events where two jets are produced from a boosted object, such as in the decay of a boosted W into two quarks. Currently an event like this is reconstructed by applying two small R sequential recombination algorithms for low energy events, to recontruct the two quark jets individualy; or one large R sequential recombination algorithm for the high energy cases, to reconstruct the two jets together. There is an intermediate regime where neither technique is effective at reconstructing the jets. It is possible that the Semi-Classical jet algorithm, as it reconstructs jet widths well at higher R values, may improve performance in this intermediate regime.

References

[1] Francis Halzen and Alan D. Martin. *Quarks & Leptons: An Introductory Course in Modern Particle Physics, p18-19*. John Wiley & Sons, 1984.

[2] Jeff Tseng and Hannah Evans. Sequential recombination algorithm for jet clustering and background subtraction. *Phys. Rev. D*, 88:014044, 2013.

[3] Gavin P. Salam. Towards jetography. 06 2009.

[4] Stephen D. Ellis, Christopher K. Vermilion, and Jonathan R. Walsh. Recombination algorithms and jet substructure: Pruning as a tool for heavy particle searches. *Phys.Rev.*, D80(094023), 12 2009.

[5] Marcela Carena, Alejandro Daleo, Bogdan A. Dobrescu, and Tim M.P. Tait. Z-prime gauge bosons at the tevatron. *Phys.Rev.*, D70(093009), 2004.

[6] Jens Erler, Paul Langacker, and Tian-jun Li. The Z - Z' mass hierarchy in a supersymmetric model with a secluded U(1) -prime breaking sector. *Phys.Rev.*, D66:015002, 2002.

[7] N. Arkani-Hamed, A.G. Cohen, E. Katz, and A.E. Nelson. The littlest higgs. *JHEP*, 2002(07):034, 2002.

[8] Torbjorn Sjostrand, Stephen Mrenna, and Peter Skands. Pythia 6.4 physics and manual. *JHEP*, 0605:026, 2006.

[9] Andy Buckley, Jonathan Butterworth, David Grellscheid, Hendrik Hoeth, Leif Lonnblad, James Monk, Holger Schulz, and Frank Siegert. Rivet user manual. 03 2010.

[10] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. Fastjet user manual. 11 2011.

[11] Fons Rademakers and Rene Brun. Root: An object-oriented data analysis framework. *Linux J.*, 1998(51es), July 1998.

[12] B.Z. Kopeliovich, I.K. Potashnikova, and Ivan Schmidt. Jet quenching: A fresh look at the energy loss scenario. *The Open Nuclear & Particle Physics Journal*, 3:16–21, 10 2010.

[13] V. Daniel Elvira. The challenge of pileup, http://indico.cern.ch/event/199123/material/slides/0.pdf.

[14] ATLAS Collaboration. Performance of large-R jets and jet substructure reconstruction with the ATLAS detector. *ATLAS-CONF-2012-065*, 2012.

[15] CMS Collaboration. Studies of jet mass in dijet and w/z + jet events. *JHEP*, 2013(90), 03 2013.