Review of jet reconstruction algorithms

Ryan Atkin

School of Chemistry and Physics, University of Kwa-Zulu Natal, Westville campus, Private Bag X54001, Durban 4000, South Africa

E-mail: rjatkin93@gmail.com

Abstract. Accurate jet reconstruction is necessary for understanding the link between the unobserved partons and the jets of observed collimated colourless particles the partons hadronise into. Understanding this link sheds light on the properties of these partons. A review of various common jet algorithms is presented, namely the K_t, Anti-K_t, Cambridge/Aachen, Iterative cones and the SIScone, highlighting their strengths and weaknesses. If one is interested in studying jets, the Anti-K_t algorithm is the best choice, however if ones interest is in the jet substructures then the Cambridge/Aachen algorithm would be the best option.

1. Introduction

A jet can be defined as a collimated spray of stable particles arising from the fragmentation and hadronisation of a parton(quark or gluon) after a collision. Jet reconstruction algorithms are used to combine the calorimetry and tracking information to define jets. The jets provide a link between the observed colourless stable particles and the underlying physics at the partonic level. A basic illustration of a collision of two protons, the subsequent particle shower and a reconstructed jet is shown in Fig.1. This link provides information on the kinematics of the originating partons which can be used to shed light on quantum chromodynamics(QCD) and infer the presence and thus the properties of the Higgs boson and other particles too short lived to be detected. An accurate jet algorithm will also be able to calculate the correct amount of missing energy in the detector, which could allude to new physics beyond the standard model if found not to match theory[1].

2. Important Aspects

Some aspects of an algorithm that need to be considered are the jet size and whether the algorithm is infra-red and collinear(IRC) safe. The jet size and area determine the susceptibility of a jet to soft radiation. A larger jet radius is important as it allows the jet to capture enough of the hadronised particles for the accurate calculation of the jets mass and energy. However a smaller jet radius is useful in reducing the amount of the underlying event(UE) and pile-up(PU) captured by the jet, preventing the overestimation of the jets mass and energy.

The splitting of a hard particle, while using a collinear unsafe algorithm, will result in the altering of the number and contents of the jets. A similar problem arises when a soft gluon is added to the system while an infra-red unsafe algorithm is in use. An IRC unsafe algorithm will affect the perturbative QCD calculations.

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

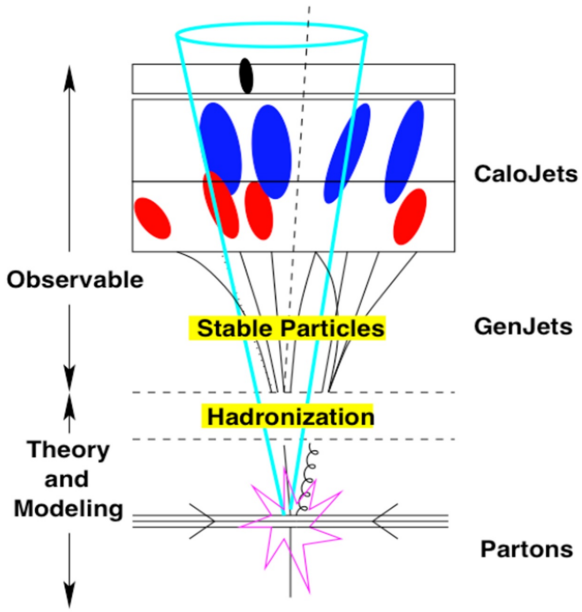

Figure 1. A simple example of an event showing the point of collision, the fragmentation and hadronization of the quarks and gluons and the resulting jet found through the detection of the stable particles[2]. Calojets are those jets created using the calorimeter output whereas Genjets are jets created using stable simulated particles. The dashed line represents the direction of the missing energy.

3. Jet Algorithms

There are two main classes of jet algorithms in use. The first being the cone algorithms, of which the most important are the iterative cone with progressive removal(IC-PR)(iterative version of [3]), the iterative cone with split-merge procedure(IC-SM)[4] and the seedless infra-red safe cone(SIScone)[5]. The second class is the sequential clustering algorithms which comprises of the K_t[6], Anti-K_t[7] and the Cambridge/Aachen[8] algorithms.

3.1. Cone Algorithms

Cone algorithms assume that particles in jets will show up in conical regions and thus they cluster based on (η-ϕ) space, resulting in jets with rigid circular boundaries. Cone algorithms in the past were preferred by experimentalists as they were easier to implement, however they are not favoured by theorists as they contain non-physical constants. Cone algorithms are generally IRC unsafe as well.

3.1.1. IC-PR: The iterative cone algorithm with progressive removal is a collinear unsafe algorithm. The CMS iterative cone and Pythia cone are examples of IC-PR. Its method of clustering is as follows:
Find the hardest(largest p_t) cell and make it a seed. Create a cone of radius R around this seed and calculate the trial jet axis by summing up the cells within this cone using four-vectors. If the trial jet axis is equal to the seed axis, the cone is labelled as stable and all the particles within the stable cone are removed from the list of particles. The next hardest remaining cell is found and this procedure is repeated. But if the trial jet axis does not equal the seed axis, the trial jet axis is made the new seed axis and the process is repeated until convergence of the

axes occurs. The entire process is repeated until there are no seeds left above a threshold energy E_{cut}.

3.1.2. IC-SM: The iterative cone algorithm with the split merge procedure is an infra-red unsafe algorithm. Examples include JetClu, midpoint cone and the ATLAS cone. Its method of jet formation is as follows:
First make all cells above a threshold energy E_{cut} a seed. Find all stable cones associated with those seeds using the same process as in the IC-PR algorithm but without removing any particles from the list once a stable cone is found. All stable cones found are labelled as protojets and a split merge procedure is run on the protojets once they have all been found. The split merge procedure is described in Appendix A.

3.1.3. SIScone: The seedless infra-red safe cone algorithm is the only IRC safe cone algorithm. Due to it's relatively smaller area, it is not badly affected by the UE and PU. It also has the ability to reach further than R for hard radiation and thus has good resolution, however this fact means that it is bad at resolving multijets. The following is the SIScone's method of jet finding:
For particle i
  Find all particles j within distance $2R$ of i
  If there is no particle j
    i is labelled a stable cone and added to the list of protojets
  Else
    Find both circles created by i and j lying on their circumference and calculate the momenta of the cones they define.
    For each circle
      Consider all four permutations of the two edge points being contained in or out of the circle. These four circles are labelled as current cones.
      For each current cone not previously found
        Determine whether the current cone's in or out status of the edge particles is the same as the cone defined by the momentum of the particles within the aforementioned current cone. If not, label the current cone as unstable.
Perform an explicit stability check on all current cones not labelled as unstable and add all stable cones to the list of protojets.
Run a split merge procedure on the protojets.

3.2. Sequential Clustering Algorithms
Sequential clustering algorithms assume that particles within jets will have small differences in transverse momenta and thus groups particles based on momentum space, resulting in jets that have fluctuating areas in (η-ϕ) space. Sequential clustering algorithms have always been favoured by theorists but not by experimentalists as in the past they had slow computational performance. However since the introduction of the FastJet program[9], clustering algorithms are much faster and preferred by experimentalists as well. Sequential clustering algorithms are also IRC safe.
All sequential clustering algorithms have a similar method. The first distance variable is the one between two particles $d_{ij} = min(p_{ti}^a, p_{tj}^a) \times \frac{R_{ij}^2}{R}$, where a is an exponent corresponding to a particular clustering algorithm, $R_{ij}^2 = (\eta_i - \eta_j)^2 + (\phi_i - \phi_j)^2$ is the (η-ϕ) space distance between the two particles and R is the radius parameter which determines the final size of the jet and is usually between 0.4 - 0.7. The second distance variable is $d_{iB} = p_{ti}^a$ and is the momentum space distance between the beam axis and the detected particle.

The sequential clustering algorithms work by first finding the minimum of the entire set $\{d_{ij}, d_{iB}\}$. If d_{ij} is the minimum then particles i and j are combined into one particle (ij) using summation of four-vectors after which i and j are removed from the list of particles. If d_{iB} is the minimum, i is labelled a final jet and removed from the list of particles. This process is repeated until either all particles are part of a jet with the distance between the jet axes R_{ij} greater than R, which is inclusive clustering. Or until a desired amount of jets have been found, this is exclusive clustering.

3.2.1. K_t: The a value corresponding to the K_t algorithm is 2, resulting in the following equations:

$$d_{ij} = min(p_{ti}^2, p_{tj}^2) \times \frac{R_{ij}^2}{R} \tag{1}$$

$$d_{iB} = p_{ti}^2 \tag{2}$$

The dominance of low p_t is shown in Eq. (1) and so the K_t algorithm prefers to cluster soft particles first, resulting in an area that fluctuates considerably and an algorithm that is susceptible to the UE and PU. Due to its method of clustering, K_t does a good job at resolving subjets.

3.2.2. Anti-K_t: The a value corresponding to the Anti-K_t algorithm is -2, resulting in the following equations:

$$d_{ij} = min\left(\frac{1}{p_{ti}^2}, \frac{1}{p_{tj}^2}\right) \times \frac{R_{ij}^2}{R} \tag{3}$$

$$d_{iB} = \frac{1}{p_{ti}^2} \tag{4}$$

So Eq. (3) is dominated by high p_t and the algorithm prefers to cluster hard particles first. Thus the area only fluctuates slightly and the algorithm is only slightly susceptible to the UE and PU. The Anti-K_t's clustering preference results in a algorithm that is the best at resolving jets but due to its poor de-clustering, it is the worst for studying jet substructure.

3.2.3. Cambridge/Aachen: The a value corresponding to the C/A algorithm is 0, resulting in the following equations:

$$d_{ij} = \frac{R_{ij}^2}{R} \tag{5}$$

$$d_{iB} = 1 \tag{6}$$

Both of the distance variables are independent of momentum and so its area fluctuates somewhat and is somewhat susceptible to the UE and PU. An example of the four main algorithms' jet areas are illustrated in Fig. 2. Due to the purely spatial character of the distance variables, C/A de-clusters the best and so is the best suited for studying jet substructure, although it is slightly more complicated to de-cluster than the K_t algorithm.

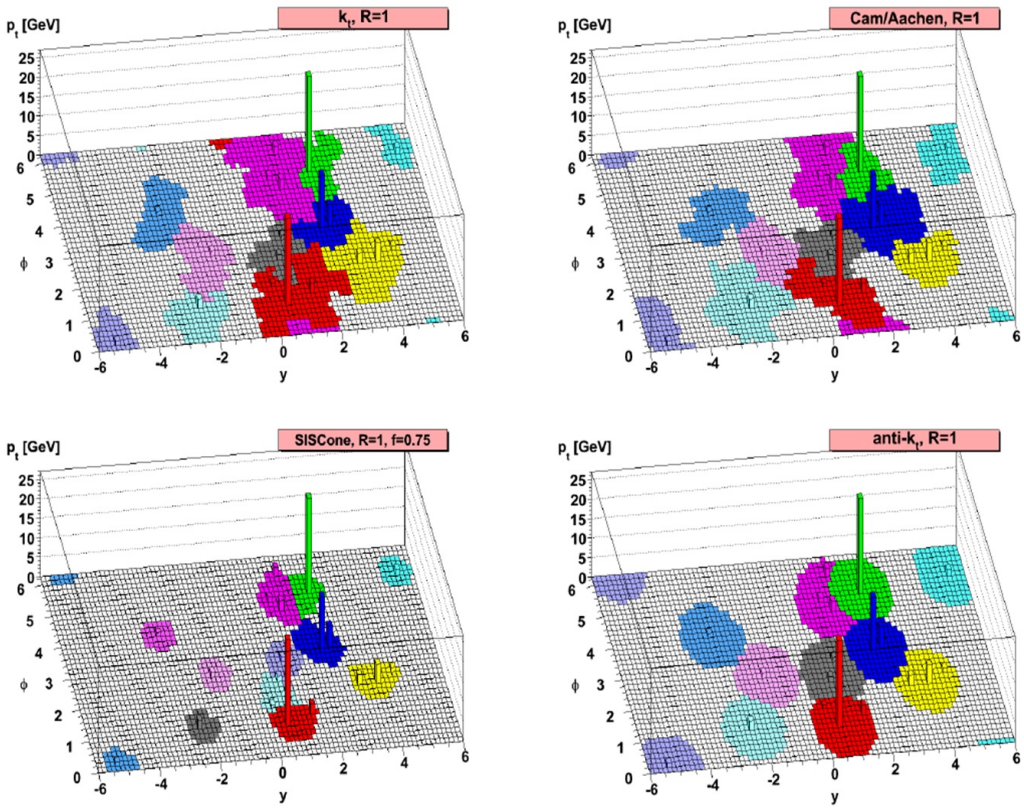

Figure 2. The four main jet reconstruction algorithms' areas, performed on the same data with the same input radius[7]. Noted features are the high irregularity in the the K_t algorithms area, the conical shape of the Anti-K_t's jets illustrating this algorithms preference for hard radiation and the smaller effective radius of the SIScone, due to the split merge procedure, which can be observed via smaller jet areas and two jets being resolved in the place of just the one grey jet. The different colours are used to represent the different jets and their areas.

4. Conclusion

When using jet reconstruction algorithms, two important things to consider are infra-red and collinear safety. Due too this the iterative cone with split merge procedure and the iterative cone with progressive removal are no longer widely used, as they are not both infra-red and collinear safe. Other important aspects to consider are jet size and shape. A large enough jet is necessary to capture the required amount of non-perturbatively hadronised particles while not being too large as to capture an excessive amount of the underlying event and pile-up.

The most accurate jet algorithm for resolving jets is the Anti-K_t algorithm whereas for the study of jet substructure, the Cambridge/Aachen algorithm is best suited. Accurate jet reconstruction is the only way we are able to study the properties of quarks and gluons as they are unable to be directly observed.

Acknowledgments
I would like to thank my supervisor Dr. Sahal Yacoob for his important input and assistance, Prof. Bruce Mellado and Prof. Alan Cornell of the University of the Witwatersrand for organising the High Energy Particle Physics workshop held at iThemba labs in Gauteng and finally the NRF for funding my accommodation and transport during my stay at the workshop.

Appendix A - Split merge procedure

The following is the method used in the split merge procedure[5]:
Remove all protojets with $p_t < p_{t,cut}$ (Helps reduce effects of pile-up)
Repeat
  Find the hardest protojet i through scalar sum of constituents' p_t
    Find the next hardest protojet j that shares particles with i
    If there is no overlapping protojet j
      i is labelled a final jet and removed from the list of particles
    Else
      If $p_{t,shared} < f \times p_{tj}$
        Share the particles according to which axis they are closer to and recalculate the protojets' momenta
      Else
        Merge protojets i and j into one and remove i and j from the list of protojets
      If either of these two processes produces another protojet matching one already found, keep both as two distinct protojets
Until (no protojets left)

The merging parameter f is usually between the values of 0.5 and 0.75 and determines the size of the jets after the split merge procedure.

References

[1] M. Hodgkinson, in Proceedings of the 34th International Conference in High Energy Physics (ICHEP08), Philadelphia, 2008, eConf C080730 [arXiv:hep-ex/0810.0181]
[2] P. Schieferdecker. Available at: <https://twiki.cern.ch/twiki/bin/viewfile/Sandbox/Lecture?rev=1;filename =Philipp_Schieferdeckers_Lecture.pdf> [Accessed 17 December 2015]
[3] G. Arnison et al. [UA1 Collaboration], Phys. Lett. B 132 (1983) 214.
[4] G. C. Blazey, J. R. Dittmann, S. D. Ellis, V. D. Elvira, K. Frame, S. Grinstein, R. Hirosky and R. Piegaia *et al.* [arXiv:hep-ex/0005012]
[5] G. P. Salam and G. Soyez, JHEP **0705** (2007) 086 [arXiv:0704.0292 [hep-ph]].
[6] S. Catani, Y.L. Dokshitzer, M.H. Seymour, and B.R. Webber. Longitudinally invariant Kt clustering algorithms for hadron hadron collisions. Nucl.Phys., B406:187224, 1993.
[7] M. Cacciari, G.P. Salam, and G. Soyez. JHEP **0804** (2008) 063 [arXiv:0802.1189 [hep-ph]].
[8] CMS collaboration. A Cambridge-Aachen(C-A) based jet algorithm for boosted top-jet tagging. CMS PAS JME-09-001, 2009
[9] M. Cacciari, G.P. Salam, and G. Soyez, Eur.Phys.J. C72 (2012) 1896 [arXiv:1111.6097 [hep-ph]]