

2005-10-17 LCD Weekly

SM: Got sigma of 2.7 GeV at the Zpole after tuning up pflow algo.

(slides)

Photons from hmatrix.

Neutrals are leftovers.

Distribution is symmetric. (usually not) Fit with double gaussian.

Photon finder was not picking up the tail of the photons. Wanted to modify fixed cone to use instead.

To find photons, just allow NN to skip some layers.

MC: Not really an 80/20% split on the 2 gaussians.

RC: What were parameters for finding photons using NN?

SM: Looking...

MC: Which detector?

SM: cdcaug05, because RPCs are difficult to work with for this application.

SM: Distance 2 & 2 in transverse dimension. ~Moliere radius. Number of layers = 5.

No hits in tail, because missing hits a few layers away.

Min # cells = 8.

10 cell clusters into hmatrix and used for photons.

Ratio of $E_{\text{cluster}}/E_{\text{actual}}$. Apply 1.04 E correction.

RC: For neutrals, did do clustering, not just take all the hits and use their energy?

SM: Yes, use the cluster energy. Calculate the energy by using a different sampling fraction in the ecal. Similar to hits vs. energy as done by RC. Number of hits sampling fractions for ecal hits, e.g. GeV/hit.

Allow to go across ~20% of hcal to find clusters.

For ecal, use 7 layers for the hadrons. Reclustered with different parameters after removing photon clusters and MIP hits.

NG: Work on cheater from MR?

MR: Amazed at SM's results with Z jets.

Developing cheater using tracking and cluster cheater for perfect event reconstruction.

Switches to allow mixin of reconstructed particles with MCParticles.

Reconstruction for W's at 500 GeV with mass resolution ~3.7 GeV. Worse than SM's results with actual clusters.

NG: What jet finding?

MR: JadeE.

NG: Floating kT to get cuts? (???)

MR: Yes.

Useful for reconstructing jets.

Few options for energy measures. In hcal, parameterized or digital/analog calorimeter.

Confluence page.

Maybe compare with results SM is getting.

SM: "Cheating" means cheating everything?

MR: Yes.

SM: This is the blue curve in the plot.

Cheating on tracks.

Makes reconstructed particles -> jets.

Needs to be tuned.

NG: Updates from RC?

RC: No writeup, yet.

Compare clusterers and show "how you're doing". Close to a general package.

Hopefully, next week will have a writeup.

SM has parameters for NN clusterer. Would like to see if get these results.

Compare with fixed cone for energy and # of hits. (purity and efficiency)

SM: Can help decide which clusterers to use for which applications.

RC: Efficiency and purity -> it is always a tradeoff.

NG: Update from MC?

MC: Porting pflow code from hep.lcd. Now have most of it ported. Keeping in the contrib area, because it is not yet complete/finalized.

Added some "quick and dirty" code to get some of the geometrical information.

NG: What geo information is needed?

MC: What geometric element is a SpacePoint inside?

Solveable if subdetectors are cylindrical.

Haven't been able how to get a specific cell.

Discussing with Niels about missing features. EM not treated properly. How to combine EM and HAD code from myself and Niels.

RC: Niels stuff is not in?

MC: MST and MIP finder are more-or-less stable but not in main repository, yet.

RC: Could use MST from contrib area for benchmarking?

MC: Yes.

NG: AFA photons, might want to look at NN and fixed cone?

Probably going to be the optimal photon finder.

MC: Would like to be able to swap in one algorithm or another.

NG: Updates from NS?

NS: Working on fitter. Construction of the error matrix for the track, every point on the track f/ multiple scattering. Reading in thicknesses and radiation lengths to construct error matrix for the entire track -> weight matrix.

In the process, found a few problems with the helical swimmer, which should be resolved by TJ and JS.

JS did not count on negative radius for the helix. Bug with negative $\tan(\lambda)$.

Helix for negative tracks?

Convention for negative tracks, ω = negative. Set the track for a helical swimmer from the track parameters. Pass the radius to the helix. In the helix constructor, there is no ω , just radius. But need to keep this information.

Trying to derive track parameters from any point on the track. In the swimmer, track doesn't have to be on POCA. From any point, momentum and point coordinates -> helix. Track parameters are POCA to reference point or (0,0,0). I was using MCParticles which don't necessarily originate close to IP. Putting MCParticle in helix swimmer -> swim back to original point.

Need a few more functions from swimmer.

Obtain a trajectory from the swimmer. Access the trajectory functions. Helix and Line are public interfaces or should allow people to use them.

TJ: Helix says signed radius of curvature.

NS: Think I solved these problems. Didn't put in the CVS.

JS: Test cases. Tried to get point along trajectory with negative distance. But doesn't work with POCA. DOCA works, point -> trajectory. POCA along trajectory does not work.

MC: Trajectory needs a few more accessors. What is the direction on a given point of the trajectory?

NS: In the helix, there is `getMomentum()` at a point, which is direction. Just need access to the helix itself. Useful to have `getTrajectory()` from the swimmer.

MC: Committed a bunch of accessors for the line but marked as deprecated, because should go into main interface.

NS: Distance is always positive.

TJ: Always make origin of POCA at (0,0,0)?

TJ: Take closest point in particular direction.

RP: Would like to second NS's request for more functionality. Right now, can initialize a swimmer with a track but not track parameters.

Methods for getting not just first intersection but arbitrary intersections (2nd, 3rd, 4th, etc.). Specific track parameters as single object and get intersections.

TJ: NS suggesting something different. Always give the cylinder intersections from (0,0,0) point.

MC: Could imagine where wouldn't want to start near (0,0,0).

RP: First POCA along trajectory where there are hits. May or may not be close to (0,0,0). Find some hits -> make track. Should be DOCA defined by the seed.

NS: Can get all other points easily if agree to have POCA as origin of the track. Method `findPointAtLength()`. Increase length by one rotation and will find all other points.

RP: No doubt that it can be done in the code. Encapsulate the facilities in main interface.

TJ: Noticed in MC's code, used `ParallelLineException`, but should use `NaN`, instead.

NG: Any more updates from RP?

RP: Discussion last week about what classes should come out of track digitization. Good suggestions from NS. Assume we're going forward with this? Someone needs to figure out raw hit format, persistence.

TJ: Trying to make compatible with LCIO. Can't arbitrarily change time to ID. Somewhat constrained by compatibility with LCIO. Can change or "cheat".

NS: Waiting for TJ to define the `TrackerHit` interface. Right now, make `TrackerHit` implementation. Waiting for the final definition of what is a `TrackerHit`, e.g. `Hep3Vector` or array.

NG: Updates from JS?

JS: Trying to understand the helix problem, write tests to fix problem reported by NS. Might need help understanding the `HelixSwimmer`.

Still need a way to quantify the `ZvTop` performance, e.g. real position vs. found position, precision.

NG: `org.lcsim` forum has been added at [linearcollider forum](#).

TJ: GL suggested including user name in CVS email messages.

MR: Found link for describing to `lcd-dev` mailing list on Confluence.

NG: Can show CVS comments in Netbeans 5.

Questions to forum, announcements on `lcd-dev`.

JS: Have logging facility?

TJ: Java supports logging.

`getLogger()` from `Driver`.

`java.util.logging`

JS: Would like to organize debugging printouts.

MR: Put in histograms to cheater code. Good idea?

TJ: `getHistogramLevel()` should determine whether hists are filled.

NS: Encountered case in geometry of `sidaug05` where layers in reverse order of subdetector. Always order radius in increasing radius, z. In `VertexBarrelSupport`, foam layer is layer 0 and aluminum layer is layer 1 and has smaller r.

TJ: On fixed cone clusterer, RC added some methods, but names were kind of unclear.

RC: Just temporary until agree on common interface.