Software Development in TWIST

Experiment E614

Roy Wilds
Simon Fraser University

Date: August 30, 2001

Supervisor: Art Olin

TRIUMF
4004 Wesbrook Mall
Vancouver, BC

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The *TWIST* (TRIUMF Weak Interaction Symmetry Test) experiment proposes to make the first high precision measurement of the entire differential spectrum (in energy and angle) of positrons from the decay of positive, polarized muons. The goal of the experiment is the precise testing of the structure of electroweak interactions in the Standard Model.

The experiment will consist of having highly polarized "surface" muons (produced from pion decay) entering a large super-conducting solenoidal magnet (2.2 Tesla field) on axis, and then stopping in a thin target at the center of a spectrometer. The positron from the muon decay will then be tracked in the magnetic field, as it passes through the spectrometer.

Chapter 2

$\mathcal{TWIST}$ Experiment at TRIUMF

2.1 Physics Behind the Experiment

Normal decay of the muon is given by:

$$\mu \longrightarrow e\nu\bar{\nu}$$

This is the simplest and most direct test of the weak interaction, since only Leptons are involved in the decay process. For this reason, muon decay experiments are the most precise determinants of the basic weak interaction properties.

In the limit where the electron mass and radiative corrections are neglected, the differential decay spectrum of the muon (called the Michel Spectrum, see figure 2.1) is given by:

$$\begin{aligned}\frac{d^2\Gamma}{x^2 dx d(\cos\theta)} \propto \quad & 3(1-x) + \tfrac{2}{3}\rho(4x-3) \pm \\ & P_\mu \xi \cos\theta \left(1 - x + \tfrac{2}{3}\delta(4x-3)\right)\end{aligned}$$

where θ is the angle between the muon polarization and the outgoing electron, $x = E_e/E_{max}$ (E_{max} is the maximum energy of the electron from the muon decay), and P_μ is the muon polarization. The three Michel parameters ρ,ξ,δ completely determine the spectrum if neither neutrinos nor the spin of the outgoing electron are observed. A fourth parameter η contributes to the energy spectrum when the electron mass is included in the analysis.

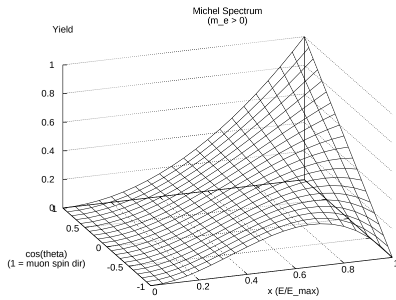

Figure 2.1: Michel Spectrum

$\mathcal{TWIST}$ will measure ρ,$P_\mu\xi$,δ to a precision of a few parts in 10^4 and η to a few parts in 10^3. Note that $P_\mu\xi$ is measured instead of ξ since we have no way of measuring P_μ on its own. Table 2.1 outlines the accepted values, the Standard Model values, and the precision with which $\mathcal{TWIST}$ will measure the 4 Michel parameters.

	Accepted Value	Standard Model Value	TWIST Precision
ρ	0.7518 ± 0.0026	0.75	± 0.0001
δ	0.7486 ± 0.0038	0.75	± 0.00014
$P_\mu\xi$	1.0027 ± 0.0085	1	± 0.00013
η	-0.007 ± 0.013	0	± 0.003

Table 2.1: The accepted values of the Michel parameters along with the Standard Model values.

The current accepted values of these parameters agree with the Standard Model values. $\mathcal{TWIST}$ results will be able to either show that there is evidence for physics beyond the Standard Model, or will place far more stringent limits than possible at present on those extensions to the Standard Model that influence the space-time structure of the weak interaction.

2.2 Experiment Setup

2.2.1 Muon Production

The highly polarized muons that will be used in the $\mathcal{TWIST}$ Experiment are produced using TRIUMF's cyclotron. 500 MeV protons are directed at a Graphite target where they collide with carbon nulcei. Pions (among other particles) are produced. The pions then decay into muons and neutrinos. Muons emmitted into the M13 Beamline, pass through 2 magnetic dipoles, and then exit the beam line and enter the experiment area.

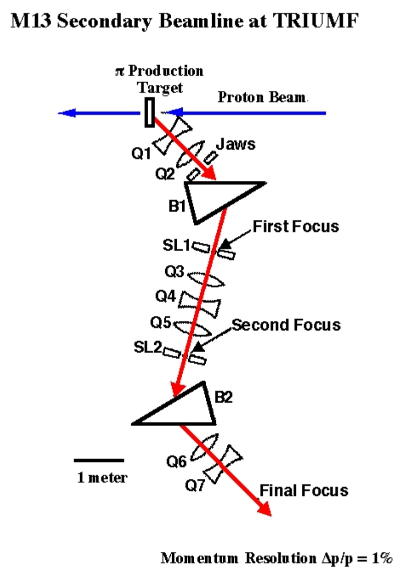

Figure 2.2: M13 Beamline. B1 and B2 are the magnetic dipoles

The magnetic dipoles are used to select particles with a particular momentum. By tuning the field strength, and knowing the radius of the bend in the beam line, we can determine the momentum that a particle needs to make it through the bends in the beam line:

$$F = Q(v \times B) \longrightarrow p = QBR$$

Thus, by this selection procedure, muons of a particular momentum can be directed into the experiment area. It should also be noted here that particles other than muons can make it through these bends, if there momentum satisfies the above equation. However, we aren't interested in these particles, so they will need to be filtered out.

The momentum of the muons is chosen such that muons produced from pions decaying at rest at the surface of the production target will make it through the beam line. These "surface" muons are chosen because they have the high polarization that $\mathcal{TWIST}$

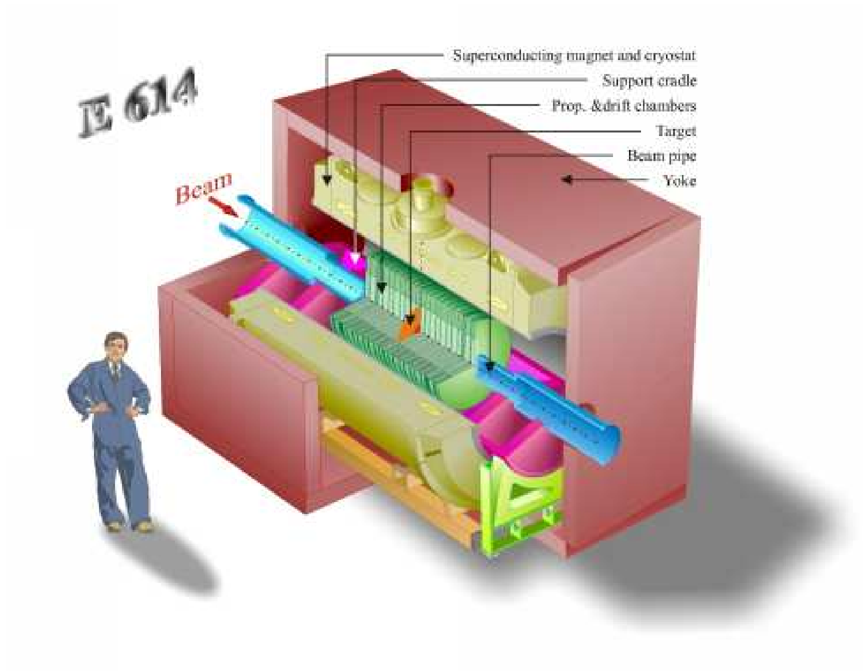

Figure 2.3: The Detector

requires in its experiment. Studies have been done with the experiment setup that will be present, and it has been determined that the optimum momentum is ~ 29.79 MeV/c.

2.2.2 Detector Layout

The detector (see figure 2.3) consists of a super-conducting solenoid inside an iron yoke. Inside the solenoid resides the spectrometer. The M13 beam line extends into the yoke, where an adjustable degrader will be used to place the stopping position of the muons at the target. Downstream of the degrader, the particles pass through a beam scintillator and then enter the spectrometer. The entire detector is built with the goal of having the upstream and downstream ends of the detector symmetric about the target. This symmetry is essential to remove any biases that can occur in the generation of the Michel spectrum, and hence the measurement of the Michel parameters.

The co-ordinate system being used by $\mathcal{TWIST}$ is as follows. The z coordinate is defined as being parallel to the axis of the magnetic field in the solenoid, with increasing z in the direction of downstream. Originally, the y coordinate was defined as being vertical and the x coordinate was defined to make the system right-handed. However, to remove possible biases in the sagging of wires due to gravity, the x and y coordinates were rotated by 45° and renamed u and v. Hence, spatial points are defined by a triplet $\rightarrow$ (u,v,z).

2.3 The Spectrometer

2.3.1 Description

TWIST's spectrometer consists of the following active detector elements:

- 44 Drift Chamber Planes (DC)
- 12 Proportional Chamber Planes (PC)

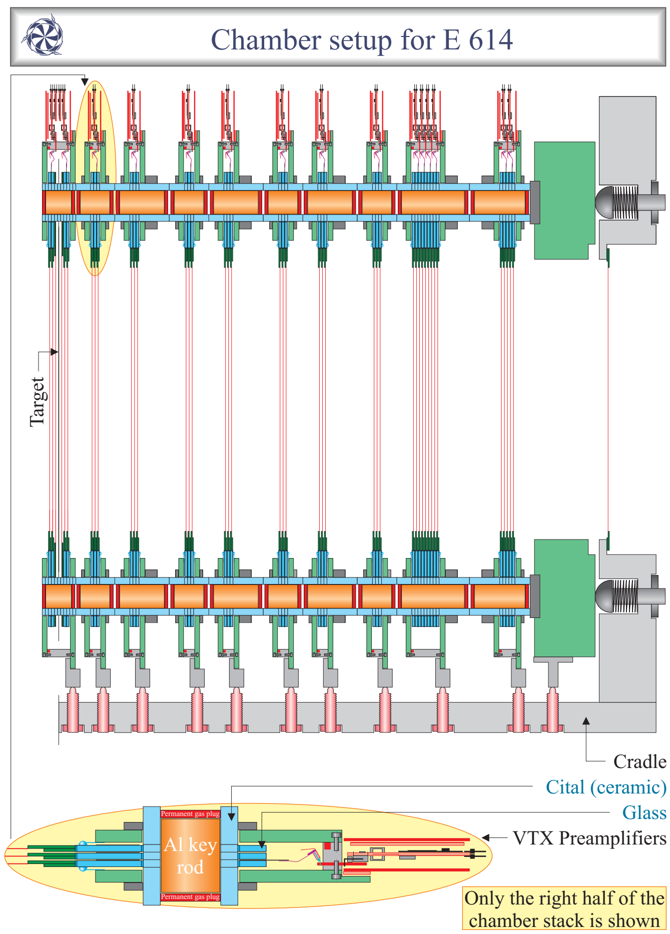

Figure 2.4: Half of the *TWIST* Spectrometer

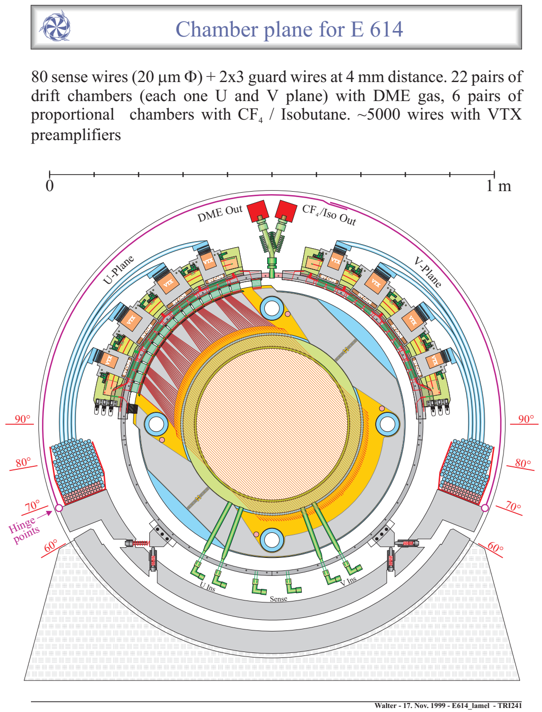

Figure 2.5: Schematic of a Drift Chamber

Drift Chambers

The Drift Chambers contain 80 sense wires strung at 0.4mm spacings. Each Drift Chamber is filled with Dimethyl Ether (DME) which has several advantages as a gas. Among these is its small Lorentz angle, which means that in the presence of a strong magnetic field (which $\mathcal{TWIST}$ has), there will be minimal corrections needed to the Drift Times. This improves $\mathcal{TWIST}$'s ability to accurately know the trajectory that a particle takes through the Drift Chambers.

Proportional Chambers

The Proportional Chambers contain 160 sense wires at 0.2mm spacings. The gas used in the PCs is CF_4-Isobutane which has a very fast drift time (<15 nanoseconds). This permits rapid detection of a particle passing through a PC plane, while the close wire spacing allows the determination of the particle's position to within a few thousand microns accuracy. This is useful in monitoring the position of the incoming beam of muons.

2.3.2 Calibrations

In order to have the spectrometer functioning properly, the analysis software has to understand completely how all the hardware functions. This requires that all detector elements be calibrated properly. The response of a detector element could lead to possible biasing, in the determination of the Michel Spectrum, if we do not understand how it is functioning. Calibrations include: efficiencies of sense wires, differences in cabling lengths for different components, relative sizes of signals, and others. These will be measured and then incorporated in the analysis of the spectrometer data.

Chapter 3

Software in $\mathcal{TWIST}$

A large portion of my time spent with the $\mathcal{TWIST}$ project involved working with the software being used in the experiment. There are 2 main areas in the software group:

Simulation

&

Analysis.

3.1 Simulation

TWIST uses the program GEANT as the basis for its experiment simulation. Incorporated into it are the details and specifics of the experiment, so that the program is able to simulate the actual experiment that will be occurring. This is done for a variety of reasons, but the 2 main benefits of having a simulation are as follows:

- By simulating exactly the output of the detector, the analysis code can be developed and tested prior to the spectrometer being built. This allows bugs and flaws in the code to be found and sorted out before the actual data taking begins
- Possible biases in the design of the detector can be found and corrected for by examining the simulation

3.2 Analysis

Adopting the program KOFIA (Kaon OFfline Interactive Analysis) from the E787 experiment[1], the code was altered to interface with the data produced from *TWIST*'s spectrometer, and renamed MOFIA (Muon OFfline Interactive Analysis) . The workings of the analysis can broadly be characterized as:

1. Decode the dataset (read in the digitized output from the spectrometer or simulation)
2. Filter out noise
3. Sort the information so that track fitting can be performed
4. Fit the tracks
5. Determine the parameters of the tracks, thus producing the desired values (namely the energy and angle of the positron)

Determining calibrations and corrections to the spectrometer is also a large part of the role of the analysis program. Code to calculate plane efficiencies, position corrections and other calibrations is part of the same program that does the physics analysis.

[1]Several memeber of *TWIST* were involved in this experiment

Chapter 4

Projects Worked on

During the 8 months I have spent with the $\mathcal{TWIST}$ project, I have worked on many aspects of the experiment. Software was my main focus, however, I did have the opportunity to spend some time working on the calibrations of some hardware devices. Having had the opportunity to work on several aspects of the experiment provided me with an excellent view of how the experiment will proceed.

I will now describe some of my major contributions to the $\mathcal{TWIST}$ project in more detail, and in the next chapter I will describe in more depth the STR project that I worked on for 2 months.

4.1 Hardware Calibrations

Signals received from the sense wires in the spectrometer are passed through a series of stages before reaching the actual recording device that digitizes the signal. One of these stages is a Post-Amplifier Discriminator (PA). The role of this device is to take a signal from the sense wires, and if it is above a set threshold, amplify the signal and send it on. The threshold that these devices are set to is used to eliminate noise, since we do not want a weak signal caused by some electronic "jitter" or "blip". The setting of the threshold, however, is not exact. For example, in a given Post-Amp the threshold can be set to 150 millivolts, but the actual threshold will be 153 millivolts. This variation is Post-Amp dependent, and we want a consistent setting to be used for all Post-Amps. Due to this offset, it was decided that each Post-Amp would be calibrated using a formula

$$ActualSetting = DemandSetting * A + B$$

where A and B are determined for each Post-Amp. The process of calibrating involved placing the Post-Amp devices into a crate, and then running a test program that ranged the Demand Setting from 0 to 4000 millivolts in 4 millivolts steps, and recorded the actual setting for each input. The data points of Actual and Demand values were then fit using a fitting program, and the parameters A and B were stored in a database for each Post-Amp. While doing these tests, it was discovered that the data points were not fit properly when the full 0 to 4000 millivolts range was incorporated in the fit. It was then decided that since our expected settings would be in the 100 to 500 millivolts range, only the data points lying in the range 0 to 1000 millivolts would be used, and the fits were much better. Figure 4.1 contains a graph of a set of data points and the fit for the data.

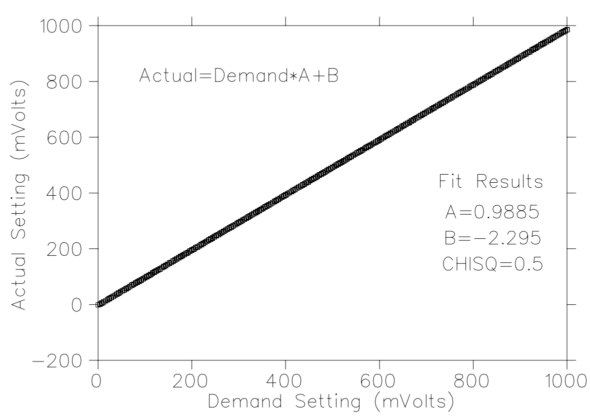

Figure 4.1: Calibration data from Post-Amplifier Discriminator

4.2 Digitization

A lot of the work that I have done with GEANT has dealt with adding features of the experiment to the simulation. One such feature was the way in which the simulated electronics response of the detector (called the digitization) worked.

One of the first problems I dealt with involved putting the Scintillator detectors, that the experiment will have, into the simulation's code, and "wiring" them up. In order to do this, I had to learn how GEANT works with respect to defining materials (the scintillators) and how it characterizes a detector element (ie. how it produces a signal in a scintillator). Once I understood how this worked, I then had to incorporate the additions into the existing code, and then make sure everything was running properly.
Another feature of the simulation that I worked on involved making it more realistic.

In one case I dealt with the problem of a wire having multiple signals triggered on it, which can lead to signal overlap If the signals do overlap then they should not be treated as separate signals, but combined into an appropriate single pulse. This also had added complications in that some wires in the Proportional chambers are Multiplexed[1]. Thus the process of resolving possible signal overlap becomes a bit more complicated. However, this complication was dealt with in a manner that resolved the overlapping of signals properly, and efficiently.

I also worked on adding the effect of sections of wires going "dead". This comes about when a particle deposits a large amount of charge on a section of a wire, and the wire becomes saturated. Once this happens, then for a period of time afterwards, that section of the wire will not send a signal if more charge is deposited on it.

4.3 Code Management

In addition to developing new features in GEANT I was also involved in releasing a new version of the simulation to the $\mathcal{TWIST}$ software group. This involved merging code that was developed by other people with the code I had developed. Merging code from different sources together was especially interesting when being done with the aid of a program (such as Concurrent Versions System (CVS)). Linking together different logic and algorithms resulted in myself re-designing various routines involved. The end result though, was a very functional simulation that ran better than before.

[1]Multiple wires connected to a single output channel

4.4 Analysis Development

Much of the work I did on the analysis software dealt with interpreting the changes that were made to the simulation. Introducing code to store information from scintillator output was one of the first tasks I was given. This was an especially good choice since I was the one that added this aspect to the simulation. I was required to develop my own structures and functions/procedures to operate on the structures that would be used for the scintillator data. Shown in figure 4.2 is the scintillator structure that was used for defining the physical characteristics of the scintillators in MOFIA.

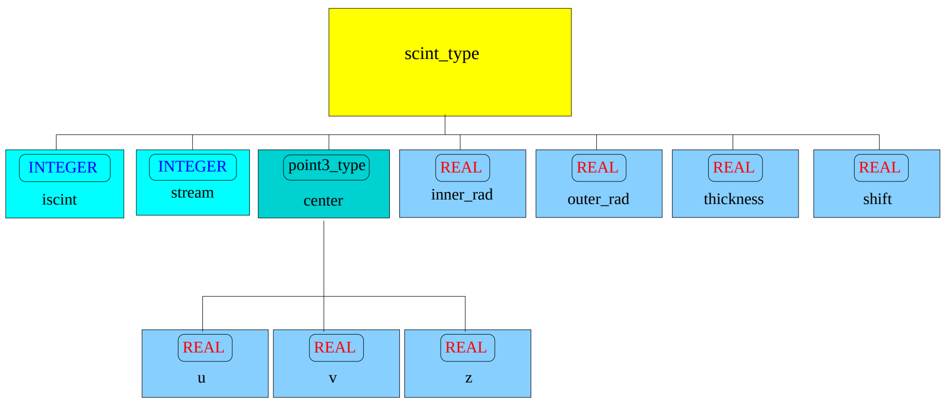

Figure 4.2: Scintillator Geometry Structure

The components of this structure are as follows:

iscint	$\rightarrow$	ID # for scintillator
stream	$\rightarrow$	Flag for upstream or downstream designation
center	$\rightarrow$	(u,v,z) co-ordinates at center of disk
inner_rad	$\rightarrow$	Inner radius of scintillator disk
outer_rad	$\rightarrow$	Outer radius of scintillator disk
thickness	$\rightarrow$	Thickness of scintillator disk
shift	$\rightarrow$	Correction to nominal position in z

Table 4.1: Scintillator Structure Components

Another problem that I worked on involved developing code to determine the resolution of the spectrometer. This involved making histograms and then fitting them with a Gaussian function to determine the standard deviation of the distribution. Fitting with a Gaussian did not always work, so alternative methods were developed, such as finding the Full Width at Half Max (FWHM) and finding the standard deviation from that by virtue of σ=FWHM/2.35. I developed the code to perform these calculations, and also incorporated code to allow a user to choose between fitting methods, into the analysis program.

A major undertaking of mine was the re-designing of the method of how STRs are interpreted in MOFIA. Originally files were read in that were in a format that was directly usable by the rest of the analysis code. However, once it was decided to have both the simulation and analysis programs use the same STR input file, this format was no longer available. Hence, I programmed an interpretation method that would take the new input files and generate the format that the analysis required. This was achieved by adding a module to the existing code that took care of this intermediate step. In the following chapter I describe the work done here in more detail.

Chapter 5

Focus on STR Project

5.1 Space Time Relations

Both the Simulation and the Analysis programs make use of Space Time Relations (or STR for short). The STRs are a means of determining the time it takes ions/electrons in a gas to drift in the presence of an electric and magnetic field, based upon there starting position. This is the basis of the Drift Chambers being used in our experiment, since when a charged particle passes through a Drift Cell (see figure 5.1) it ionizes some of the gas present and these ions then drift to the sense wire. By recording the time it takes these ions to reach the sense wire, we can use the STRs to determine at what position in the Drift Cell the ions were created, and thus where the particle passed through the cell. To obtain the STRs, a program called *GARFIELD* [1] was used. It is a simulation that produces a variety of values describing the behavior of ions/electrons in a gas.

[1] Developed at CERN. See http://consult.cern.ch/writeup/garfield for more information

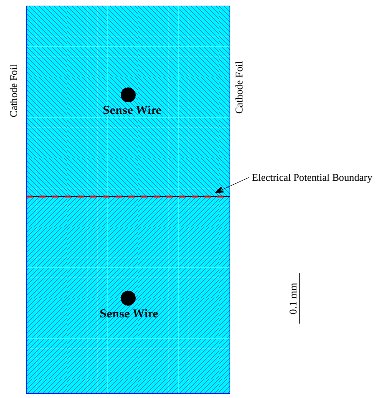

Figure 5.1: Schematic of 2 Drift Cells

In GEANT the STRs are used to determine what Drift Time is appropriate as a particle passes through a Drift Cell, while the Analysis does the reverse procedure; determining what position in the Drift Cell corresponds to the recorded time that the ions took to drift. These 2 different applications require different formats for the STR information that they each make use of. Originally a total of 4 files were needed to represent the STRs for a given combination of gas, pressure, temperature, electric field, and magnetic field. I was given the task of reducing the number of files necessary, so as to make the generation of various STRs easier and more efficient.

In the next chapter, I present a more thorough description of STR use in the $\mathcal{TWIST}$ experiment, and the work I contributed to them.

5.2 Description of *GARFIELD*

GARFIELD has many different features for determining the properties of a Drift Chamber Cell. These include Ionization Tables, Diffusion Tables, Space-Time Relations, and many others. The one that we are chiefly interested in, with respects to the $\mathcal{TWIST}$ experiment, is the Space-Time Relation of charge clusters. By defining the Drift Cell geometry, gas properties, electric and magnetic fields in the *GARFIELD* simulation we obtain the time that an ion takes to drift from a point in the cell to the wire (called the Drift Time). *GARFIELD* allows the user to define a grid in the cell, and the simulation then outputs the Drift Time for each grid point in the cell. This output is what GEANT requires. As well, instead of defining a grid in the cell it is possible to specify a track that a particle takes through the cell, and the minimum Drift Time along this track is computed by *GARFIELD* . This data is what the analysis software made use of.

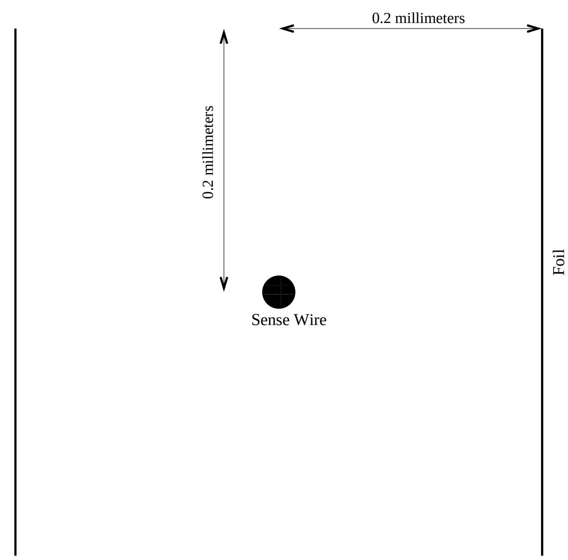

Figure 5.2: Close up of a Drift Cell

5.3 Modification of Source Code

One of the first issues that I resolved was increasing the grid density that was allowed in *GARFIELD* . In the distributed *GARFIELD* package, a maximum grid density of 50 by 50 points is allowed. The Drift Cell that we wish to calculate the STRs for, is 0.4mm square. By virtue of symmetry, the Drift Time only actually needs to be looked up for 1 quadrant of the cell. The desired spacing to be used for the STRs was determined to be 10 microns between grid points. However, with the limit of a 50 by 50 grid in a 0.2mm square cell, it was not possible to achieve this 10 micron spacing directly (it would produce a 40 micron spacing). To solve this restriction, previous calculations of the Drift Time grid were done by calculating 5 grids each with a 40 by 40 spacing, with each grid offset by 10 microns from the other. By combining these 5 grids the 10 micron spacing was obtained.

By increasing this maximum grid density parameter to 200, it was possible to avoid the 5 separate grid calculations and simply use 1. After a brief investigation into using a denser grid, it was decided not to pursue this as the computation time was increased and the benefits from the finer grid were determined to not be that great. If at a later date this issue is revisited then changes will be relatively easy to implement.

In conjunction with adjusting the grid parameter to allow for 1 grid to be defined and outputted, the format of the resulting data was changed. Previously the data was output in block sections (see figure 5.3). This was not a very intuitive way of viewing the data so I altered the format to printout the data in strips (see figure 5.3). This format was also the same as that required by GEANT for reading in the Drift Times.

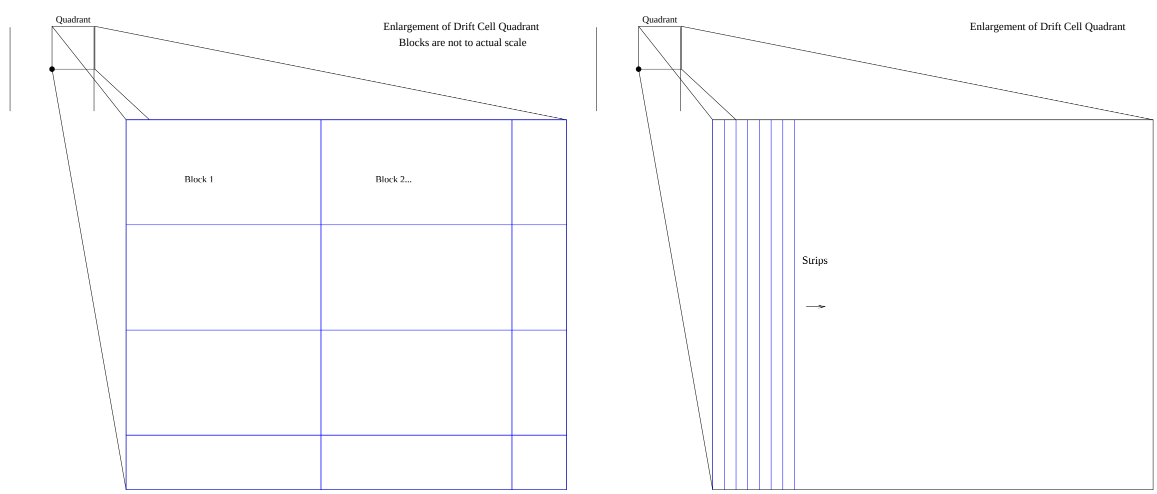

Figure 5.3: Block format of data output from old *GARFIELD* and Strip format from new *GARFIELD* (enlarged for effect)

5.4 Alteration of Analysis Software

To use the altered *GARFIELD* output format, the method of reading the STR files into MOFIA had to be adjusted. As well, the calculation of STRs in MOFIA had to be adjusted. Unlike GEANT, MOFIA does not take a grid of drift times for STR calculations. Instead, the angle of a track going through a cell and the minimum distance from the wire to the track in the cell are related to the Drift Time (see figure 5.4).

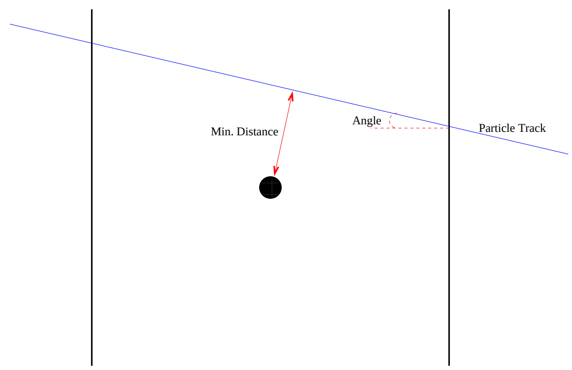

Figure 5.4: Track going through a Drift Cell

Thus when using the Drift Time grid in MOFIA, a "conversion" is necessary to obtain the format needed by the MOFIA code. The method of this conversion that I developed is as follows (see figure 5.5 for flowchart):

1. Read in the Grid of Drift Times
2. Cycle through specified angles
    (a) For each angle Generate Tracks at specified distance intervals
        i. Determine the Minimum Drift Time for each track
        ii. Store the angle, Minimum Drift Time and Minimum Distance to the track
    (b) Repeat for all Distances wanted
3. Repeat for all Angles wanted

This method was much quicker than the old method of having *GARFIELD* generate this data. As well, it was possible to have smaller distances between adjacent tracks, which produced much better resolution than before. Additionally, using this method it is possible to generate tracks at any chosen angles using the same STR grid file (previously, files had to be generated by *GARFIELD* for each angle, which was very slow).

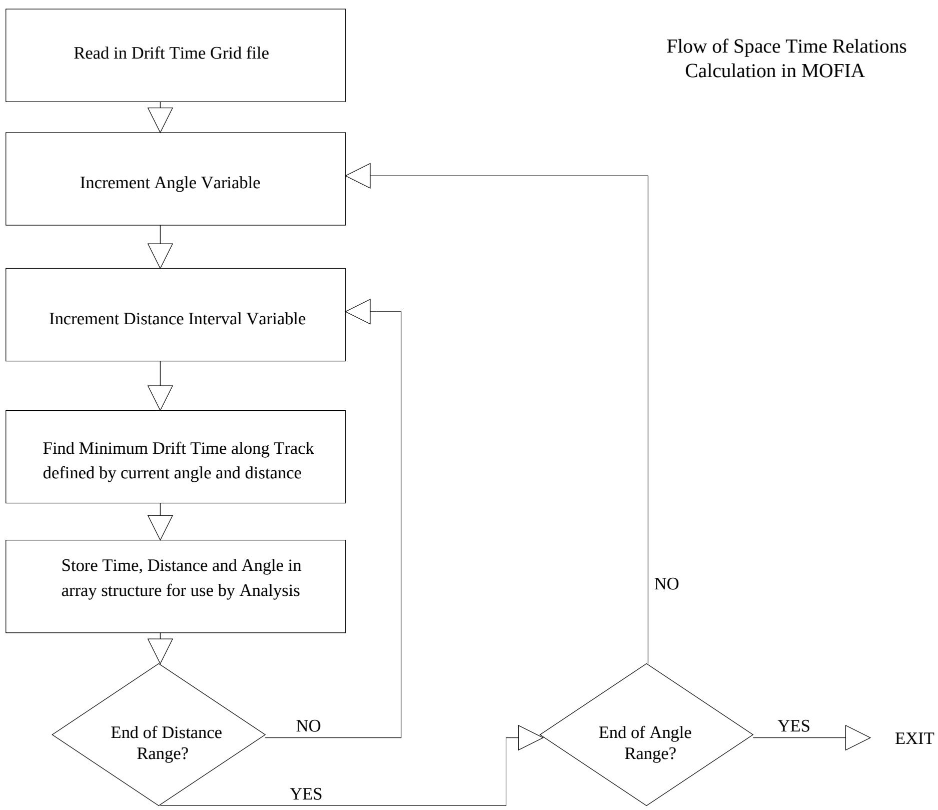

Figure 5.5: Flowchart of STR calculations in MOFIA

5.5 Analysis of New Method

After sorting out some rudimentary problems with the code that was developed for MOFIA, I began comparing the results being produced with those from the old method. Figure 5.6 shows a histogram of the differences.

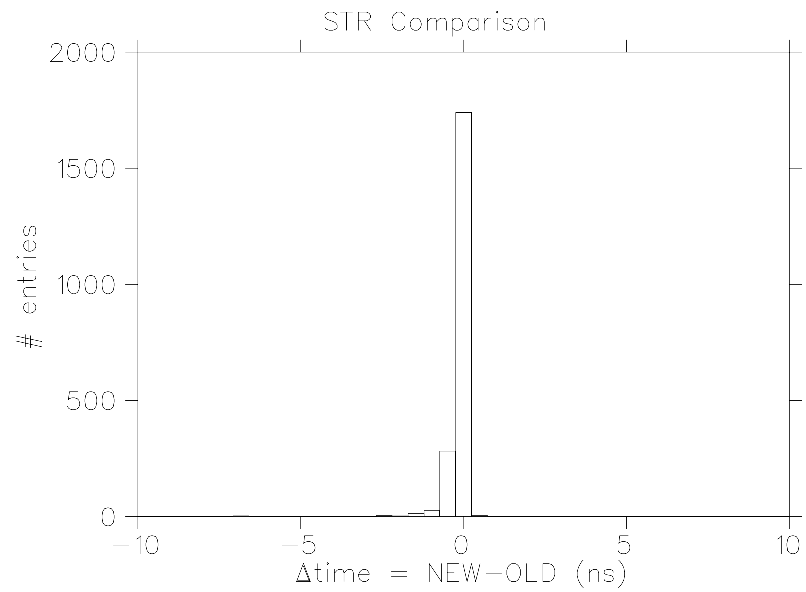

Figure 5.6: Comparison of STR generation using NEW method and OLD method

This histogram was constructed by comparing the Minimum Drift Times (obtained by the 2 different methods) for tracks at the same Angle and Distance. The difference is extremely small, and was determined to be well within acceptance. I then proceeded to give a short presentation on the new method to the $\mathcal{TWIST}$ group at the June 2001 collaboration meeting that was held[2]. The code was then integrated into the general usage of the Analysis code, and has been functioning fine.

[2]Refer to http://e614slow.triumf.ca/e614/offline/garfield/talk.ps

Chapter 6

Summary

Much of the work that I did with the $\mathcal{TWIST}$ project was solving small problems that popped up. Hence, many of them are not present in this report, as they are trivial to speak of. However, dealing with these many little problems as I worked on other tasks helped me learn more about the way all the software functioned as a whole.

Working on both ends of simulation and analysis enabled me to effectively implement additions to the code that would be used on both ends. It also provided me with an excellent understanding of how both the Simulation and Analysis are designed to function, since in many respects, they are complementary to each other.

As well, my experience in calibrating the Post-Amps gave me insight into how calibrations are incorporated into the running and analysis of an experiment.

Chapter 7

Acknowledgements

I would first and foremost like to thank David Gill for giving me the opportunity to work with the $\mathcal{TWIST}$ project. It has been an amazing 8 months, in which I have learned more than I could possibly have hoped.

During my time with $\mathcal{TWIST}$ I have been supervised by several excellent members of the collaboration. Many thanks to Renee Poutissou, Peter Gumplinger and Maher Quraan for their guidance. I would especially like to thank Art Olin for his supervision and for handling my co-op requirements.

I would also like to thank Jim Musser, Rob MacDonald and Andrei Gaponenko for being patient and helpful with many of my questions.

And I would like to thank Nate Rodning, for giving me a good understanding of the $\mathcal{TWIST}$ project as a whole, and for making me feel very welcome in the group.

Best of luck to all the members of $\mathcal{TWIST}$ in the future.