2005 International Linear Collider Workshop - Stanford, U.S.A.

Application of Layer-by-layer Clustering to a Generalised Calorimeter

C. G. Ainsley
Cavendish Laboratory, Madingley Road, Cambridge, CB3 0HE, U.K.

An algorithm for cluster reconstruction in a highly granular, layered calorimeter is presented. Its general applicability to arbitrary detector geometries, comprising a rotationally-symmetric barrel closed at either end by an endcap, is demonstrated.

1. INTRODUCTION

Excellent jet energy resolution is a mandatory requirement of the detector for the International Linear Collider (ILC) [1]. To achieve this, studies have demonstrated that the calorimeter must be highly granular, comprising a pixelated, layered structure, so that the individual particles within jets can themselves be resolved. Such imaging capability in the calorimetry is unprecedented and warrants the investigation of novel clustering techniques. Moreover, it is desirable to devise algorithms for calorimeter clustering that have minimal reliance on geometry, enabling detector configurations to be optimised and objective comparisons between competing designs to be made straightforwardly. The clustering algorithm reported here has been developed with these needs in mind.

2. THE ALGORITHM

The algorithm consists of three stages, described in turn below.

2.1. Stage 1

The first objective is to establish the cluster cores. This is achieved by exploiting the high granularity of the active cells to track closely-related hits layer-by-layer outwards through the calorimeters. The procedure is illustrated in Figure 1.

1. For each hit in a given layer, l, of the calorimeter, the distance d is minimised with respect to all hits in layer $l-1$. For the hit that yields the minimum:
   - if $d<$`distMax` (where `distMax` is a tunable parameter of the algorithm), the hit in layer l is added to the cluster already containing the hit in layer $l-1$;
   - if $d\geq$`distMax`, the procedure is repeated with all hits in layer $l-2$, then, if necessary, those in layer $l-3$, and so on, culminating with those in layer $l-$`layersToTrackBack` (a tunable parameter of the algorithm), if the condition still cannot be satisfied.
2. New clusters seeds are formed from groups of nearby hits in layer l that have not been associated with clusters in previous layers by step (1); hits within `proxSeedMax` (a tunable parameter of the algorithm) of the remaining unassociated hit of highest local hit density are grouped into the same seed.
3. A direction cosine is assigned to each hit in layer l. To reflect the different stages of shower development, the method employed depends upon whether the layer lies in the electromagnetic (Ecal) or hadronic (Hcal) calorimeter:

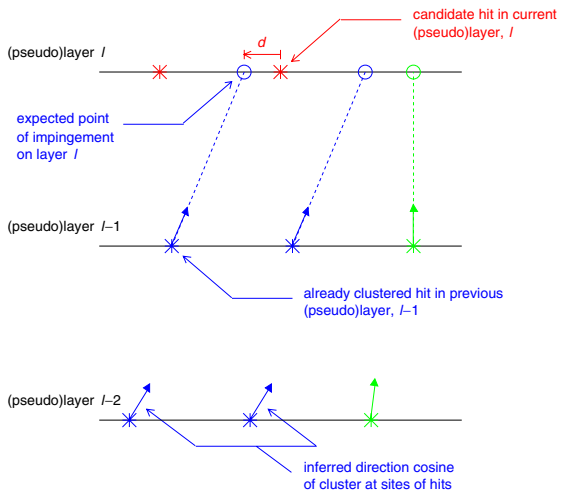

Figure 1: Schematic of the first stage of the clustering algorithm. Two clusters have so far been reconstructed in the vicinity: one comprising two hits in layer $l-1$ and two in layer $l-2$; the other comprising one hit in each of layers $l-1$ and $l-2$. An attempt is made to attach the candidate hit in layer l to one or other of these clusters continuing from an earlier layer.

- in the Ecal, the mean position of the hits belonging to each cluster in layer l is calculated and direction cosines for every hit in a particular cluster are aligned along the displacement of the cluster's mean position in layer l from either its mean position in its seed layer, or (0,0,0) if l is the seed layer;
- in the Hcal, direction cosines for each hit are aligned along the hit's displacement from either the hit in a previous layer to which it is linked, or (0,0,0) if l is the seed layer.

4. Steps (1)–(3) are propagated through each layer of the Ecal and then, in turn, through each layer of the Hcal.

2.2. Stage 2

Owing to the layer-by-layer outward-going nature of the first stage, particles which produce backward-spiralling tracks in the calorimeters will likely be reconstructed as two forward-going clusters broken at the apex of the spiral, as suggested by Figure 2. The second stage attempts to unite such fragmented clusters, using their relative proximity and directional properties.

1. For each hit in the terminating layer, l, of a cluster, the distance, p, to each hit in neighbouring clusters in that layer is calculated, as is the angle, γ, between their direction cosines:
   - if, for one or more pairs of hits, both p<`proxMergeMax` (a tunable parameter of the algorithm) and $\cos\gamma$<`cosGammaMax` (a tunable parameter of the algorithm) are satisfied, and all such pairs involve a hit in the same neighbouring cluster, the two clusters are merged into one;
   - if for more that one pair of hits, both p<`proxMergeMax` and $\cos\gamma$<`cosGammaMax` are satisfied, and successful pairings can be made with more than one neighbouring cluster, the neighbouring cluster with the earliest seed layer is deemed to be the progenitor of the terminating cluster and these two clusters are merged into one;
   - if both p <`proxMergeMax` and $\cos\gamma$ <`cosGammaMax` are not satisfied for any pair of hits, the terminating cluster is deemed to be a cluster in its own right.
2. Step (1) is repeated for all clusters found after the first stage.

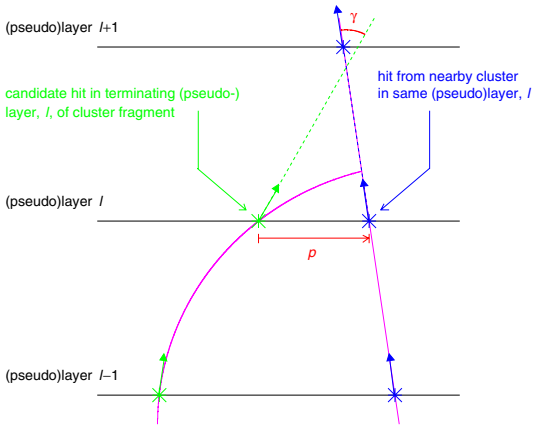

Figure 2: Schematic of the second stage of the clustering algorithm. An outward-going particle on the right gives rise to a backward-travelling secondary, which registers hits in layers l and $l-1$. An attempt is made to reunite the two forward-going clusters that this reconstructs as during the first stage.

2.3. Stage 3

If the clustering cuts in the first stage are set with aggression sufficient to resolve cluster cores of high proximity, the hits in the outer-lying parts of the showers will likely just fail to pass these cuts—the result being a halo of low multiplicity clusters around the principal cluster cores. The third stage attempts to associate these low multiplicity clusters with the appropriate core clusters, as shown in Figure 3.

1. For the hit of highest local hit density in the seed layer, l, of a cluster whose hit multiplicity is smaller than `clusterSizeMin` (a tunable parameter of the algorithm), the angle β is minimised with respect to all hits in layer $l-1$. For the hit that yields the minimum:

   - if $\tan\beta<$`tanBetaMax` (a tunable parameter of the algorithm), the low multiplicity cluster is merged with the cluster containing this hit in layer $l-1$;
   - if $\tan\beta\geq$`tanBetaMax`, the procedure is repeated with all hits in layer $l-2$, then, if necessary, those in layer $l-3$, and so on, culminating with those in layer $l-$`layersToTrackBack` (a tunable parameter of the algorithm) if the condition still cannot be satisfied.

2. If $\tan\beta\geq$`tanBetaMax` in all layers, step (1) is repeated with the hit of second highest local hit density in layer l of the low multiplicity cluster, and so on, terminating when $\tan\beta<$`tanBetaMax` is satisfied.

3. If $\tan\beta\geq$`tanBetaMax` for all pairings of hits in layer l of the low multiplicity cluster and hits in previous layers of all other clusters, the distance, s, between all pairings of hits in layer l of the low multiplicity cluster and hits in neighbouring clusters in that layer is minimised:

   - if $s<$`proxMergeMax` (a tunable parameter of the algorithm) for the minimum, the low multiplicity cluster and its nearest neighbouring cluster are merged into one;
   - if $s\geq$`proxMergeMax` for all pairings, the low multiplicity cluster is deemed to be a cluster in its own right.

4. Steps (1)–(3) are repeated for all clusters found after the second stage.

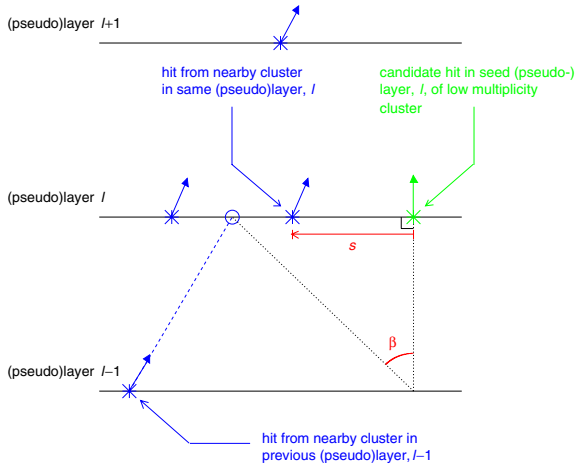

Figure 3: Schematic of the third stage of the clustering algorithm. An attempt is made to reunite a low multiplicity cluster on the right with the principal cluster core to which it most likely belongs.

3. GENERALISING THE CALORIMETER

Using as its basis the layer indices of the active cells in the detector, the algorithm implicitly requires these indices to vary continuously throughout the calorimeter. While this is the case within either one particular stave[1] of the barrel or an endcap, this is not the case at boundaries between staves, if by 'layers' the physical layers of active material in the detector are understood. This is exemplified by the left-hand sides of Figures 4 & 5. A particle crossing the boundary from one stave to another could not be tracked as a continuous cluster by the algorithm.

If, instead, *pseudolayers* are constructed by projectively intersecting the active layers in adjacent staves, and by 'layers', these pseudolayers are understood, then the index varies continuously throughout—a particle crossing a *pseudostave*[2] boundary can be tracked with no such problem. This is exemplified by the right-hand sides of Figures 4 & 5. Moreover, in this manner, any calorimeter geometry can be recast into a standard, generalised form, comprising layered shells of rotationally symmetric polygonal prisms, coaxial with the z-axis, independent of its detailed structure. To compute the pseudolayer indices of the hits, the only necessary inputs are the rotational degree of symmetry of the barrel, the azimuthal offset of the (nominal) first barrel stave with respect to the x-axis, and arrays specifying the perpendicular distances of the barrel layers from the z-axis and the endcap layers from the z=0 plane. Inasmuch as these are regarded as parameters of the algorithm, the cluster reconstruction is geometry-independent, automatically translating the layers on which the calorimeter is built into the pseudolayers with which the algorithm works.

4. IMPLEMENTATION

The algorithm has been coded in C++, is compliant with LCIO (v01-05) [2] and is structured as a series of MARLIN (v00-08) [3] processors. The various geometrical parameters and clustering cuts are set at run-time in an accompanying steering file. LCIO hit objects are taken as input and LCIO cluster objects are returned as output.

[1] A stave is regarded here as a plane of parallel layers.
[2] By analogy, a pseudostave is regarded as a plane of parallel pseudolayers.

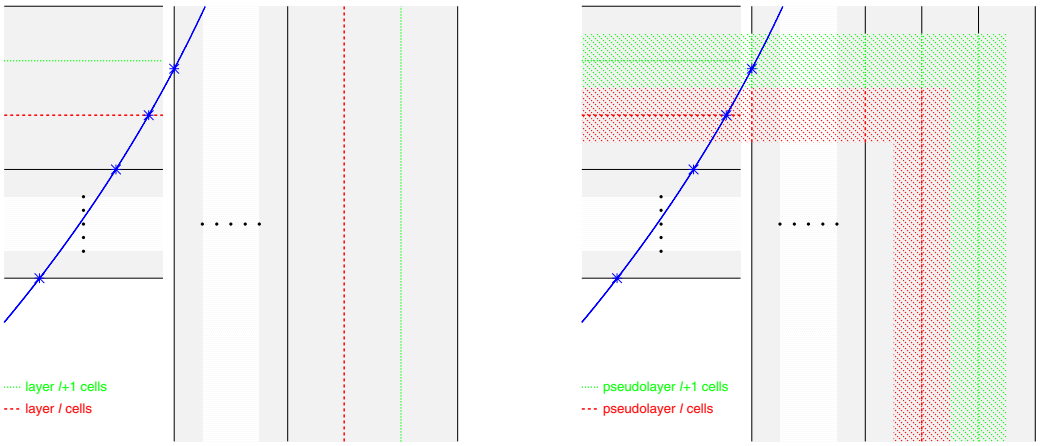

Figure 4: The barrel/endcap overlap region. On the left, active cells in layers l (dashed line) and l+1 (dotted line) are highlighted. Cluster discontinuity occurs when a particle exits the barrel at layer l and enters the endcap at layer 1. On the right, active cells in pseudolayers l (dashed line) and l+1 (dotted line) are highlighted. Cluster continuity is now restored: the particle exits the barrel at pseudolayer l and enters the endcap at pseudolayer l+1.

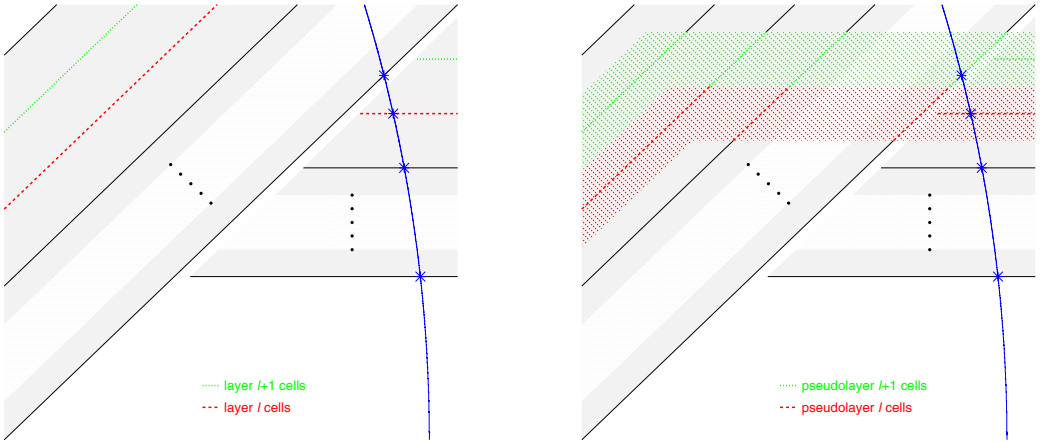

Figure 5: The overlap region between two adjacent barrel staves. On the left, active cells in layers l (dashed line) and l+1 (dotted line) are highlighted. Cluster discontinuity occurs when a particle exits one stave at layer l and enters the neighbouring stave at layer 1. On the right, active cells in pseudolayers l (dashed line) and l+1 (dotted line) are highlighted. Cluster continuity is now restored: the particle exits one stave at pseudolayer l and enters the neighbouring stave at pseudolayer l+1.

5. APPLICATION

Figure 6 shows the application of the algorithm to a hadronic Z decay event at 91 GeV in the MOKKA (v04-00) [4] simulated CALICE [5] calorimeters. The high degree of correlation between hits assigned to the same reconstructed cluster and hits belonging to the same true cluster (*i.e.* those hits attributable to the same incident particle)—and not only for *intra*-stave clusters, but also for *inter*-stave clusters—demonstrates the effectiveness of the algorithm.

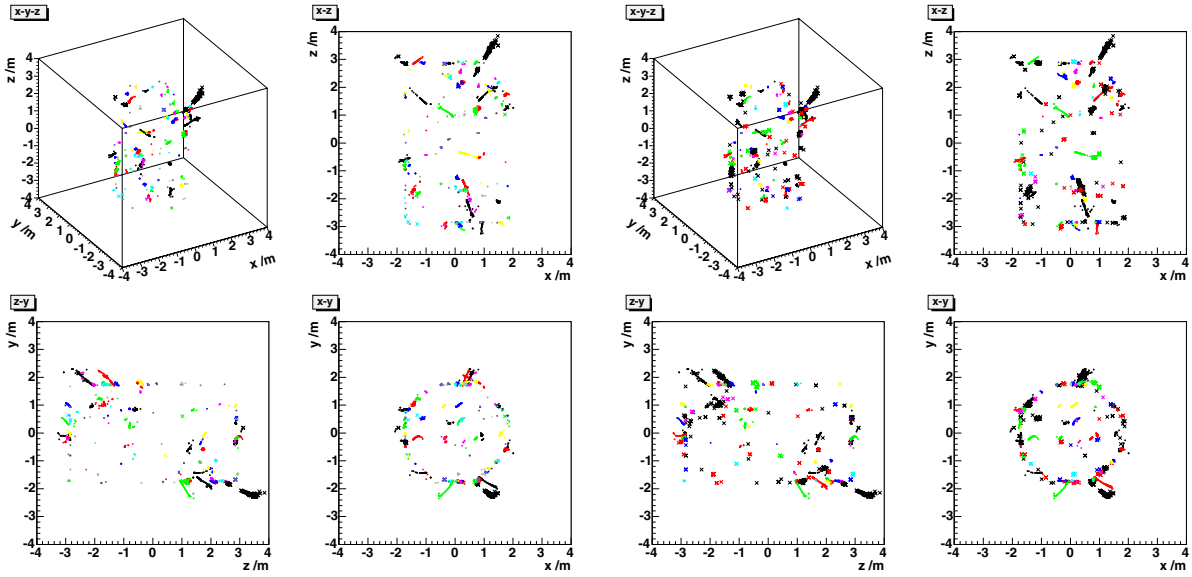

Figure 6: Reconstructed (left) and true (right) clusters for a simulated Z decay at 91 GeV. Hits from the same cluster are plotted with the same colour. Those marked as dots are located in the same pseudostave as the majority of the cluster to which they belong; those marked as crosses are located in a different pseudostave.

6. CONCLUSION

A flexible, geometry-independent cluster-finding algorithm, honed for the highly granular calorimeters envisaged for a detector at the ILC has been presented. It has has been coded in the LCIO/MARLIN framework, compatible with other ILC software applications, and its release to the community at large as a tool for event reconstruction is impending.

Acknowledgments

This work is supported by a Postdoctoral Fellowship award from the Particle Physics and Astronomy Research Council of the United Kingdom.

References

[1] "TESLA Technical Design Report, Part IV: A Detector for TESLA", DESY 2001-011, ECFA 2001-209, March 2001.

[2] LCIO—a persistency framework for linear collider simulation studies: `http://lcio.desy.de` .

[3] MARLIN—Modular Analysis and Reconstruction for the LINear Collider: `http://ilcsoft.desy.de/marlin/` .

[4] MOKKA—a detailed Geant4 detector simulation for the Future Linear Collider: `http://polywww.in2p3.fr/geant4/tesla/www/mokka/mokka.html` .

[5] CALICE Collab., CAlorimeter for the LInear Collider with Electrons: `http://polywww.in2p3.fr/flc/calice.html` .