High-Resolution Charged-Particle Tracking in BigBite Using Multi-Wire Drift Chambers

Andrew Puckett

October 12, 2004

Abstract

The physics possibilities of Jefferson Lab's Experimental Hall A will be greatly enhanced in the near future by the large-acceptance BigBite spectrometer. Its capabilities complement the two High Resolution Spectrometers(HRS) currently in use in Hall A. The initial detector instrumentation for BigBite, consisting of the so-called "trigger" and "auxiliary" planes, will provide fast triggering and crude track reconstruction for about 12% momentum resolution. The first approved experiments to use BigBite, such as E01-015: Studying the Internal Small-Distance Structure of Nuclei via the Triple-Coincident (e, e'p+N) Measurement will use this instrumentation. Other planned BigBite experiments, however, such as E02-013: Measurement of the Neutron Electric Form Factor G_E^n at high Q^2 will require momentum resolution better than 1%, so they will need the precise tracking provided by the multi-wire horizontal drift chambers(HDCs) currently being developed at the University of Virginia. A prototype chamber is already being tested on the bench at UVA with cosmic rays. Software to analyze the single test chamber is already being used on cosmic ray data and a track reconstruction algorithm for the full three-chamber tracking system has been implemented and tested with simulated tracks.

1 Introduction

The BigBite spectrometer is a non-focusing magnetic spectrometer with a large angular and momentum acceptance. It will be an extremely valuable addition to Hall A because its capabilities complement those of the High Resolution Spectrometers, which have very small acceptances but very good resolution. The HRSs can detect particles very precisely in a small kinematic range. BigBite, on the other hand, can cover a broad range of kinematics in a single setting, which is important when studying interactions for which cross sections are small.

BigBite consists of a dipole magnet with a central bend angle of 25°, a maximum magnetic field of 1.2 Tesla, and a number of detectors to count, identify and track the particles accepted by the magnet. The initial BigBite detector package in Hall A will consist of two parts, the trigger plane and the auxiliary plane. The trigger plane(Fig. 1) consists of two planes, δE (3 mm thick), and E (30 mm thick) of segmented plastic scintillator bars attached to photomultipliers on either end, for fast triggering and particle identification. Each trigger plane has 24 scintillator bars. The auxiliary plane(Fig. 2) is another plane of 56 segmented scintillator bars with photomultipliers designed to provide crude track reconstruction. This setup is shown in Fig. 3.

BigBite has a solid angle acceptance of 96 msr when placed 1 m from the target, a minimum momentum of 250 MeV, and no real upper limit on momentum, as can be seen from its geometry. It has a horizontal acceptance of $\pm$ 80 mrad(4.6°) and a vertical acceptance of $\pm$ 300 mrad(17.2°). Initial test results on the trigger planes indicate a timing resolution of better than .25 ns, and a 12% momentum resolution can be achieved using the hit pattern of the auxiliary and trigger planes. Notice how the scintillator bars in the E-trigger plane overlap those of the δE plane, so that the trigger plane is effectively divided into 48 sectors. Combined with the 56 sectors of the auxiliary plane, this arrangement allows for a crude track reconstruction.

Replacing the auxiliary plane in BigBite with horizontal drift chambers will enable much more precise track reconstruction. The HDC tracking system consists of three separate drift chambers. Each chamber consists of a number of planes of parallel wires sandwiched between high voltage cathode foils and enclosed in a gastight frame filled with a mixture of 65% Argon and 35% ethane at just above atmospheric pressure. The active area of the first chamber is 140 cm in the dispersive(x) direction $\times$ 35 cm in the out-of-plane(y) direction. By convention, the direction of increasing x is taken to be the direction of increasing momentum, in other words, towards the ground (since the paths of particles with higher momentum are bent less by the BigBite magnet), the z axis is taken to point perpendicular to the wire planes in the direction of the particle tracks, and the y axis is taken so that the $\hat{x}, \hat{y}, \hat{z}$ axes form a right handed orthogonal coordinate system. The area of both the second and third chambers is 200 cm(x) $\times$ 50 cm(y). Each chamber has wires in three different directions to precisely measure track coordinates and angles in both x and y directions. The U(V) wires are at an angle of -(+)60° relative to the x-axis, while the X wires are perpendicular to the x-axis. The first chamber is positioned approximately 10 cm downstream of the exit of the BigBite magnet. The second and third chambers are located 35 cm and 70 cm downstream of the first chamber, respectively. The configuration of planes and wires for the first and third chambers is shown in Figure 4. The middle chamber differs only in that there is just one plane each of U, V, and X wires instead of two. The combined three-chamber tracking system thus has five planes each of U, V, and X wires, making possible a robust and very precise track reconstruction algorithm that can handle very high rates.

The wires are alternating signal(anode) and field(cathode) wires separated by .5 cm. The field wires are 90-μm-diameter copper-beryllium kept at a potential of -1.7 kV, the signal(or sense) wires are 20-μm-diameter gold-plated tungsten kept at ground potential, and the cathode planes are 12-μm-thick double-sided copper-plated mylar kept at -1.1 kV. The copper layer on each side of the cathode foils is 1200 $\overset{\circ}{A}$ thick. The chosen voltage configuration gives a strong horizontal $\vec{E}$ field for drifting of electrons and ions. The measured drift times are used to determine the point at which a track crosses a given plane with a precision of ~200 μm. The wires in the second plane of each pair are shifted by .5 cm relative to the first plane so that the signal wires of the second plane overlay the field wires of the first plane. Arranging the wires as such helps resolve the left-right ambiguity of the drift time measurement. This ambiguity arises because, while the distance from the track to the signal wire is known from the measured time required for the drift electrons to reach the wire, the direction of the drift electrons is unknown. The track could have crossed the plane on either side of the wire from the same distance away. The implications of this ambiguity for track reconstruction will be discussed below.

The GARFIELD drift chamber simulation package developed by Rob Veenhof at CERN performs a number of calculations that are useful in the design, simulation and operation of drift chambers. It can calculate and plot the electric field for a given configuration of wires, planes and voltages, calculate the electron transport properties of the gas mixture, simulate the ionization of the gas by a high energy charged particle, calculate the drift time for electrons originating from a given point in the chamber, simulate the avalanche process near the signal wire, and much more. We used GARFIELD often in the BigBite drift chamber R&D effort, particularly to determine the optimal voltage configuration and to facilitate a precise time-to-distance conversion.

2 Field and Drift Properties

As a high-energy charged particle goes through the drift chamber, it leaves a trail of ionization clusters in its path. The liberated electrons drift along the electric field lines toward the signal wires, while the ions drift toward the field wires. In most of the region between a field wire and a signal wire, the electrons settle into a roughly constant drift velocity. The drift velocity depends on the electric field and the properties of the gas mixture. When the electrons reach the rapidly increasing field near the signal wires, their drift velocity becomes so large that they gain enough energy between collisions to ionize additional atoms, initiating an electron avalanche that amplifies the signal on the wire to the point where it is large enough to distinguish from other noise on the line. The signal on the wire is actually a space charge-induced current pulse. Since the signal wire is held at a constant potential by a high-voltage power supply, it must draw current to compensate for the voltage difference generated by the depletion of ions from its vicinity. This current pulse is then converted to a voltage pulse by a preamplifier and sent to suitable readout electronics. The gas mixture for a drift chamber is typically chosen to have a primary gas–usually a noble gas–responsible for the ionization and a secondary polyatomic gas to absorb excess electrons, thereby preventing a continuous discharge from occuring in the strong electric field near the signal wires. The transport properties of the Argon-Ethane mixture in this chamber are very well-suited to the detection of charged-particle tracks in BigBite.

GARFIELD has an interface to the Magboltz program that can calculate the electron transport properties of virtually arbitrary gas mixtures. Of particular interest is the dependence of the electron drift velocity on the electric field, since measuring the time for electrons to drift from the track to the signal wire is what will enable reconstruction of the position at which the track crossed the wire plane. As Figure 5 clearly shows, there is a broad valley in the drift velocity curve for electric fields from about 2-10 $\frac{kV}{cm}$ in which the drift velocity varies only weakly with the electric field. This property is especially convenient since this is exactly the range of fields in which the chambers will be operating. The average magnitude of the electric field in the plane between a signal wire and a field wire is simply $<|E|>=\frac{|\Delta V|}{d}$ where ΔV is the potential difference between the two wires and d is the distance separating them. Plugging in 1.7 kV for ΔV and .5 cm for d, we get $<|E|>= 3.4\ \frac{kV}{cm}$, which falls right in the middle of the near constant drift velocity region of the graph. Figures 6 & 7 show the electric field layout of the BigBite drift chambers. The graph in Figure 7 is a plot of the electric field on a path from a field wire located at $x = 9.5\ cm$ to a signal wire located at $x = 10\ cm$. The curve is only plotted from $x = 9.55\ cm$ to $x = 9.95\ cm$ in order to show the small-scale variation of the field between the wires, which would not be apparent if the steep rise in the electric field very close to the wires were shown. The lowest values of the field are at the lower edge of the valley in the drift velocity graph, and it is evident that the drift velocity remains nearly constant throughout the region.

Since the electron drift velocity is nearly constant, as a first approximation we can calculate the drift distance from the drift time via $d = vt$ where v is the average drift velocity. In reality, the relation between drift distance and drift time is not quite linear, and depends not only on the drift time, but also on the angle of the track as it crosses the plane. For the nominal trajectory perpendicular to the planes, the electrons originating in the wire plane and drifting horizontally to the wire have the shortest drift times, but for tracks at an angle it is possible for electrons originating outside the horizontal plane and drifting to the wires at an angle to have a shorter drift time, which changes the relation between the minimum drift time and the horizontal distance from the track to the wire. Figure 8 shows a GARFIELD simulation of a 2 GeV electron going through the chamber at an angle. The ionization of the gas and the drift of electrons are simulated by GARFIELD's interface to the HEED program.

Figure 9 is a plot of minimum drift time versus horizontal distance from a signal wire calculated by GARFIELD for a track angle of zero degrees, in 50-μm intervals. Close to the signal wire, the field increases rapidly and the drift velocity is no longer constant. Figures 10 and 11 show plots of the x(t) relation calculated by GARFIELD at track angles of 0°(the nominal trajectory) and 20°(close to the maximum angle allowed by the acceptance of the spectrometer) respectively. The linear least-squares fit to the plots shows that the constant drift velocity approximation is very good at 0° and at least reasonable as a first approximation at 20°. Electrons originating from tracks crossing very close to the wires have much shorter drift-times than would electrons moving at the average drift velocity. If the tracks are distributed evenly over the chamber area, this results in a distribution of drift times with a high spike at very short drift times followed by a broad, flat distribution spread out evenly over the rest of the range of possible drift times, corresponding to the region of near-constant drift velocity. If a more advanced time-to-distance conversion than the linear approximation is used, a good way to check the effectiveness of that conversion is to see whether this distribution looks flat after the conversion is applied. Since the distance depends not only on time but on track angle, the final track reconstruction algorithm, as discussed below, will require a two-step process for determining x for a given t, first using the constant-v approximation to get a crude fit and an angle for the track, using this track to look up the correct values in a table generated by GARFIELD(using a bilinear interpolation between the two closest values of t and θ) and then recalculating the track using the new values.

3 Track Reconstruction

When a charged particle hits the BigBite trigger plane, it generates an electronic trigger indicating the time at which it traversed the detector stack with ≈ 0.25 ns precision. The rise and fall times of hits on a wire are very short, making it possible to resolve hits on a single wire as close together as ≈ 10 ns. The time-to-digital converters(TDC's) which measure the drift times can resolve drift times with ≤ 1 ns precision. Taking the slope of the linear fit to the x(t) relation in Figure 10 as the average drift velocity, about 4.855 $\frac{cm}{\mu s}$, gives drift times ranging from 0-105 ns for drift distances from 0-.5 cm[1]. With a drift time measurement window of 125 ns, any track passing through the chamber within $\pm$ 125 ns of the trigger[2] can cause hits on wires, but as the time difference between a track and the trigger increases, the probability of any of the hits falling within the 125 ns window decreases. In addition, since the time offset of those tracks relative to the event trigger is unknown, the drift times do not accurately reflect the drift distances. In order to figure out the drift distance, we would first have to figure out the time offset of the track. The coordinate at which a track crosses a wire plane is calculated by adding or subtracting the measured drift distance from the wire's known coordinate.

In order to design and test a track reconstruction algorithm, it was necessary to simulate the real-time operation of the drift chambers. In the high-luminosity environment in Hall A under experimental conditions, the rate of good electron tracks in BigBite could be 10 MHz or even higher. At 10 MHz, with a 125 ns drift time measurement window(actually a 230 ns window within which tracks can cause hits) the average number of tracks passing through the chamber during a 125 ns window is 2.30. Since the counting rate is approximately constant, the number of tracks that can cause hits during a given event follows a Poisson distribution $p(n) = \frac{a^n e^{-a}}{n!}$. Here p is the probability of n tracks occuring, and $a = 2.30$ is the average number of tracks. Clearly there is a significant probability for even as many as six tracks or more to cause hits in the chamber at the same time at such high rates.

In addition to the high rate of hits from good tracks, there is also a large number of hits coming from random background events, predominantly low-energy photoelectrons, which typically have a range of a few cm in the chamber gas mixture and cause individual hits or small fragments of tracks having hits on several planes but not in all planes. One reason for using three drift chambers separated by a relatively large distance, besides improving the resolution of the track angle, is to make it easier to reject such background tracks in a high-rate situation, because they will typically appear in just one of the chambers rather than all three. With such high counting rates in the chambers, a robust track reconstruction algorithm will be needed that can fit multiple good tracks occurring at the same time and identify and reject unwanted background hits.

The chamber being tested on the bench at UVA with cosmic rays is identical to first of the three chambers in the full tracking system. It is well-suited to detecting and tracking particles(with limited resolution of track angles due to its narrow width) in a low-rate environment. A separate analysis algorithm has already

[1]If the drift distance was more than .5 cm, the track would be closer to the adjacent signal wire and it would fire instead, with a drift distance of less than .5 cm.

[2]Technically, tracks going through the chambers within $-105\ ns \leq \Delta t \leq +125\ ns$ can cause hits, because, although a track occuring within +125 ns of the trigger could cause hits if the drift times for those hits were very small, a track occuring more than 105 ns before the trigger could not cause hits because the drift time for any of the hits caused by said track could not be greater than 105 ns.

been tested on the cosmic ray data and produces sensible results, although insufficient statistics are as of yet available to test the efficiency of that algorithm.

Each wire plane measures the coordinate of the track along the axis perpendicular to the wires. Whereas in the Hall A VDCs the tracks cross each wire plane at a nominal 45° angle and the electrons drift vertically to anode wires, in the BigBite HDCs, the tracks cross each plane at a nominal 90° angle and the electrons drift horizontally. In the VDCs a typical track causes 5 hits in each plane, allowing 5 drift distances to be measured, while in the BigBite HDC's only one drift distance is measured in each plane for a given track, and the track's projection onto the axis measured by a set of planes is found by fitting a line to the drift distances from each plane corresponding to that direction. While having planes of wires in at least two different directions allows precise track reconstruction, having three wire directions allows multiple tracks to be reconstructed at the same time, because the third direction can be used to properly match the various fragments of tracks from the other two directions. As an example, suppose we have two tracks in the chamber at roughly the same time, and wires in two different directions, which I will call U and V. From these two sets of planes, we separately reconstruct U, U' (U' $\equiv \frac{dU}{dz}$) and V, V' (V' $\equiv \frac{dV}{dz}$) for each track, but how do we determine which U's go with which V's? The answer is, by using the U and V coordinates and angles to compute the track coordinate and angle along the axis measured by the third set of planes, which I will call X. The proper combination of U and V track-pieces is that which agrees with the measured X track-pieces in both coordinates and angles. It is thus immediately apparent that having three directions of wires instead of two enables track reconstruction in a much higher-rate environment than would be possible with just two.

In designing a track reconstruction algorithm for the BigBite HDC's, my top priority was making the algorithm as general as possible so that it would be easily adaptable to any anticipated layout of planes, not only in BigBite, but in any spectrometer using horizontal drift chambers for particle tracking, including the Medium Acceptance Device(MAD) spectrometer planned for the post-12-GeV physics program in Hall A. There is no specific information about the properties of the chamber hard-coded into the software. Instead, all the necessary information is read in from an appropriate database. For the moment, the database reading software is geared specifically towards the prototype HDC at UVA, but the analysis algorithm is capable, with no modification whatsoever, of handling the three-chamber tracking system for the G_E^n experiment, designs for an HDC for MAD, or other potential HDC layouts for BigBite.

At present, however, there are still several limitations on what kind of HDC layout the analysis software can handle. It assumes that the chamber has exactly three types of planes(directions of wires), no more and no less. It also requires that there be either exactly two planes of each type or at least three planes of each type. There is currently no option to look at two planes of one type and more than two planes of other types, nor is there any option to look at just one plane of a certain type, although it would not be too difficult to add these capabilities to the software. In the case of only two planes of each type, the software also implicitly assumes that planes of a given type are arranged in the "checkerboard" configuration as in figure 4, in which the wires of adjacent planes of a given type are shifted by $\frac{1}{2}$ wire spacing relative to each other so that the field wires overlay the signal wires of the next. Why require 3+ planes and/or a checkerboard wire layout? Because of the left-right uncertainty in the drift time measurement. Remember that the drift time only gives distance information, not direction. For that reason, the possibility that a track crossed at either plus or minus the drift distance from the wire must be considered. One way of resolving this ambiguity is by fitting lines to all possible left-right combinations of a set of drift distances by the method of least-squares, and finding the combination that minimizes the χ^2 of the line of best fit. Since it is always possible to fit a perfect line to just two points, at least three planes are needed to determine the correct left-right combination in this manner. Even when at least three planes of each type are used, the checkerboard wire configuration is recommended, but not required, since it helps minimize instances in which the wrong left-right combination could accidentally give a better χ^2 than the correct combination. How it helps in the two-plane case is discussed below.

Tracking with two planes in each direction In the two-plane case, the following is done for each event. I have denoted the three plane types as U, V, and X:

1. Find all combinations of two hits(one from each plane) that are close enough to each other to have been possibly caused by the same track[3]. Repeat for each plane type.

2. Go through all possible combinations of three such groups of two hits(one each of U, V, and X).

3. For each two-hit group within each such combination of three groups, fit all four possible lines corresponding to the four possible left-right combinations. If the angle of the line is inconsistent with the angular acceptance of the chamber, toss it out. Now each two-hit group will have four or fewer possible lines of best fit.

4. Find the combination of one U-line, one V-line, and one X-line for which the X coordinate and X angle calculated from the U and V hits best matches that calculated from the X hits.

5. If the combination of lines that best agrees does so to within a cut determined by an estimate of the resolution of the coordinate measurement, reconstruct a track from these lines, and flag the combinations of hits used to fit that track so that no two-hit group will be used to reconstruct more than one track.

This algorithm offers an alternative way to solve the left-right problem when χ^2 cannot be used to select the correct combination, but it is of course much less accurate. In a high rate situation in particular, it is easy to imagine a certain combination of hits and left-right that would accidentally give a false agreement, superseding the real track. However, for cosmic ray testing it should be fine since the rates are very low. Note that the above algorithm requires all six planes to fire. If one of the hits from a track goes missing, due to firing inefficiency in one of the planes, it is, in principle, still possible to reconstruct a track from five hits, by using the lines from the two directions for which both planes fire and the individual hit from the third direction and finding the left-right combination that best matches the coordinate of the individual hit with that calculated from the two lines. Since no angle can be calculated for the direction with only one hit, and therefore only coordinates can be compared and not angles, this method is even less certain. At present the algorithm to reconstruct a track from just five hits is incomplete, but the algorithm to reconstruct tracks when all six hits are present is complete and functional.

[3]In Hall A under experimental conditions, good tracks will have a limited angular range corresponding to the angular acceptance of the BigBite spectrometer. If, as is the case in the UVA prototype HDC, the planes of each type are in pairs very close together, so close, in fact, that the maximum change in position of the track before the next plane is less than $\frac{1}{2}$ wire spacing, and if the wires are in the checkerboard layout, there are only two wires on the second plane that can be hit by the same track, the wires immediately to the left and to the right of the hit wire. If, on the other hand, the signal wires in the two planes lined up with each other, the second hit could be on any of the three surrounding wires. This is one advantage of the checkerboard setup.

The two-plane tracking algorithm, despite its limitations, is a useful tool in determining the resolution of the chambers, since the cut for satisfactory agreement of the lines constructed from U and V hits with the line constructed from the X hits is based on the intrinsic resolution, that is, the precision with which the coordinate of a track at a given plane can be measured. This resolution is determined by the limited precision with which the wires can be placed($\approx 50\ \mu m$), the TDC resolution (≈ 1 ns ($= 50\ \mu$m for typical drift velocities)), and, at least in the initial stage of the analysis during which a constant drift velocity is assumed, another ($\approx 50\ \mu m$) due to the error in the assumption of constant v. 200 μm is being used as a conservative initial estimate of σ, and the software succesfully fits tracks to cosmic ray data using this σ value. Though, as mentioned above, we lack sufficient statistics to draw any conclusions about the efficiency of the analysis algorithm, the initial success at least suggests that we are in the right ballpark on the resolution estimate.

Tracking with more than two planes in each direction When at least three planes in each direction are available, we can use χ^2 minimization[4] to determine the correct left-right combination of a group of hits. First the program looks for combinations of three hits(one hit each from three different planes) in a given direction and tries to fit these combinations. It does so by fitting all $2^3 = 8$ left-right combinations of those three hits, tossing out lines with angles outside the angular acceptance, and finding the line among the remaining combinations with the least χ^2. The cut that a line consisting of N hits must satisfy to be considered as possibly a good track is

$$\chi^2 \leq \sigma^2 \frac{10N}{3} \tag{1}$$

where σ is the estimated resolution[5]. If the best left-right combination of the three hits in question satisfies (1), then for each hit in the group a logical variable indicating that it has been used to fit a track is turned on, so that the same hit will not be used to fit more than one track. The group of three hits, together with its line of best fit, is then added to an array consisting of all such combinations (with lines of best fit) for the current group of three planes under consideration. The program searches for such three-hit groups in each possible combination of three planes in a given direction. After finding all three-hit groups in a combination of three planes, the program searches for hits in the remaining planes to add to each group before searching the next combination of three planes for three-hit groups. A hit from one of the remaining planes is added to a group if the new line of best fit calculated for the group with the new hit still satisfies the χ^2 restriction above. If there are exactly three planes of a given type, then any three-hit group will be considered the projection of a good track onto the direction measured by those planes. If there are N planes of a given type(where, of course, $N > 3$), the program requires a group to have at least $N - 1$ hits to be considered a good track.

Time offset and multi-tracking Given the very high rates possible under experimental conditions in the BigBite spectrometer, there is a high probability of having multiple good tracks traverse the chamber even during as brief a window as 125 ns, each causing enough hits to look like a good track to the analysis algorithm. For this reason, the track reconstruction algorithm has to be able to reconstruct multiple tracks per event. The most important function of the algorithm is to reconstruct the track which caused the event trigger as precisely and efficiently as possible. After this, any additional successfully reconstructed tracks

[4] $\chi^2 \equiv \sum_{i=1}^{N} [y_i - (mx_i + b)]$ is the sum of the squares of the deviations of the data from the line of best fit.

[5] The factor $\frac{10N}{3}$ just means that we allow each hit to differ by, on average, $\sqrt{\frac{10}{3}}\sigma$ from the line of best fit. While this specific number seems somewhat arbitrary, it was the number that, in real-time simulations of track reconstruction, seemed to strike the best balance between rejecting false tracks and accepting good tracks.

are considered a bonus but are not usually essential to the basic functionality of the tracking system. Time offsets of a few ns or less relative to the event trigger do not substantially alter the measured drift distance relative to the actual drift distance, so such tracks can only be distinguished by matching the coordinates and angles measured by the first two sets of planes with those measured by the third set of planes. Using this capability alone enables reconstruction of multiple tracks *very* close to the event trigger.

For tracks with time offsets greater than a few ns, the measured drift time no longer accurately reflects the drift distance. Since the only information contained in an HDC hit is the drift time, there is no way to know what the time offset is for the track that caused a given hit. If we are only interested in reconstructing trigger tracks, then the fact that an unknown offset is added to the drift time is good because it will be less likely that those hits (if they are not lost altogether) can be fit to a line with a good χ^2. It is also possible to actually reconstruct these tracks if there are at least three planes of each type so that χ^2 minimization is the method used to fit tracks. This is accomplished by including a parameter t_0 in the minimization. The program iterates t_0 in small ($\approx 1ns$) increments (consistent with the TDC resolution), subtracting vt_0 from vt_{drift} and finding the best left-right combination for each t_0, checking each line to make sure that its slope falls within the acceptance and that none of the drift times for any of the hits in question would lie outside the 125 ns measurement window when added to t_0. The range of iteration of t_0 is typically about $\pm$ half the drift-time measurement window, because tracks with time offsets much greater than this are unlikely to cause enough hits to be successfully reconstructed. After finding the best sensible left-right combination for each t_0, the program chooses the t_0 with minimum χ^2 as the time offset of the track.

Even when the program attempts to reconstruct tracks with large time offsets relative to the event trigger, it first attempts to reconstruct as many $t_0 = 0$ tracks as possible. After doing so and (presumably) having reconstructed the trigger track and any tracks close enough to it not to have their apparent drift distances significantly altered, the program can then search the remaining hits for any good tracks with larger time offsets. Determining the time offsets of tracks relative to the event trigger establishes another criterion for matching the different track-pieces from the three directions–the t_0's of any three track-pieces must match before the group can be used to reconstruct a track.

Once the program has found as many track pieces as it can in each direction using the hits for a given event, it attempts to reconstruct hits from those tracks. Given a group of three track-pieces, one from each direction, it transforms the measured (U, V) and (U', V') to (X, Y) and (X', Y') via the following transformation:

$$\begin{pmatrix} x(x') \\ y(y') \end{pmatrix} = \begin{pmatrix} \sin\theta_u & \cos\theta_u \\ \sin\theta_v & -\cos\theta_v \end{pmatrix}^{-1} \begin{pmatrix} u(u') \\ v(v') \end{pmatrix} \tag{2}$$

Be careful to note the inverse sign on the matrix in (2), since it is clear from figure 12 that it is the matrix that transforms from $(x, y) \rightarrow (u, v)$, not the other way around, so it must be inverted for (2) to hold. The coordinates and angles measured from the U and V planes thus transformed are then compared to those measured by the X planes. The criteria for agreement between these track-pieces are as follows:

$$\Delta X \leq 10\sigma \tag{3}$$

$$\Delta(\frac{dx}{dz}) \leq \frac{\Delta X}{\Delta Z} \tag{4}$$

$$\Delta t \leq \frac{\sigma}{v} \tag{5}$$

In (3)-(5), ΔZ is the width of the chamber and σ is the estimated resolution. If inequalities (3)-(5) are satisfied by the track-pieces in question, then a track, which is defined by five coordinates $(x_0, y_0, \theta, \phi, t)$, is reconstructed as follows:

$$\begin{pmatrix} x_0 \\ y_0 \\ \theta \\ \phi \\ t \end{pmatrix} = \begin{pmatrix} \frac{2x_{uv}+x_x}{3} \\ y_{uv} \\ \frac{2\theta_{uv}+\theta_x}{3} \\ \phi_{uv} \\ \frac{t_u+t_v+t_x}{3} \end{pmatrix} \tag{6}$$

The subscripts u, v, and x of course denote the planes used to measure a given coordinate or angle. This is how to reconstruct a track if at least one piece in all three directions is present. However, sometimes after all such tracks have been reconstructed there may still be some apparently good track-pieces left over, or it may be the case that for whatever reason only two track-pieces are available to reconstruct even the trigger track. When this happens, it is still possible to reconstruct a track from any two sets of planes, simply by constructing the matrices to transform from the coordinates measured by those planes to the usual (x, y). If only two directions are available, however, it seems impossible to check that the various track-pieces match up correctly, other than by t_0 matching, which is useless if two tracks occur very close to each other in time. However, even though there may not be a track-piece reconstructed from the third direction, there may still be one or two hits from the track on the third set of planes. So we can check whether a track reconstructed from just two sets of planes is a good track by demanding that we find at least a few hits in the third set of planes sufficiently close to the extrapolation of the track to those planes. If such hits can be found, then the track reconstructed from just two sets of planes is accepted as a good track. Every time any track passes all the criteria for a good track, each of the track-pieces used to reconstruct it is flagged so that it won't be used to reconstruct a track more than once.

4 Current Status of Project

The analysis algorithm as described has yet to be completely developed within the Hall A analyzer framework. When I left it at the end of August 2004, the software to analyze data from the UVA HDC seemed to be working properly, though it was at that time only capable of reconstructing tracks if all six planes fired for a given event. I had yet to finish the part of the software to reconstruct tracks based on hits in five planes. For the chamber with only two planes of each type, it seems impossible to resolve the left-right ambiguity for a track if any fewer than five hits are present.

In the case of more than two planes of each type, in which χ^2 minimization is used, the part of the algorithm that reconstructs trigger tracks (and all other tracks within a few ns of the trigger time) works very well (see simulation results below) but I have yet to get the part of the algorithm working that determines the t_0 of and reconstructs the tracks with larger time offset, which is unfortunate because I was able to get it working in a previous standalone simulation that I wrote. However, the algorithm I used in that simulation was different from that described above and designed only for one specific configuration of the HDC's (that planned for G_E^n).

As mentioned above, it is also possible to perform a more accurate time-to-distance conversion using a table of values calculated by GARFIELD. Since the GARFIELD t-to-x calculation requires both a track angle and a drift time as input, we cannot immediately apply apply this calculation to the drift time data, since we don't yet know the track angle. Instead, we must first reconstruct a track using the constant drift-velocity approximation to get a rough angle for the track. Then, using this value for the track angle and the measured drift times, we can correct the drift distances and recalculate the track. In principle we could iteratively repeat this process as many times as we wanted to completely eliminate the error in the constant-v assumption, leaving only the error in the calculation by GARFIELD and the error in the bilinear interpolation between points in the grid of calculated values, however, beyond one or perhaps two iterations of this process, the cost in CPU time probably far outweighs the incremental improvement in precision. When I stopped working on this project in August 2004, I had used GARFIELD to generate a lookup table for time-to-distance conversions, but I had yet to write into the analysis software an algorithm to correct the reconstructed tracks using these values.

Simulation Results The histograms in Figure 13 are of the errors in tracks reconstructed from simulated tracks. The conditions for the simulation were a good-track rate of 1 MHz and a background rate of 1 MHz of background events per plane. The program analyzed 100,000 simulated events. It successfully reconstructed 101,581 good tracks, failed to reconstruct 125 trigger tracks($\approx$ 99.9 % efficiency at 1 MHz!), and reconstructed 2 false tracks. By false tracks I mean tracks reconstructed by the algorithm that do not correspond to any of the actual simulated tracks. This simulation assumed an intrinsic resolution of 200 μm. As can be seen from the histograms, the resulting position resolution from this intrinsic resolution was $(\sigma_x,\ \sigma_y) = (64,\ 144)\ \mu m$, and the angular resolution was $(\sigma_\theta,\ \sigma_\phi) = (.135,\ .288)\ mrad$.

5 Figures

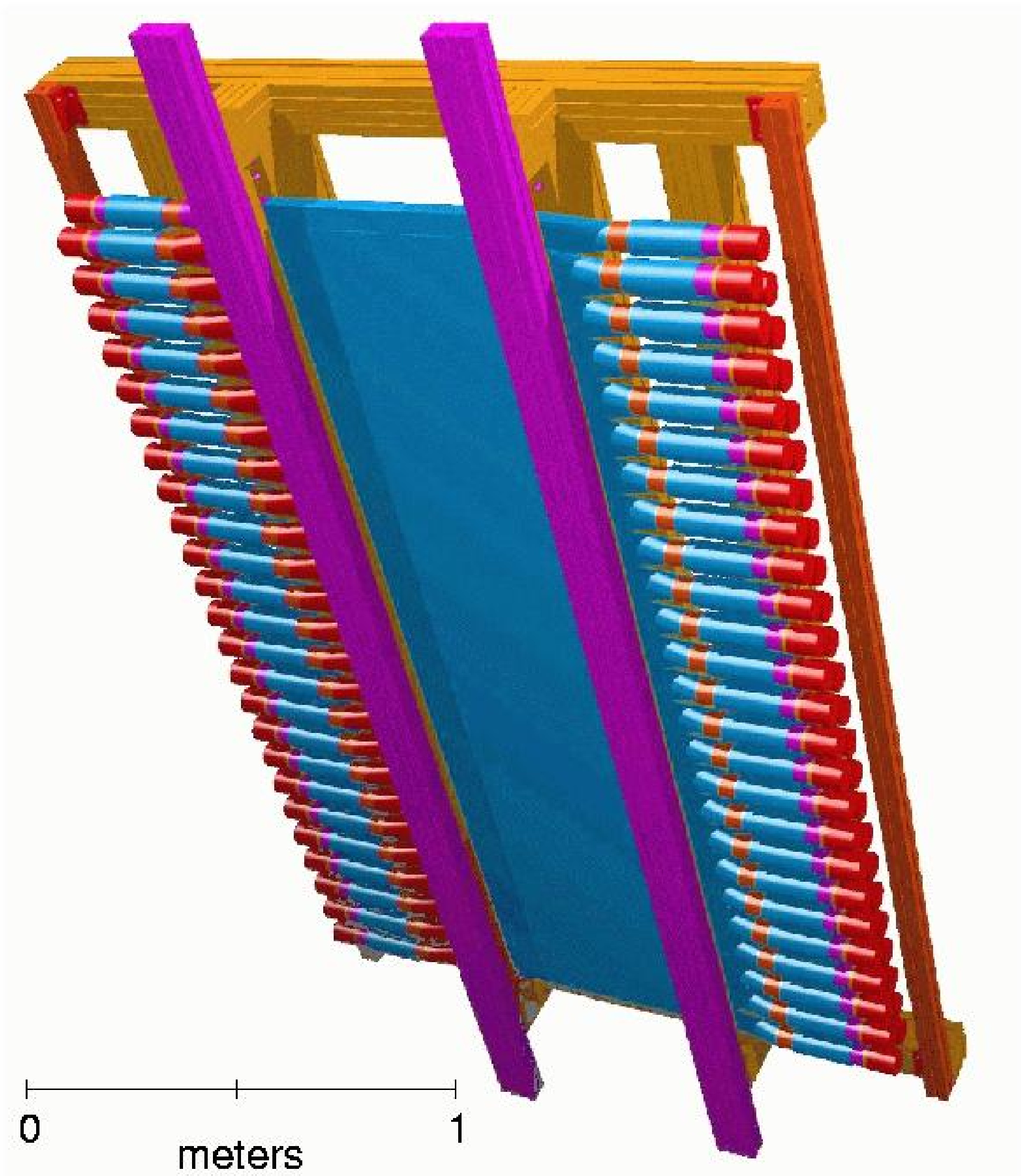

Figure 1: The BigBite trigger plane

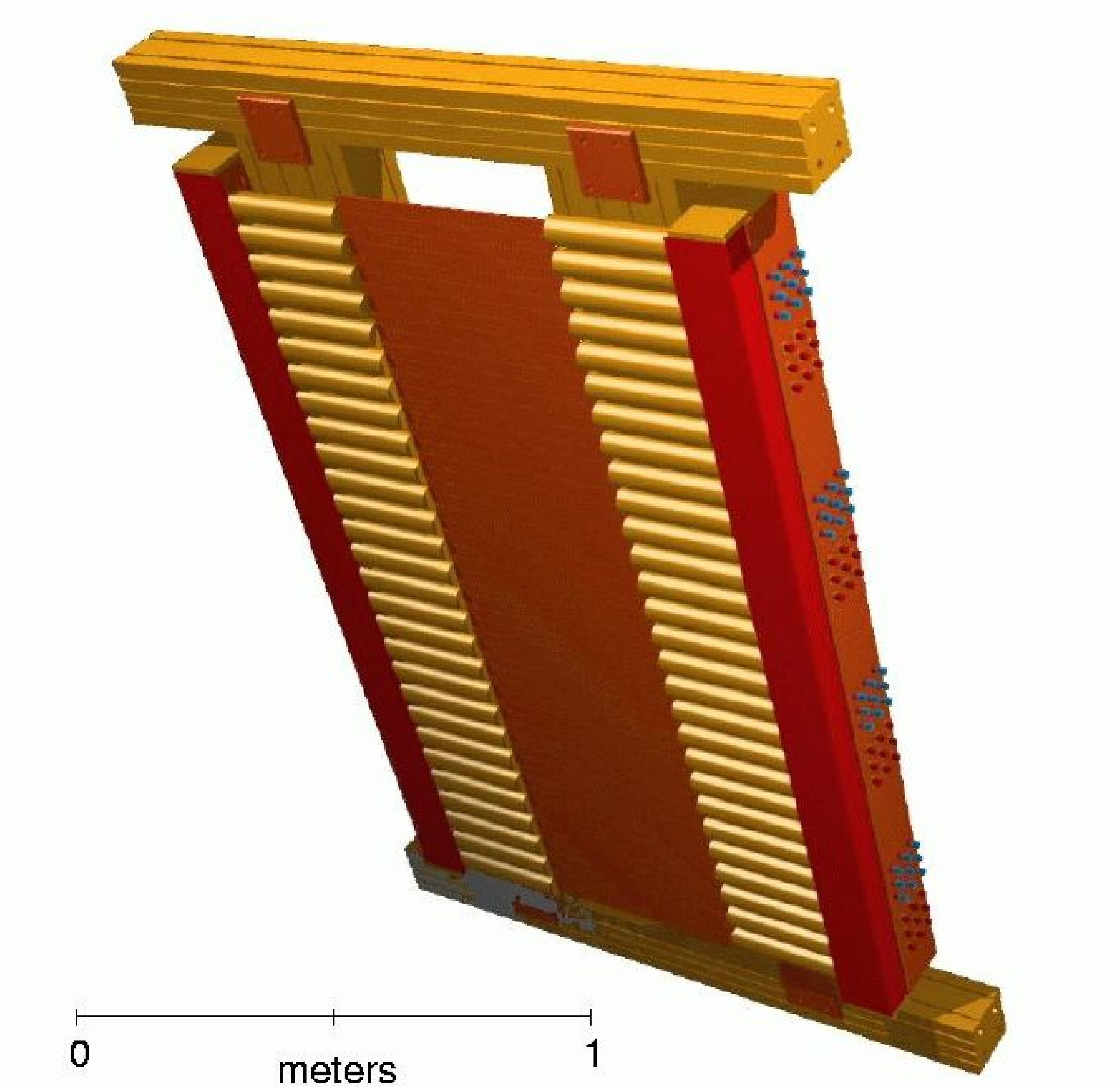

Figure 2: The BigBite auxiliary plane

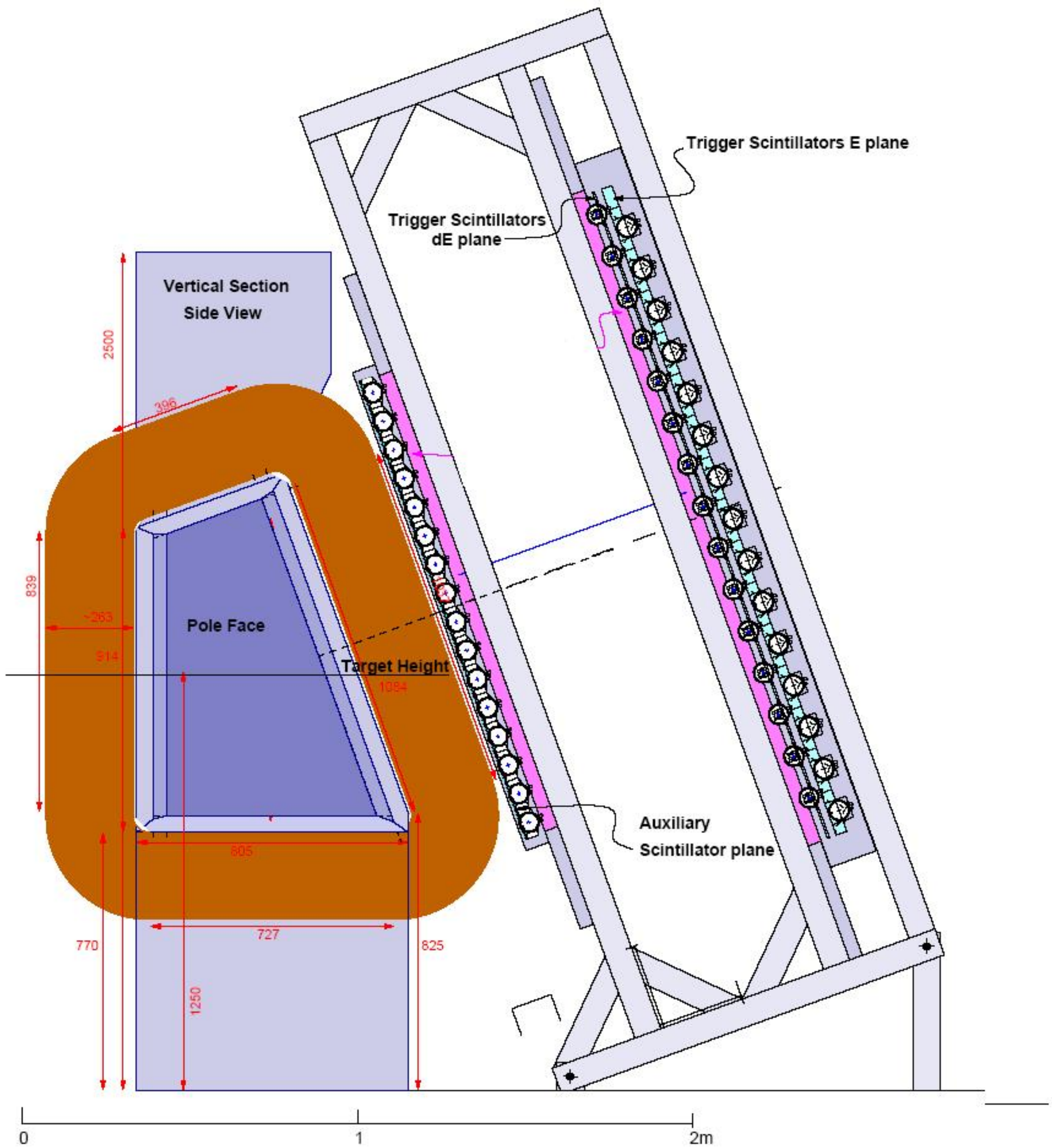

Figure 3: Side view of BigBite with initial detector package

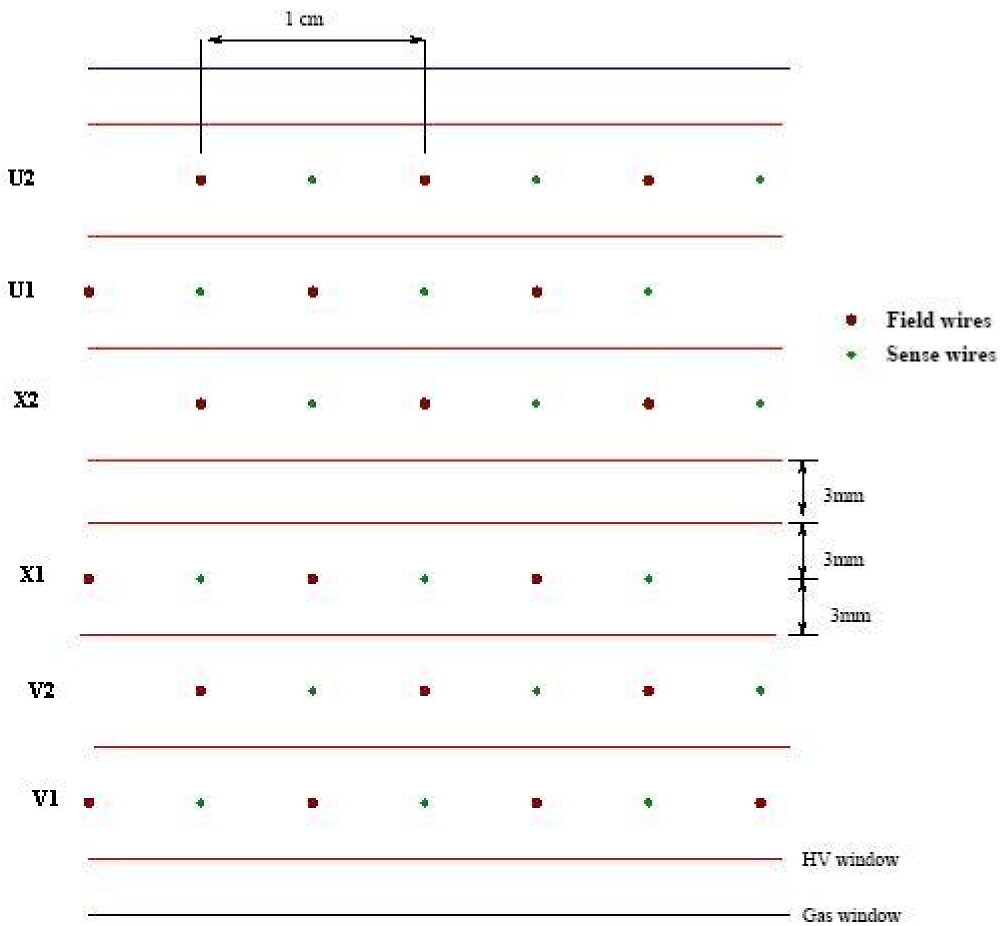

Figure 4: BigBite wire chamber layout.

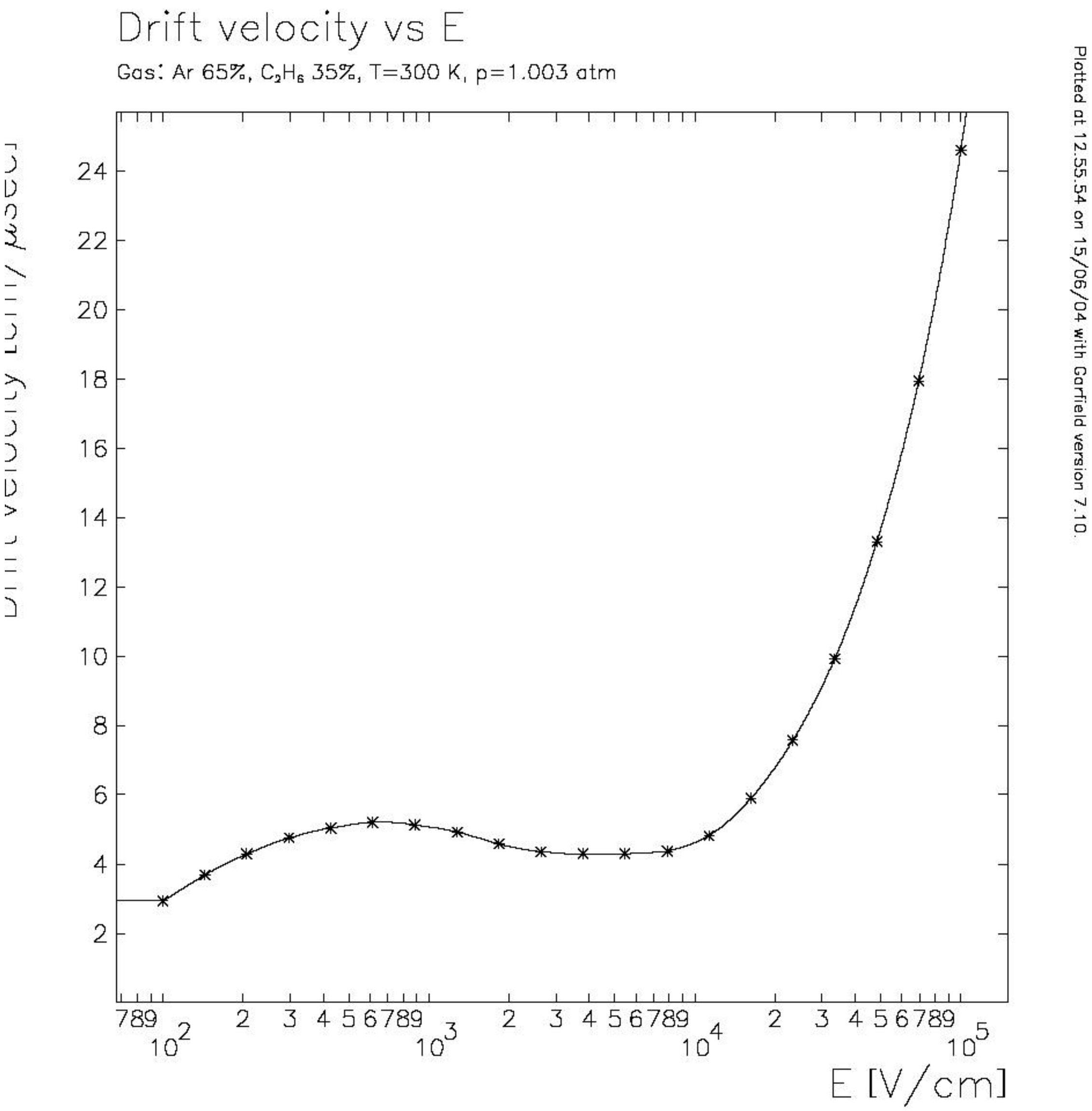

Figure 5: Drift profile for BigBite wirechamber gas mixture calculated by GARFIELD's interface to the Magboltz program

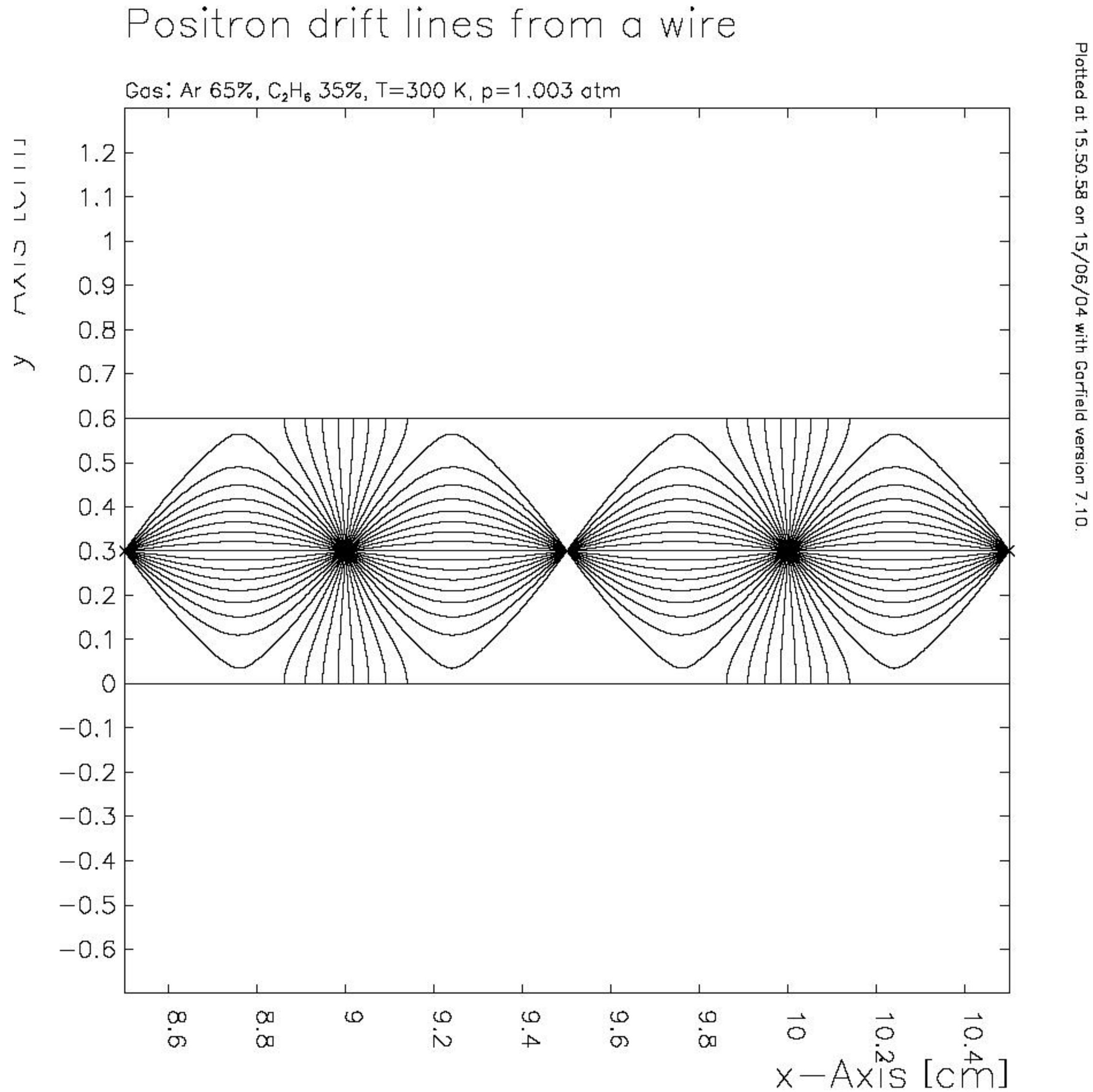

Figure 6: Electric field lines in the BigBite drift chamber

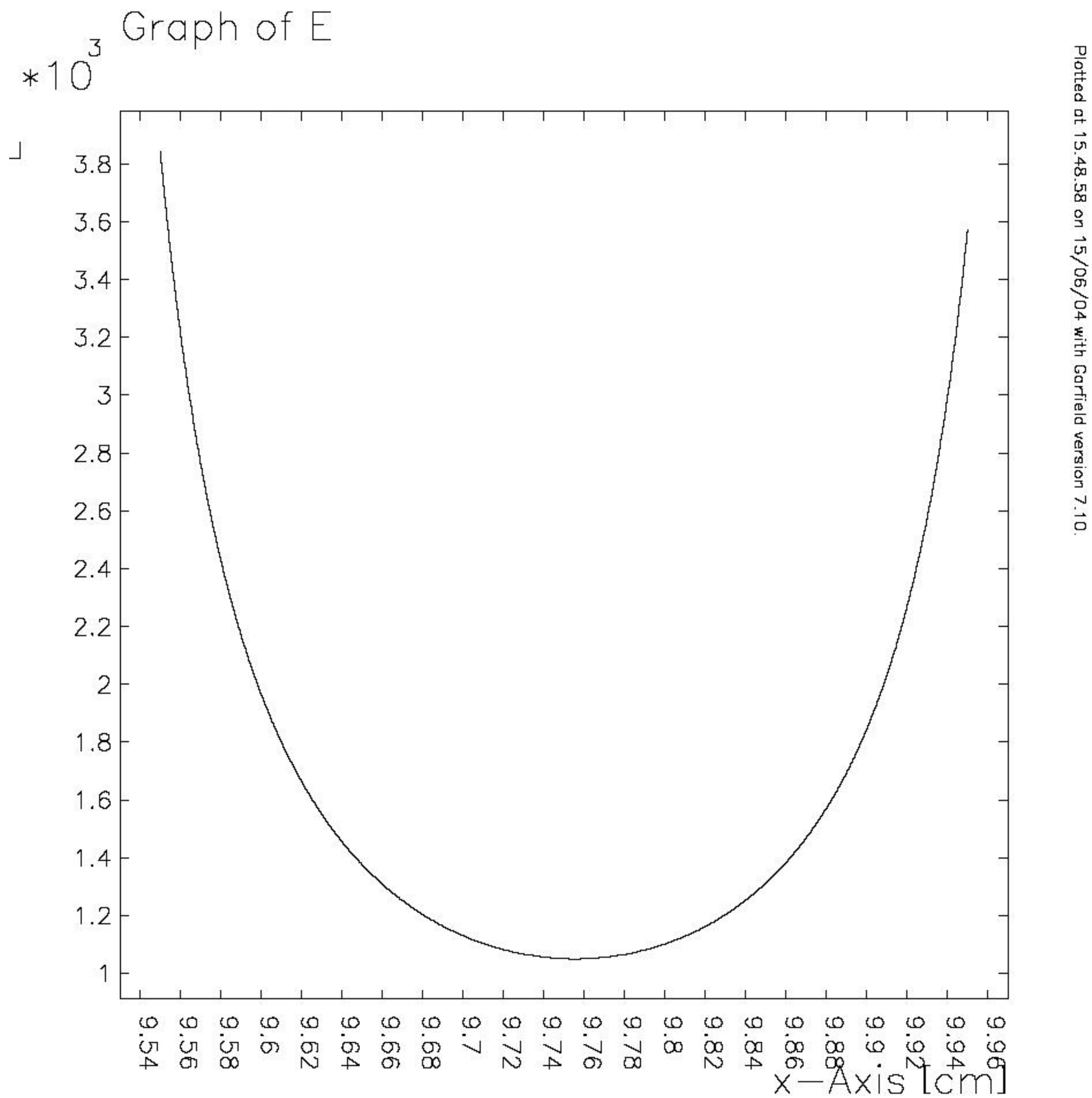

Figure 7: A graph of the electric field magnitude on a path between a signal and sense wire. The steep rise in E very close to the wires is not shown.

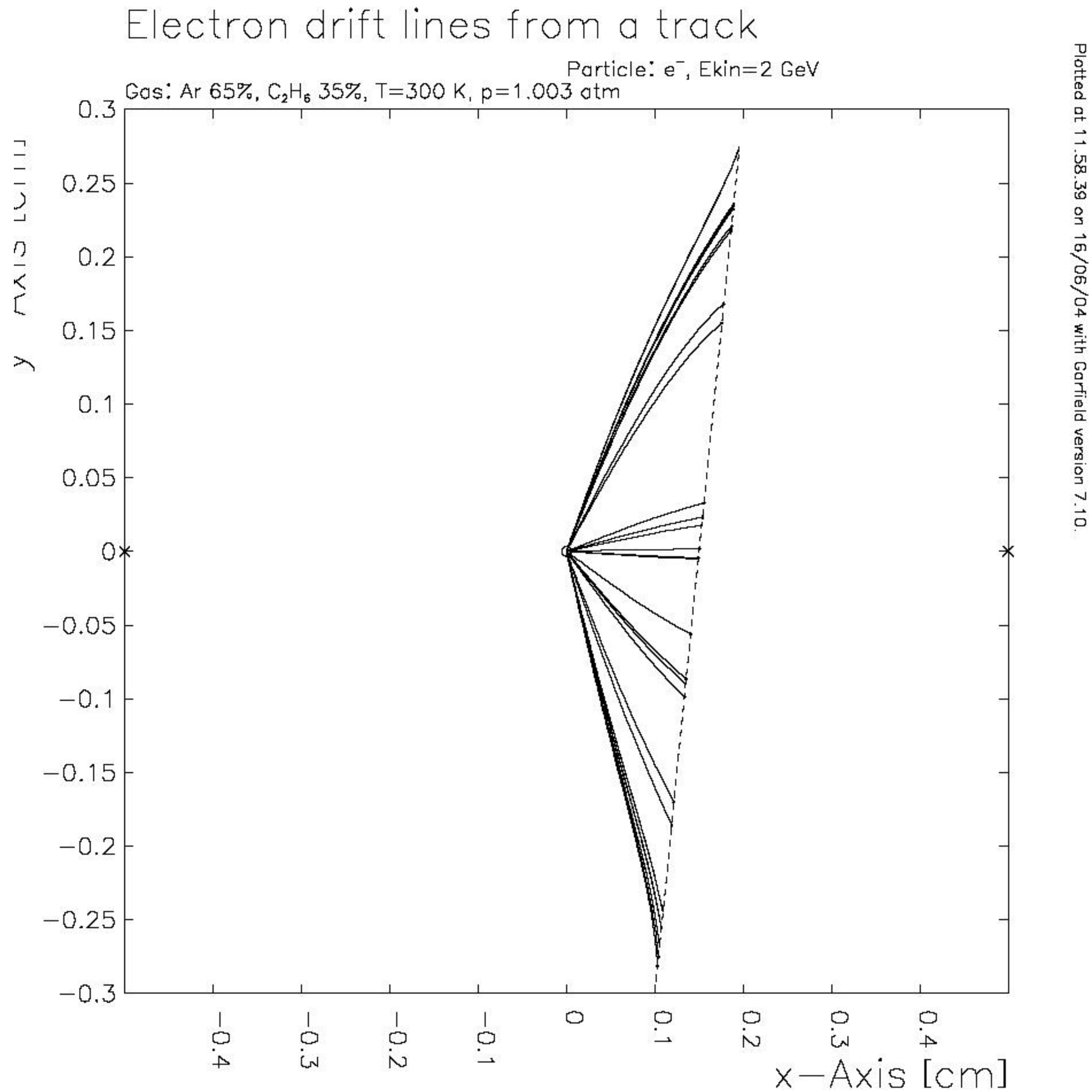

Figure 8: GARFIELD simulation of a 2 GeV electron track going through a plane of the BigBite drift chamber

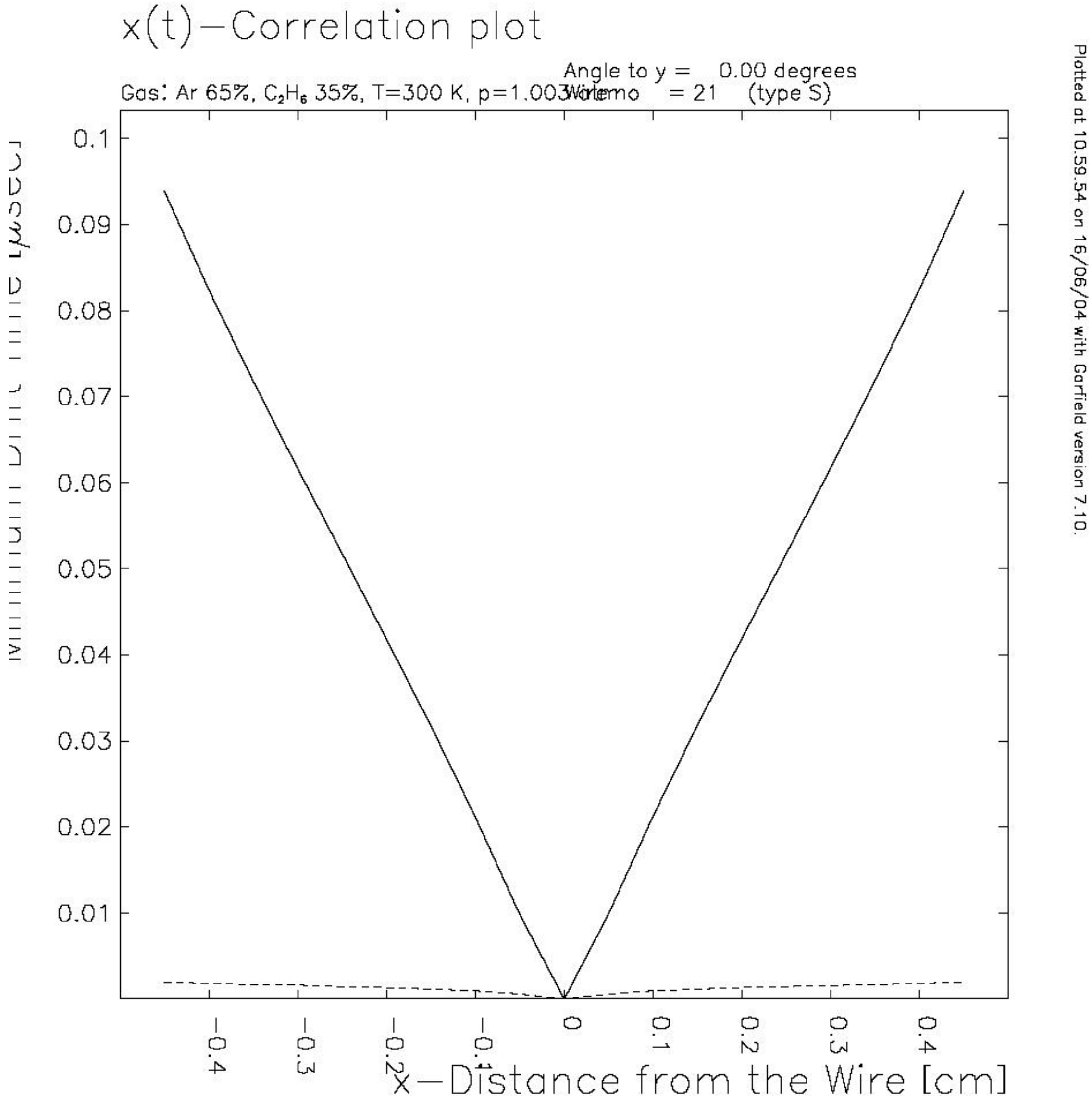

Figure 9: x(t) relation calculated by GARFIELD

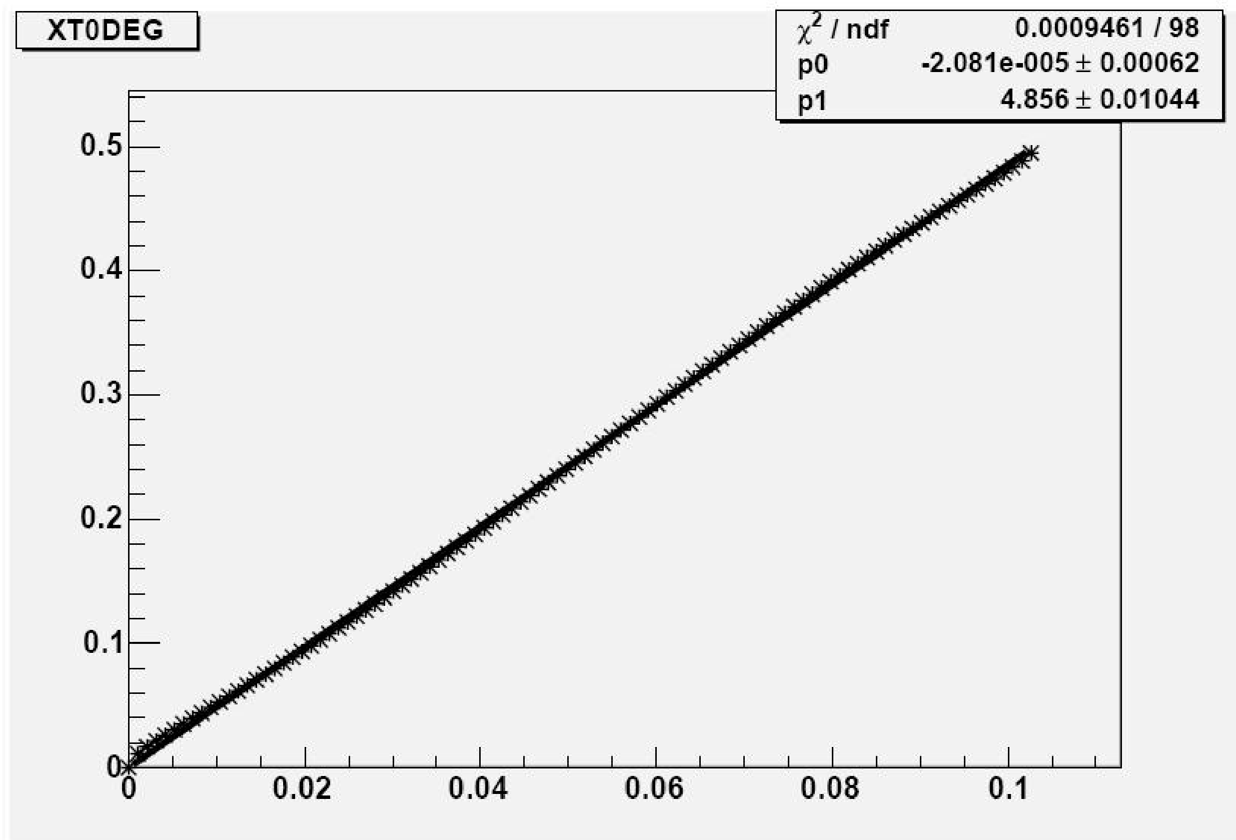

Figure 10: x(t) for a track angle of 0°

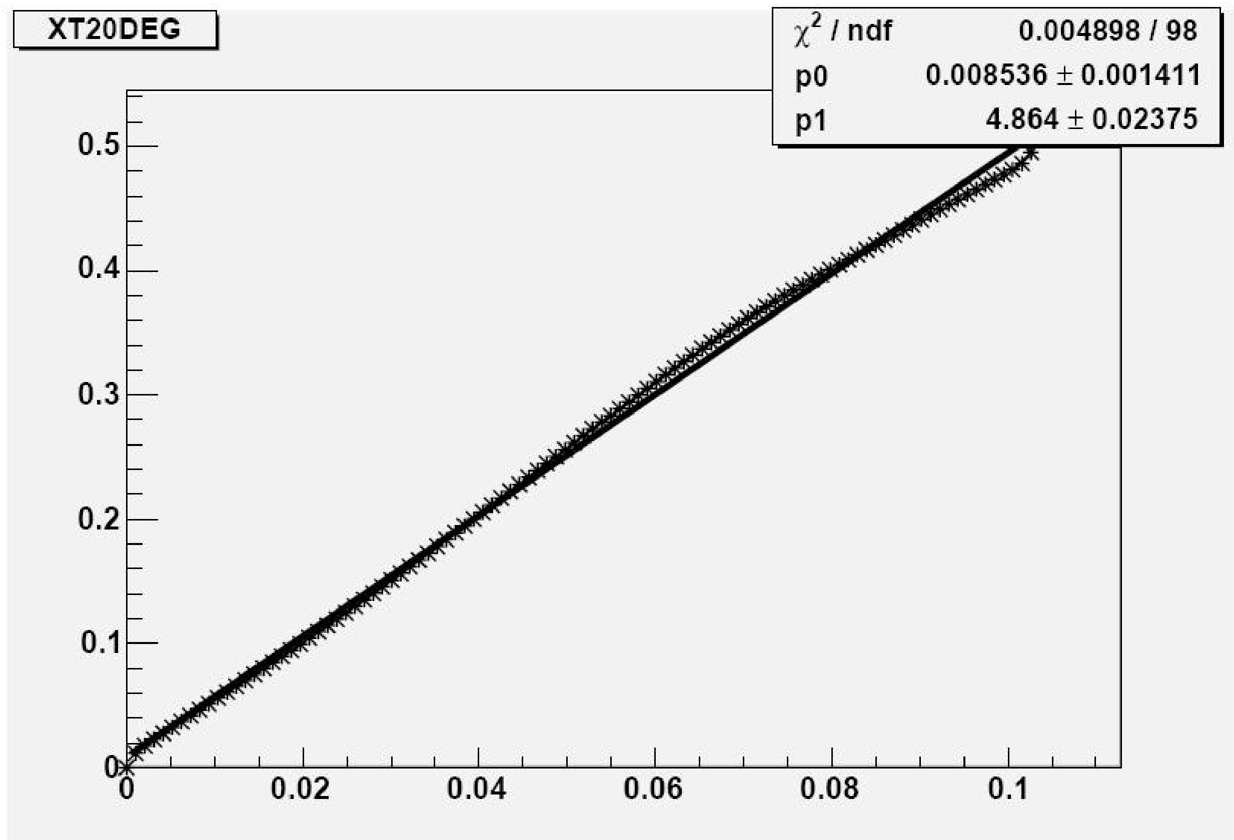

Figure 11: x(t) for a track angle of 20 degrees

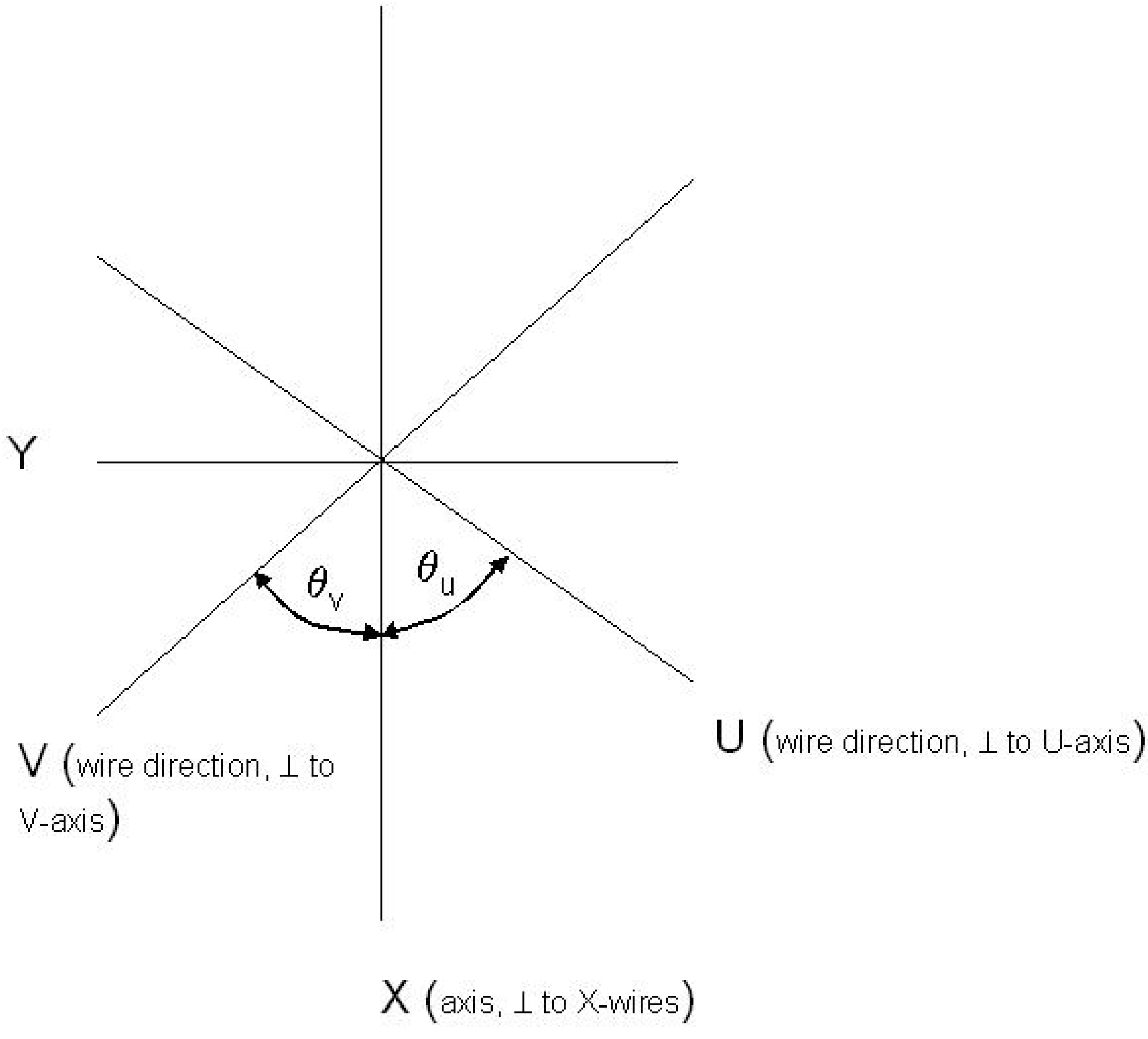

Figure 12: illustration of coordinate system and angles used in rotation matrices

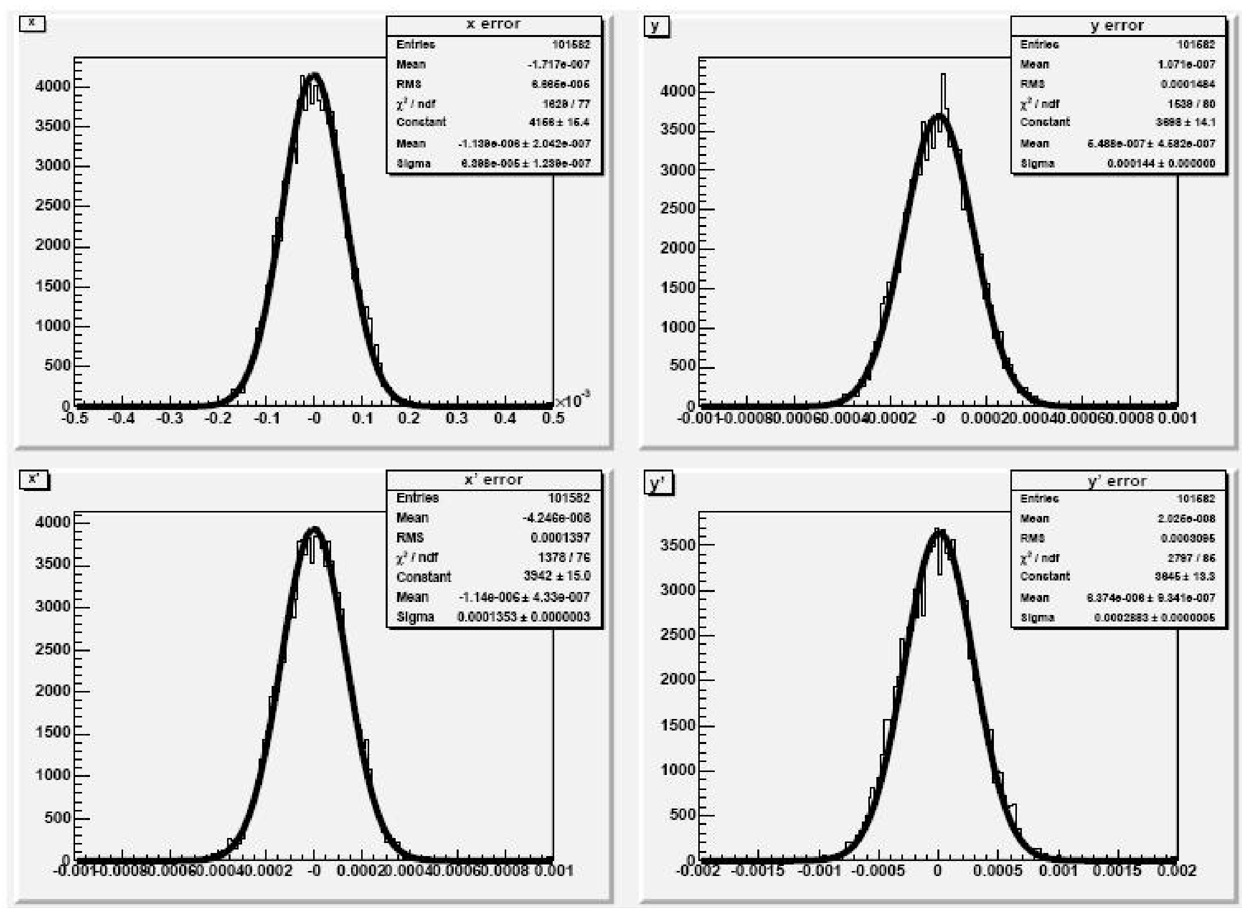

Figure 13: Histogram of errors in $(x,\ y,\ \theta,\ \phi)$ for reconstructed tracks