Cosmic-Ray Muon Tracker in CLEO III

William Rice

Physics, Brown University, Providence, RI, 02912

(Dated: August 23, 2003)

Cosmic-ray muons trigger events that are devoid of experimental wealth. Using information from the muon chamber, it was hypothesized that an algorithm could be developed that would be able to reconstruct muon trajectories to determine whether or not the event was triggered falsely. This was shown to be true if a few assumptions were made about the muons and geometry of the muon chambers. The algorithm developed shows that muon trajectories could be reconstructed, which makes implementation of a similar muon tracking program into Level 3 a feasible goal.

I. INTRODUCTION

CLEO was developed to study electron-positron collison events. Muons are among the particles produced during this annihilating process and are studied in the muon chambers surrounding the interaction point. Because the muons are caused by an electron-positron collison, their energy is dependent on the center of mass energy of the electron/positron.

Muons can also be produced via high-energy (on the order of tens to hundreds of TeV) cosmic-rays hitting atmospheric particles. This natural collison produces pions, which quickly decay into muons. These cosmic-ray muons constantly bombard the earth at relativistic speeds at the rate of 1 $\text{cm}^{-2}\text{min}^{-1}$. Hence, in addition to muons created in the lab, cosmic-ray muons, whose energy is based upon high-energy cosmic-rays, are another significant source of muons that can be detected in the muon chambers of CLEO.

For the vast majority of cosmic-ray muons hitting CLEO, nothing is recorded or detected since the event trigger is not fired. Yet, when a cosmic-ray muon cuts through the interaction region, the trigger is activated and an event that isn't wanted is captured. The event, which is useless, is thus stored as viable data at a needless cost to the lab. In addition, it becomes time consuming to wade through copious amounts of events that aren't worth studying.

Originally, this situation was not much of a problem since most events during a given period of time were caused by an electron-positron collison. The COM energy for studying the resonances that CLEO had been interested in were around 10 GeV, which gave a beam luminosity of $8 \times 10^{32}\text{cm}^{-2}\text{sec}^{-1}$. Currently, CLEO is being reconfigured to study lower resonances (3-4 GeV). This gives lower beam luminosities of approximately $3 \times 10^{30}\text{cm}^{-2}\text{sec}^{-1}$. Hence, even though the number of cosmic-ray muon induced collisons remain essentially the same for a given COM energy, the number of electron-positron induced events drops two orders of magnitude. This translates into proportionally more cosmic-ray muon induced events per run.

II. GEOMETRY OF THE MUON CHAMBER

The muon chambers form the circumferential boundary of the CLEO detector assembly. Fig. 1 shows a number of major components of the detector assembly forming a core with the outer octagonal structure surrounding it. The muon chambers are positioned in the spaces between the three foot iron octagonal forms. The endcaps of the CLEO detector are covered by endcap chambers oriented vertically to complete the coverage. As a general rule, the endcaps are very noisy in comparison to the muon chambers in the octagon (they are actually divided up into eight 45 degree sections called octants). In order for a muon to penetrate through the endcaps and still hit the trigger, it would have to travel nearly parallel to the ground. Since this is a highly unlikely scenario and since the endcaps are very noisy, they were neglected in the algorithm.

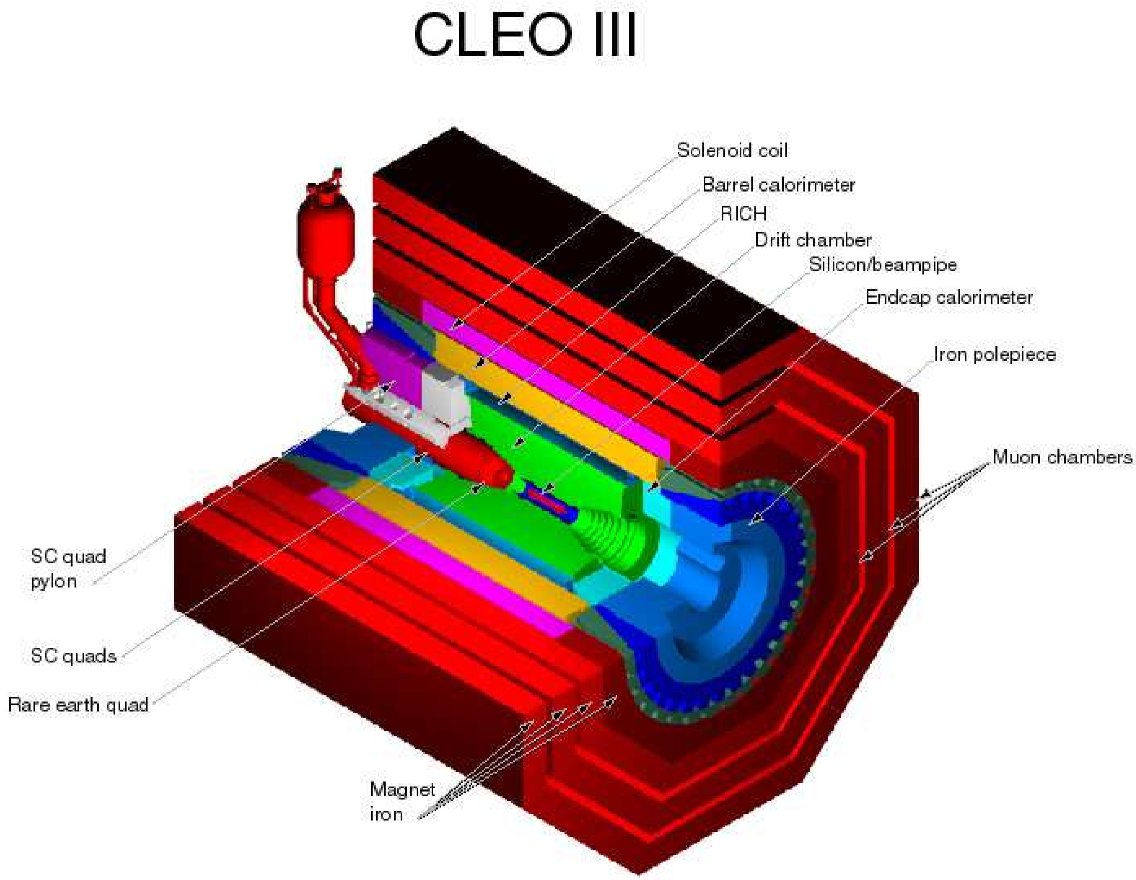

FIG. 1: This is a rendering of CLEO III. The octagonal circumference is composed of three foot iron sections and the muon chambers[1].

The iron that separates each layer in the barrel only allows muons to hit the muon detectors. Since interaction length of iron is 16.8 cm, no strongly interacting particles will penetrate the three feet of iron to hit the last muon chamber. Muons, however, do not interact strongly and can penetrate the full depth of the iron if they have enough energy. The layers of the barrel are sometimes referred to in terms of hadronic interaction lengths which are as follows: return layer is 3, inner layer is 5, and the outer layer is 7. Muons at low energies, like the energies that CLEO will be operating at to study charm quarks, have trouble making it to the outer layers of the barrel. Conversely, muons from cosmic-rays are at higher energies and pierce through the entire CLEO detector.

The muon chambers are divided up into eight octants and two endcaps. Each octant is divided up into three layers separated by one foot of iron. Each endcap only has one layer, which is further subdivided into three subunits. Similarly, each layer in the octants are also broken down into three subunits each.

A subunit is composed of six multiplets, which are formed via the groupings of counters and strips. Both the counters and strips lie in the plane that is perpendicular to the radial line running from the center of the detector through the middle of the octant. In this plane (which forms the subunit), the strips are orthogonal to the counters. The radial positon, the counters, and the strips form three basis vectors, which allows a hit to be described in any three dimensional coordinate system (though, for the majority of our purposes, we will use a Cartesian coordinate system).

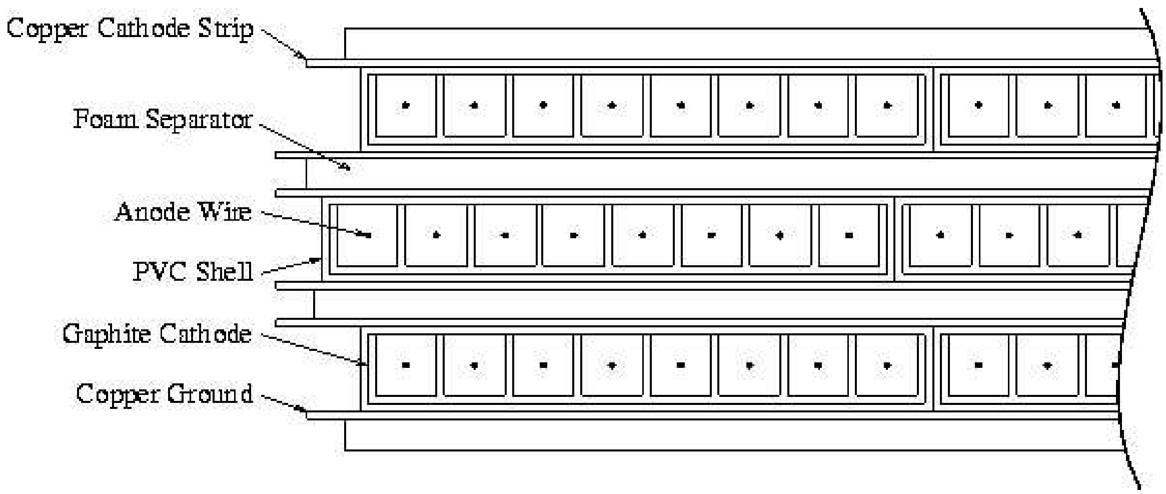

FIG. 2: The image shows the basic components of a multiplet. The cathode copper strips form the top of the plastic tubes. Inside the tubes, an anode wire running through the center. Eight of these tubes comprise a counter[2].

The counters and strips are really “seen” by the program as multiplets. Multiplets are the basic macroscopic unit for discriminating muons; they form the building blocks of the octant, layer, and subunit structures. A multiplet is really a division of counters and strips to form a unified way of looking at the muon hit in a coordinate frame. The composition of a multiplet varies in terms of how many counters and strip it contains[2]. For example, the first layer (a.k.a. “return”) is composed of two sets of ten counters each; in the outer-most layer (a.k.a. “outer”), one multiplet has 14 counters, while the other has 15. A subunit's counters are *always* grouped into two multiplets.

The other way that multiplets are formed is via the compilation of strips. There are always four multiplets constructed of a grouping of strips in a given subunit. These four strip multiplets plus the two counter multiplets combine to achieve our six multiplets per subunit. The number of strips in each of these multiplets, like with the counter multiplets, can vary. In one subunit of the return layer, 9 strips compose one multiplet, while 15 strips form the other three multiplets. The accompanying chart shows how the two counter and four strip multiplets are constructed. Notice that the number of strips and counters needed to form a multiplet is increased as the area of the layer is increased.

A hit is detected by a process that in many ways has its analogue in a photo multiplier tube. A muon, which is a charged particle, goes through a gas mixture of 60% He and 40% propane, which creates a trail of positively charged ions and electrons. The anode wires running through the center of a plastic tube (8 plastic tubes make up one counter) are held at 2500 V (see Fig. 2). Three sides of the tube are coated with graphite and are

TABLE I: A Breakdown of the Number of Counters and Strips in Each Multiplet by Layer[1]

Unit Location	Size(m^2)	Counters in Counter Multiplets	Strips in Strip Multiplets
Return	4.39x1.67	10 + 10	9 + 15 + 15 + 15
Inner	4.87x2.00	10 + 15	15 + 15 + 15 + 15
Outer	4.87x2.42	14 + 15	15 + 15 + 15 + 15
Endcaps	4.38x2.00	10 + 14	9 + 15 + 15 + 15

held at ground (with respect to the anode) to form the cathode. The free electrons created by the muons are attracted to the anode wire. As they travel, they gain enough energy to ionize other gas atoms, creating an avalanche of electrons. In effect, this process multiplies the signal from the electrons. The electrons' arrival at the anode induces a charge on the cathode strips, which is also sensed and digitized.

Each counter (the same can be said about strips) is connected via 100 ohm resistors, which we can utilize to find where the hit occurred in one direction (if we are only considering using the counters). By reading out the signal from the two ends of the string of counters as indicated in Fig. 3, we can tell which counter has been hit using charge division. The same thing can be applied to the copper strips, which constitute the fourth side of the plastic tubes (the other three, as mentioned above, being coated with graphite). The resolution of

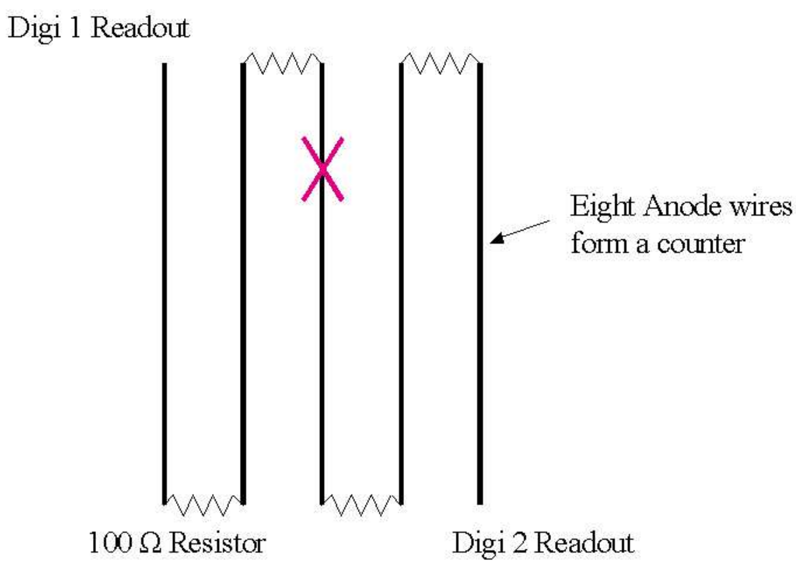

FIG. 3: Charge is placed on the wire from the ionized gas molecules. Digi1 and Digi2 are able to tell the position of the charge using charge division.

this method depends on the geometry of the counter. A counter is composed of eight plastic tubes, which measure 8 mm in height, 9 mm in width, and are 5 m long[1]. They are the objects that contain the ionizable gas mixture of He and propane. They are surrounded by 1 mm of plastic on every side, which means that all together, the eight tubes are 10 mm high, 83 mm wide and 5 m in length. Even though the eight anode wires comprise the counter, it is the width that is measured by the counter. How is this so? As discussed before, each counter in the multiplet is connected together via a 100 ohm resistor. The resistor provides the voltage drop between the read out at the beginning of the counter chain (let's call this output digi 1) and read out at the end (digi 2). It is only along the direction of the voltage drops that we can use charge division, since along the length of the wire, the voltage is constant. Moving from one group of eight anode wires (a counter) to another, though, incurs a different voltage as read from digi 1 to digi 2. The direction along the anode wires is measured via the strips that are in the copper that form the top side of the plastic tubes.

Thus, the resolution of the counter is really the width of the counter (.83 m) divided by $\sqrt{12}$ (this is a common result of statistics). This translates into a rms counter space resolution of .024 m and a rms strip width of .053 m. There is negligible error in the radial direction, since that is known absolutely: the position in the radial direction of the hit is simply which multiplet has been hit, which is set by the design.

III. ALGORITHM

The algorithm takes advantage of several key assumptions we can legitimately make given the constraints of our problem. One of those simplifying assumptions is that the muons travel in a linear fashion through the barrel. As opposed to the crystal calorimeter or the trigger, there are no magnetic fields present in the octants. Hence, although charged, the muons should be traveling in a linear trajectory. One problem to this is that once the muons enter the core of the detector ensemble, their paths can be influenced by magnetic fields. This means that if the muons have enough energy to proceed into another octant to exit the chamber, they do not necessarily have to be traveling linearly (although, they almost always still do).

Another assumption that is made is that since the muons have linear trajectories and since they need to pass through the trigger to initiate an event, a plot of θ versus ϕ (both spherical coordinates) should have a tight grouping of points. This idea will be explored later on in this paper. It is most instructive, though, to begin with the first parts of the algorithm to see how the operation of it maximizes the efficiency of geometry of the muon chambers as well as the simplifying assumptions made based on the physical intuition of the problem.

The algorithm begins by taking in all the hits that make up the event and sorting them according to what octant they penetrate. The discrimination of what octant was hit is done by a function that looks at the multiplet number associated with the hit. The pattern of the multiplets is such that this analysis is easy, efficient, and practical. Every octant that has more than four hits in it is recorded. The program then proceeds to run a huge loop of looking through all the "interesting" octants to look for those that are red herrings and those that actually have possible muon tracks in them. The initial multiplet analysis has reduced the problem to one that is repeatable for every octant, which takes advantage of the repetitive nature of the barrel.

The next cut is one that has yet to be perfected. It utilizes digi 1 and digi 2, which, as mentioned above, correspond to the signals read out when charge is deposited on the anode wire in a counter. In theory, digi 1 should be inversely related to digi 2, since a hit close to end 1 should have a large output at digi 1 and a small output at digi 2 due to the voltage drops provided by the resistors. As Fig. 4 illustrates, there are points all over the spectrum. A reasonable assumption is that those with both a small digi 1 and a small digi 2 are noise hits. That is, if we add digi 1 and digi 2, those with a much smaller than the average are most likely junk hits.

The program thus runs over the hits in a given octant, recording each sum so that the average can be obtained. If the average is under 1301, the cutoff for the digi sum is the average of the sums minus 100; if over 1300, the cutoff is average minus 250. The reason for the switch at 1300 is the recognition that high averages usually are the result of very strong signals, which pull up the average. The numbers to subtract from the average are somewhat arbitrary, but the underline a very necessary point: a cut can be made on the basis of the sum of the digi values from the counters. Whatever this cutting algorithm may be, the concept of making digi cuts is extremely useful.

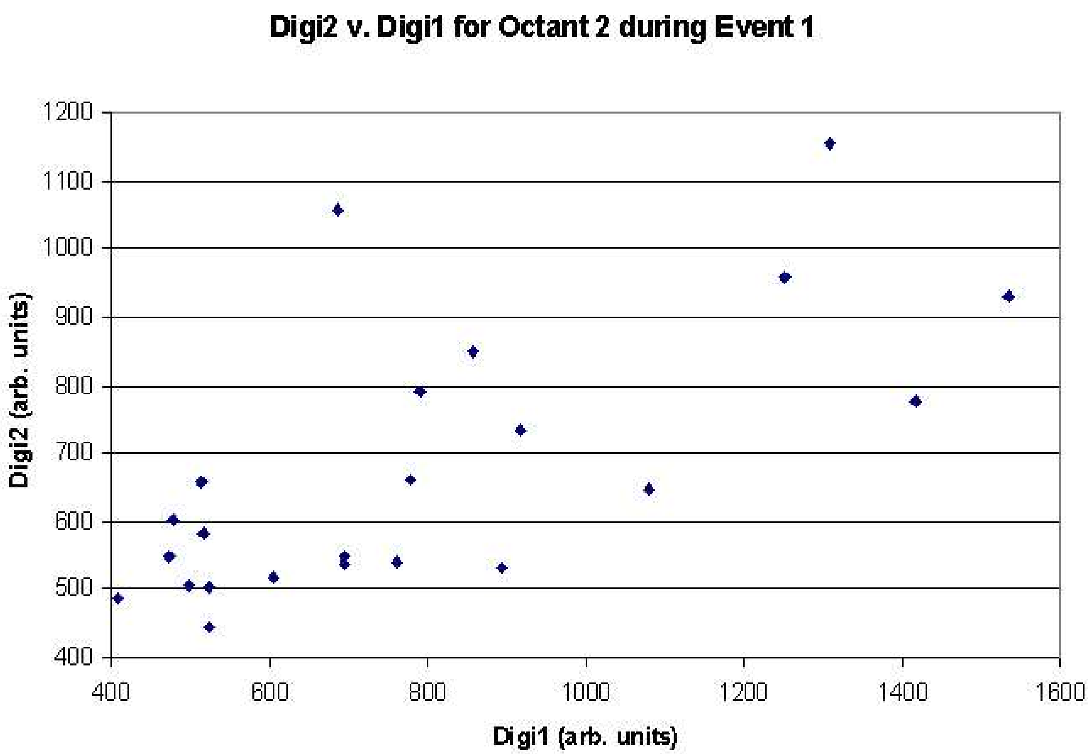

FIG. 4: Plot of Digi2 versus Digi1 shows that an inverse pattern is not observed. The lower values in Digi1 and Digi2 are considered false hits.

The question of how far below the average to make the cutoff is still under dispute. Taking all the points does not make sense, while taking only the top ten often gives poor results. So far, the points constituting a line have always had high digi sums. Indeed, they are often far higher than the cutoff established by the algorithm. One thought for the future is to use the median of the digi sums instead of the average, since the average can be sensitive to very high or very low sums, whereas the median is much less easily swayed. More testing needs to be done to set this cutoff more absolutely.

After the digi sum cuts, the program sees if it still has over four hits in the octant. In addition, it checks to see if all three layers have been hit. The function that checks to see if each layer has been hit uses the multiplet analysis scheme to apply the regular pattern of the multiplet numbers to the layers. Later, we will see how this function is utilized at the end of the program to give a necessary component to the weighing scheme the program uses to rate line fits.

The third and last major cut that the program makes on the hits is the most important part of the algorithm. This is the solid angle cut, and it relies on good physical assumptions about the problem. As mentioned above, the points should be moving in a relatively linear way. Thus, if we take the θ and ϕ angles (from their spherical coordinates) and use the interaction point as the origin, we should see a tight groups of points in a plot of θ versus ϕ. We can take the interaction point as a reasonable spot for an origin because of its relative size. Real events take place in a cylinder that is approximately 40 cm along the axis of the beam with a radius in the $x - y$ plane of 5 cm. Compare this to the radial distant of the muon chambers (closest layer in the $x - y$ plane is over 1 m away from the interaction point). The trigger must be an area where the muon goes through, since in order for the event to be recorded, the muon must fire the trigger.

One of the great advantages to this method is that in one of the classes being called up by the program has a function call for θ and ϕ. Thus, we have at our ready disposal, the constructs for the evaluation of our solid angle cuts. What the program does is to basically start at the beginning of the octant (octants span 45 degrees in ϕ) and then proceeds to look at ten degree swaths of the polar angle (polar angle describing a given octant goes from roughly 30 to 150 degrees) while holding ϕ constant. Then ϕ is increased by ten degrees and the process repeats. Thus, the program uses a ten degree by ten degree solid angle (in both θ and ϕ) as our tool to look for groupings of points, as illustrated in Fig. 5.

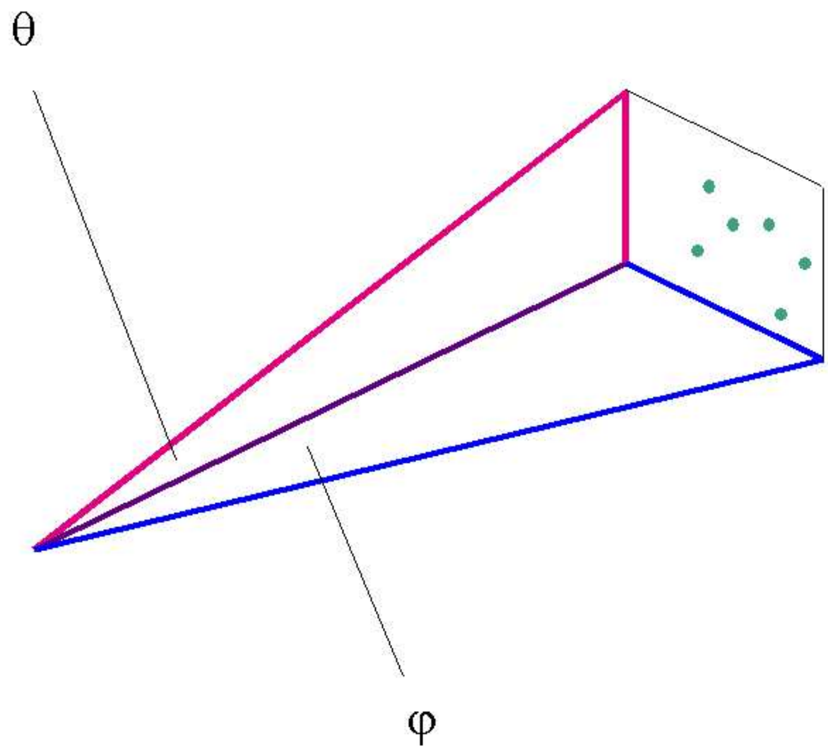

FIG. 5: A solid angle is defined by θ and ϕ.

When hits are found, the solid angle actually "slides" with the average of those points. That is, we originally defined our θ average and ϕ average to pre-determined values. When points are actually found, it is their θ values and ϕ values that are used to compute the θ and ϕ averages with which we define the middle of our solid angle. This is necessary because a pre-determined value of θ average and ϕ average does not take into account the fact that hits related to a group of hits already found by the solid angle search might be outside of this range. These hits, even though they belong with the other points, would thus be mistakenly characterized because of the searching pattern. The pre-determined ranges are just a way for the program to do an orderly search through the octant. When points are actually found, we are looking for a certain θ/ϕ region around those points. Hence, our average slides as more points are added on, while the actual bounds around these averages of the solid angle remain constant.

Within a given range of θ and ϕ (I tend to use 10 degrees for both), a grouping of points usually does not occur. On the chance that they do and if over three points are found, these points become the program's "guess" at where a muon's trajectory went. These "base" points are supplemented by a set of "possible" points: points that fall within a greater solid angle of the initial range. These points are not used during the initial fitting of the base points.

The reason for keeping a set of possible points is to account for the scenario that multiple muons cut through one given octant during a given event. If one muon triggered the event, while another one cut through the octant such that it missed the middle entirely, that second muon would not have a very tight solid angle that the trajectory would fall into; rather, it would cut across a θ/ϕ plot and would buck our initial assumption. Consequently, by taking a bigger solid angle, we can keep a bunch of points to fit to the base set of points if the situation arises that the line fit of the base points is poor.

In the case that the base points line fit is good, though, we simply store the parameters of the line and move on to look for more lines in the octant. A poor line fit, on the other hand, would take every combination of possible points and try to fit a line using the base points (our initial line guess) and those possible points. A few of the combinations for four possible points would be: just base points and no possible points (1 possibility of this = 4 C 0); base points and one possible point (4 ways this can happen = 4 C 1); base points and two possible points (6 ways = 4 C 2); and so on. For any given n possible points, there are 2^n combinations possible.

IV. LINE FITTING

The question of how to consider a line fit to a given set of x, y, z points is one that is not simply a mathematical exercise in least-squares minimization. Although the program does fit a line using a least-squares line fit, other physical parameters must also be satisfied for a trajectory to be considered "good." The way that the lines are weighted is that a certain numerical value is assigned to each criteria. In some cases such as the number of degrees of freedom, the more the better up until a certain point. Hence, the value of that weight reflects the fact that we want 7 dof (9 actual hits) and that anything under 3 dof and over 11 dof is unacceptable.

One of the weights used is how many layers and subunits the fitted trajectory is hitting. By using the multiplet numbers for each hit, we can easily find which layers and subunits are being included in the trajectory. A trajectory that includes more than two subunits must have at least some noise incorporated into it, which thus makes it a poor fit and it is weighted as such. A line that only hits two out of three layers is acceptable, but one that hits three is much more preferable.

By far, the most important weight is the chi-square density function statistic. Chi-square is a measure of how far a given fit deviates from its expected value. For the purposes that we are using it for, the chi-square number will measure how close our line fit comes to fitting a set of lines with a given minimum space resolution. The chi-square statistic with error in the dependent variables is [3] :

$$\mathcal{X}^2 = \sum_{i=1}^{N} \frac{(y_i - a - bx_i)^2}{\sigma_i^2} \tag{1}$$

where σ_i is the deviation in y for every point i. This formula can describe the chi-square statistic for lines in octant one and five, since all errors are either in the y or z direction (which are the dependent variables). The problem comes in when we move to other octants, since the error is now also in the independent variable, x. The reason for that is the Cartesian coordinate system that we imposing on the muon chambers makes it such that octants 2 and 6 are described by $y = f(-x)$, octants 4 and 8 are described by $y = f(x)$, and octants 3 and 7 are parallel to the x-axis. Thus, for octants 3 and 7, the error that was just in y direction for octants 1 and 5 is now in the x direction. For the other four octants, the error is split between the x and y axes.

The reason that this division is important is that for error in the independent variable, the chi-square formula becomes [3]:

$$\mathcal{X}^2 = \sum_{i=1}^{N} \frac{(y_i - a - bx_i)^2}{\sigma_{i_y}^2 + b^2\sigma_{i_x}^2} \tag{2}$$

To include z, which is not influenced by the different positioning of the octants in our coordinate system, we simply use Eqn. 1 and add that chi-square to whatever the chi-square is for the y versus x line.

The three dimensional line is broken up into two fits: the y versus x fit and the z versus x fit. This is done for a number of reasons. First, the function from the CERN library was designed for the two-dimensional line fit. It makes sense to use code that is widely available, known to all, and works reasonably well. Secondly, the breakdown into two different lines poses no challenge for us to reconstruct a three dimensional line. As long as we use the same independent variable for both lines, the three-dimensional case is trivially adapted. And thirdly, it is often more useful to know the respected y and z derivatives with respect to x than to have to breakdown a parametric description of the line.

The most useful application of the chi-square number is not in the immediate reading of its value; rather, it is how well it is positioned along a chi-distribution. The chi-distribution is created from the number of degrees of freedom (for our purposes, dof is equal to the number of points minus two), and it provides a way for us to gauge whether or not our chi-statistic is good. A more explicit way of saying this is that $P(\mathcal{X}^2|\nu)$ is the probability that the observed chi-square for our line should be less than $\mathcal{X}^2$ given ν is represents the dof. Following Press, we can define $Q(\mathcal{X}^2|\nu)$ to be $1 - P(\mathcal{X}^2|\nu)$, which is the probability that the observed chi-square will exceed $\mathcal{X}^2$ by chance even for a correct line. $Q(\mathcal{X}^2|\nu)$ is mathematically defined to be [3]:

$$Q(\mathcal{X}^2|\nu) = \frac{\Gamma(\frac{\nu}{2}, \frac{\mathcal{X}^2}{2})}{\Gamma(\frac{\nu}{2})} \tag{3}$$

where Γ is the full gamma function. The program utilizes the approach laid out in Press by taking the necessary algorithms to calculate $Q(\mathcal{X}^2|\nu)$. The line is defined as a mathematically good fit if $Q(\mathcal{X}^2|\nu)$ is above 0.1; the line may be accepted (if non-normal errors or the under-estimation of errors are present) if $Q(\mathcal{X}^2|\nu)$ is above 0.001 and less than 0.1. Anything lower than .001 indicates the fit is poor. Because of these various cutoffs, the program weights the goodness of line fit based on how high $Q(\mathcal{X}^2|\nu)$ is with values above or close to 0.1 being very heavily biased. Those lines with $Q(\mathcal{X}^2|\nu)$ below or close to .001 are very poorly weighted.

V. RESULTS

Although results are very difficult to obtain, since there is no "true" muon trajectory that we are working from, analysis by hand reveals certain patterns that most likely correspond to muon tracks. It is because of this that we can derive a muon's track to compare with the results of the program.

In each event the program encounted, the hits were divided into their respective octants correctly all of the time. The most significant events (those with over four hits in one octant) were always identified correctly by the program. Almost all of the time, the program could discriminate a possible line. A possible line is defined as an octant that has hits in all three layers (before the angle cuts but after the digi cuts) and has over four hits. Some of the error can be attributable to the digi sum cuts, which can be easily influenced by outlying sums. Another source of error may have been caused by the human analysis, since it is very difficult to tell if all of the layers have been hit in a given octant.

TABLE II: Comparison of Program's Results to Human Analysis over Thiry Events

Octants Found (%)	One Line(%)	All Lines(%)	One Line Correct(%)	All Lines Correct(%)
100	86.7	80.0	66.7	33.3

The digi sum cuts adversely affected the angle cuts, since without enough points falling within a given θ/ϕ region, a line fit was not attempted. Consequently, this tendency to not have enough points fulfilling the angle cut requirements led to missed tracks. Most events could have one muon trajectory identified correctly. The problem was that other trajectories were missed on a frequent basis. Some of this is due to the high $\mathcal{X}^2$ values that resulted from such low rms values. Even for what seem to be very good fitted lines (from r^2 and human eye), the low rms of the detector made it such that these trajectories were treated as bad fits. Hence, a mismatch of human and computer generated results due to an awkwardly weighted line (or a deceptively bad one).

The angle cuts, which were just implemented recently, need some modification. In almost all cases where there were over six points within five degrees of each other (in both θ and ϕ), a relatively high quality fit was achieved. Problems were seen when the dof got in the 2-4 range. One of the troubles with this function is that the sliding average method needs more work so that a couple points don't end up dominating the average. Also, there should be some method of justifying a line fit to a few points in the tight solid angle if there are a lot of points in the wider solid angle (the so-called possible points set). This may have the effect of slowing the algorithm down and/or fitting a poorer line than can be found, yet it has the benefit of catching more possible muon trajectories. Of course, fitting a line with a lot of points from the larger solid angle cut (a.k.a. "possible points") could also generate a false track that has a good fit but isn't real.

A more daunting problem is when a muon travels through the center of the detector while another muon in the same octant travels at a funny angle to the origin. To discriminate this event, the program checks a redundancy vector to avoid putting in points in the second line that went into the first line. A better way of finding that second line while making sure that points from the first line don't go into the second fit needs to be developed. Currently, the rough outline that I just elucidated accounts for most situations we would ever see. Still, possible trajectories that have not been encountered could pop up, which is why a more rigorous method needs to be devised.

The combinatorial line fitting has continued to perform as originally envisioned. This was tested under a variety of situations and was found to work well. The problem was not line fitting as much as the weighting mechanism, which is still be worked on. Given different weighting parameters, different lines could come out as the "best" line, which indicates that a rigorous testing of which criteria are most important is needed.

VI. CONCLUSION

The program developed clearly demonstrates several of the methods that are needed to eliminate points to form a correct and true muon trajectory. These point reduction methods are based in sound physical assumptions that allow a quick initial line fit to be guessed.

The problems that were incurred tend to be centered on where various cutoffs are set. The line weighting and the digi sum cutoffs were the two that caused the most amount of trouble. They still have not been correctly set. Although their values are necessary to establish a correct algorithm, the point of the algorithm was to demonstrate a method for determining a line describing a muon's trajectory via information from the muon chambers.

One of the failings of the algorithm was that multiple lines could not be found with as high a success ratio as just one line. This is attributed to function that is still in development. The concept of using a solid angle search through the octant, though, still stands as the best way to find one or more muon tracks in an octant. In that respect, the algorithm has achieved its goal of finding a solution for the multiple track problem.

The key conclusion that can be drawn from use of the program over a number of different events is that it is possible to reconstruct muon tracks using just the muon chambers. Implementation into Level 3 for processing in real time, although clearly not obtained by this program, is a real possibility using methods developed by this project. Furthermore, the program that is written can reasonably distinguish most tracks in most events it is run over. Further testing is needed to ascertain exactly what efficiency the program has and where and why the program goes wrong at certain events. The efficiency of the program can be rigorously tested by using the track finder from the central drift chamber to compare the results from the program with muon tracks known to be true and accurate. A methodology like this would quickly determine the efficiency of the program over a very large range of muon trajectories.

VII. ACKNOWLEDGMENTS

I want to thank Dr. David Kreinick of Cornell University for giving me the chance to work on this project. Many of his suggestions have saved me enormous expenditures of time and effort. Also, his patience with my computer skills (which are lacking) were very much appreciated.

I would also like to thank Prof. Richard Galik, who spent many hours trying to make this NSF/REU experience fun for us all. His efforts, patience, and endurance will not soon be forgotten by anyone.

This work was supported by the National Science Foundation REU grant PHY-0243687 and research co-opertaive agreement PHY-9809799.

---

[1] Y. Kubota, *et al.* Nucl. Instr. and Meth. **A320**, 67 (1992).

[2] D. Bortoletto, *et al.* Nucl. Instr. and Meth. **A320**, 114 (1992).

[3] W.H. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical Recipes in C.* Cambridge University Press: New York, 1988; Chapters 6 and 15.