Level3 Z Tracking Algorithm

Jacob Anderson
Department of Physics and Astronomy
Brigham Young University, Provo, Utah, 84602

Abstract

Level3 as part of the CLEO III detector has worked as a filter and classification system. Because of deterioration of the ϕ silicon portion the tracker portion of Level3 is inoperable. In order to regain the functionality of the r-ϕ tracker, I have been working on a z tracking algorithm to distinguish and filter events. Overcoming some of the limitations of the z silicon and beam, the initial z-tracker achieved an efficiency of 84% when applied to previous runs. This is below the target of efficiency of 90%, but with minor changes, this efficiency can hopefully be boosted to the goal.

Introduction

Level3 is a real time event filtering system for the CLEO III detector. The Level3 algorithm has relied on two different methods for distinguishing "interesting" events from "garbage", the CsI calorimeter crystals and an r-ϕ silicon tracking algorithm. Because of deterioration of the r-ϕ silicon the second method used by Level3 is no longer viable. To supplant some of its functionality, I have been working on a silicon tracker that works in the r-z plane. Although the z side silicon has not suffered the same deterioration as the ϕ side, the efficiency is still an issue. Other problems relating to detector readout design also complicate the z tracking. Using simple methods, I hoped to be able to filter the events that did not originate on the beam line and to reduce the number of events recorded to permanent storage.

Geometry and Solution Strategy

As can be seen in Fig. 1, the silicon portion of the CLEO III detector is arranged in four concentric cylinders of increasing length. Notice in Fig. 1 that the silicon layers are divided into 61 flat sections called ladders. Each ladder read out at both ends so that each sensor wafer is read out by either the east or west hybrid. The number of wafers read out by each hybrid can be as few as 1 and up to 5. Each wafer is identical with ϕ measuring sensor strips on one side and z measuring strips on the other. The number of wafers per dimension per layer is indicated in Fig. 1.

Reading out the data from the silicon wafers is done from the ends of each hybrid. This facilitates reading ϕ data because each phi channel is separate and unique. The z strips are more difficult to read out. To keep the number of channels manageable, all of the corresponding z strips on each wafer in a hybrid are ganged together into a single channel. Because of the ganging, each hit channel could represent up to 5 different z positions in the hybrid, but it also makes the number of readout channels 512 for each hybrid regardless of its size. The resulting ambiguity makes it necessary to compare several layers against each other in order to resolve the tracks from the ghost hits and it also multiplies any noise in the system.

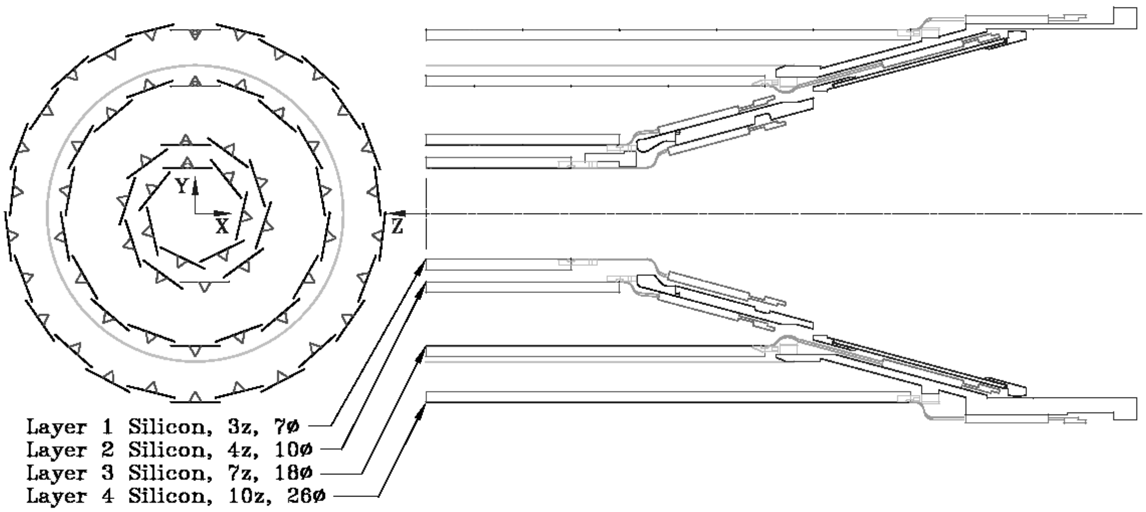

FIGURE 1. A view of the silicon detector in the r-ϕ and r-z planes

Tracks of “good” events ought to originate from the beam line. In r- ϕ the beam line is essentially a point, but in the z direction the beam interaction area is a macroscopic, roughly Gaussian distribution with a σ of 1.2 cm[1]. In order ensure that the tracker includes most of the events, I use 3σ in either direction, yielding a beam spot of 7.2 cm, a lot of area to look for tracks. The z-tracker does have an advantage over the r-ϕ tracker. The tracks in z should be straight lines not curved paths. Therefore, a “good” track is one that originates from the beam region and leaves a straight-line track in at least 3 of the silicon layers.

Algorithm Design

With a working definition of an “interesting” track, I began to design an algorithm to find the tracks. Several constraints were placed on the algorithm including: speed, memory footprint and portability. In addition to the computing constraints, I also the needed to follow the CLEO III coding standards, adhere to established Object-Oriented Programming (OOP) principles and work within the SUEZ data environment. I spent a considerable amount of time familiarizing myself with these last constraints. With all of the constraints in mind I began to devise the simplest algorithm that would accomplish the task.

One of the primary concerns is the volume of data that is available about each event. Because time is of prime importance, sifting through all of the data isn’t feasible. Level3 minimizes its checking by only using the information from the crystal calorimeters, the trigger layer and silicon. The z tracker goes further, using only the trigger layer and the silicon. Nonetheless, the z silicon comprises 62464 readout channels; minutely checking each one isn’t an option.

In order to reduce the bulk of the silicon data the z tracker employs a charge cut and then sparsifies the data. The charge cut for “normal” channels is a signal of 3σ above pedestal; “noisy” channels require 5σ. Sparsifying the silicon information into bins reduces the volume of the data. Currently the z-tracking algorithm groups 4 silicon strips into a single bin. This gives a bin pitch of just over 400 microns. Another feature of the binning is that some of the ϕ information is retained because each bin is local to a hybrid. This gross ϕ binning proves useful in the algorithm.

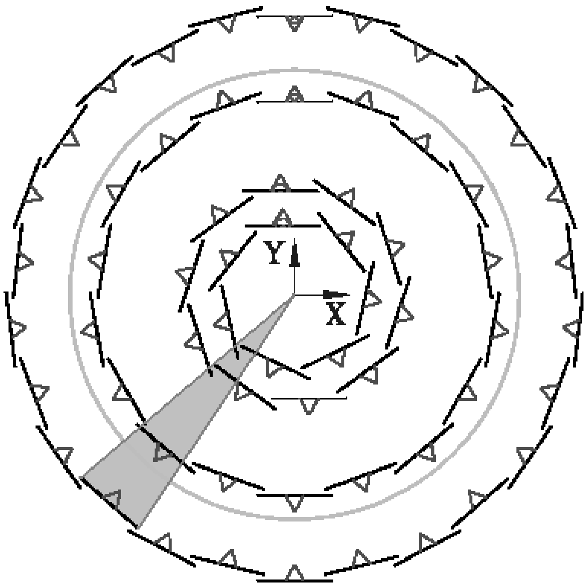

FIGURE 2. The r-ϕ slice taken by the z-tracker

Although the tracker is looking in the r-z plane, it uses information from the r-ϕ plane to narrow its search area. Simply, the algorithm takes a narrow pie slice in the r-ϕ silicon like the one in Fig. 2, eliminating a large portion of the detector with its hits and noise. It uses the slice to create a 2D representation of the z data where each ambiguous hit from the ganging is copied and represented by a unique coordinate. Finally, it searches the 2D representation for tracks emanating from the beam region.

The Level1 axial trigger identifies tracks in the drift chamber and reports them in 112 ϕ bins. The tracker uses the information from the trigger bins to identify the ladder from layer 4 to use as the basis for the r-ϕ pie region (Fig. 2). All of the ladders between this initial layer 4 ladder and the beam spot are added to the 2D r-z data representation. To know quickly what the layer 4 ladder and other inner ladders are the tracker employs a lookup table, which is initialized at the beginning of a run. This lookup associates each trigger bin with the appropriate layer 4 ladder based on their sharing of ϕ angle. The lookup also associates layer 4 ladders with ladders from each of the other three layers based on ϕ angle sharing.

After the lookup has been used to create the 2D representation, the search starts in layer 4. Each hit in layer 4 creates a triangular area, like the one in Fig. 3, where a track could pass and include the layer 4 hit. The tracker then begins to search the area in layer 3. If a hit is found in layer 3 then the two hits produce a single possible track. This track becomes a valid track by finding a hit along the same line in either layer 1, layer 2 or both. Because layer 3 might not generate a track, the tracker will use layers 2 and 4 to generate possible

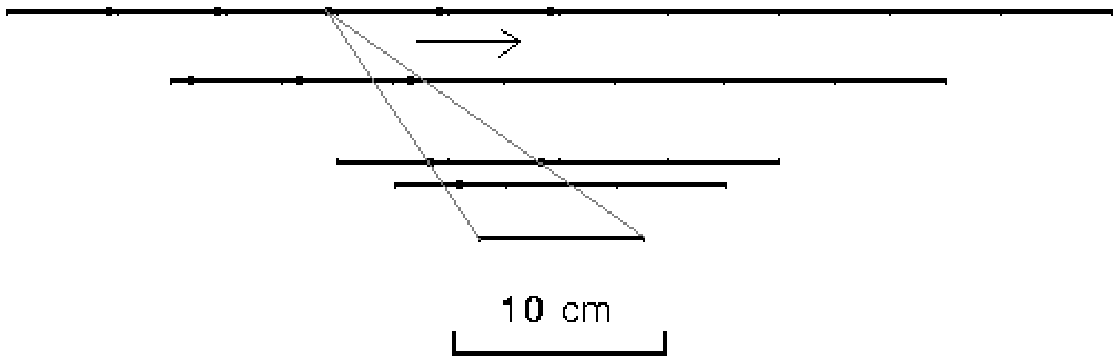

FIGURE 3. A z representation of the silicon layers and the beam region including some hit points and the search triangle for a layer 4 hit

tracks and validate them with a hit in layer 1. If no tracks are found for all of the hits in layer 4 then the algorithm continues with a similar search of layer 3.

Once the tracker has checked all of the ladders associated with the hit triggers it makes it decision on the events. The tracker accepts any events with at least 2 tracks and rejects the rest. It also returns the number of tracks that it found so that the information can be added to the event frame. Later the entire Level3 Algorithm combines the tracker's decision with the one from the crystals to determine if the event is worth keeping and to roughly classify the type of event.

Results

In order for the z tracking algorithm to be useful it needs to have an efficiency of around 90% of accepting "good" events and of about 80% of rejecting "garbage". The initial tracker had an efficiency of about 84% for accepting good events. While not as bad as it could be this is still insufficient. Some of the factors that could contribute the tracker inefficiency are silicon inefficiency, poor track resolution, and misaligned layer internal tracking structures. The latter two problems are algorithmic problems that can be solved with modifications to the algorithm code. The former is an inherent problem in the detector that might be partially corrected for using a clever trick but more than likely will be fatal to the success of the tracker. Although the tracker is not up to spec on its efficiency, it is very fast and has a very small memory footprint, meeting its other requirements.

Conclusions

The initial results look promising for the z tracking algorithm. While the algorithm does not currently meet the constraints placed upon it. It is not far from them and modifications will probably bring its efficiency up to operating tolerances. I regret not having the time to make the modifications myself because the tracker was essentially mine and I had hoped to see it working. With a working z tracker the Level3 should be able to filter out as many as half of events that are currently accepted because of an unreliable tracking algorithm.

Acknowledgments

I would like to thank Dr. David L. Kreinick of the Cornell University Laboratory of Nuclear Studies for his ideas and help as the algorithm took shape. I would also like to thank Alan Magerkurth of Cornell University for his help with all of my C++, silicon and other random questions. This work was supported by the National Science Foundation REU grant PHY-0097595 and research grant PHY-9809799.

Footnotes and References

1. David Cinabro, private communication via electronic mail.