Monte Carlo Simulation of the OLYMPUS Experiment

by

Nicholas Lee Harrington

Submitted to the MIT Department of Physics
in partial fulfillment of the requirements for the degree of

Bachelor of Science in Physics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2009

Author .....
MIT Department of Physics
May 8, 2009

Certified by .....
Richard Milner
Professor
Thesis Supervisor

Accepted by .....
David E. Pritchard
Professor

Monte Carlo Simulation of the OLYMPUS Experiment

by

Nicholas Lee Harrington

Submitted to the MIT Department of Physics
on May 8, 2009, in partial fulfillment of the
requirements for the degree of
Bachelor of Science in Physics

Abstract

The OLYMPUS experiment seeks to measure the ratio of the cross sections for e^--p and e^+-p scattering in order to determine the magnitude of two photon interactions in lepton nucleon scattering. Measuring this observable to the accuracy required is dependent on a good understanding of the systematic uncertainties associated with the scattering experiment. To accomplish this, a simulation using the GEANT4 library and reconstruction code was written and studies were performed. This paper serves to document the software written and its use in understanding the experiment and some systematic uncertainties.

Thesis Supervisor: Richard Milner
Title: Professor

Acknowledgments

Thanks to Douglas Hasell and Richard Milner for putting up with the endless delays in the software development process and to Chris Crawford and Michael Kohl for providing incredibly useful advice. Beyond the work, thanks to TB, DVP, BP, Josh Winn, Joe Formaggio, John Belcher, Lauren and Karen.

Contents

List of Figures

List of Tables

Chapter 1

Physics and Hardware

1.1 Physics

The OLYMPUS experiment has the goal of measuring the ratio of the cross sections for e^--p and e^+-p scattering. Traditional measurements of the ratio of the Sach's form factors with Rosenbluth scattering techniques differ strongly from recent polarization measurements. These inconsistencies are explained by including two photon interactions in the elastic scattering cross section. The ratio of e^-p to e^+p cross sections is sensitive to the two photon contribution[35].

1.1.1 One Photon e-p Elastic Scattering

If we model the proton as a point charge, the differential scattering cross section for the electron in e-p elastic scattering can be calculated using QED as:

$$\sigma_{pl} \equiv \frac{d\sigma_{pl}}{d\Omega} = \frac{E'_e}{E_e} \frac{\alpha^2 \cos^2 \frac{\theta_e}{2}}{4E_e^2 \sin^4 \frac{\theta_e}{2}} \tag{1.1}$$

The proton is, however, very poorly approximated as a point particle. We can parameterize the electromagnetic charge and current densities with two form factors, $G_{E_p}(Q^2)$ & $G_{M_p}(Q^2)$. These Sach's form factors appear as observables in the elastic scattering cross section. In the non-relativistic limit, they are the Fourier

transforms of the spatial distributions of charge and magnetization[11].

Before we get to the scattering cross section formula, here are a few definitions.

Take k_μ as the initial 4 momentum of the electron and k'_μ as the final. Let $q_\mu = k'_\mu - k_\mu$. For spacelike photons, we use this to define a further value parameterizing the momentum transfer, $Q^2 = -q^\mu q_\mu \approx 4E_e E'_e \sin^2\theta/2$ where E_e is the initial energy of the electron and E'_e is the final. If we set $\tau = Q^2/4M_p^2$, we can define the virtual photon polarization as $\epsilon = \left[1 + 2\left(1+\tau\right)\tan^2\left(\theta_e/2\right)\right]^{-1}$[24, 14]. With all of this notational machinery in place we can write the one photon differential scattering cross section for elastic e^--p scattering (or e^+-p scattering for that matter) as:

$$\frac{d\sigma}{d\Omega} = \left(\frac{\sigma_{pl}}{1+\tau}\right)\left[G^2_{E_p}\left(Q^2\right) + \frac{\tau}{\epsilon}G^2_{M_p}\left(Q^2\right)\right] \tag{1.2}$$

[24, 14] The form factors are normalized such that $G_{M_p}(0) = \mu_p$ and $G_{E_p}(0) = 1$, where μ_p is the proton magnetic moment. This differential scattering cross section is labeled the Rosenbluth scattering cross section because of Rosenbluth's 1950's paper on the topic [37].

Measuring this differential scattering cross section at different beam energies and scattering angles provides a means of measuring the electric and magnetic form factors at a given value of Q^2. Historically, this has been the most popular method for measuring the cross section. Developments of polarized beams, targets and polarimeters has allowed for a different measurement of these form factors in the 1990s[34].

If a polarized electron beam strikes an unpolarized hydrogen target, measuring the polarization of the scattered proton yields the ratio of the form factors. For a transverse polarization, P_t, and a longitudinal polarization, P_l, of the proton in the scattering plane, we relate the ratio of the polarizations to the ratio of the form factors as follows:

$$\frac{\mu G_{Ep}}{G_{Mp}} = -\mu_p \frac{P_t}{P_l}\frac{\left(E_e + E'_e\right)}{2M_p}\tan\frac{\theta_e}{2} = -\mu_p\sqrt{\frac{\tau\left(1+\epsilon\right)}{2\epsilon}}\frac{P_t}{P_l} \tag{1.3}$$

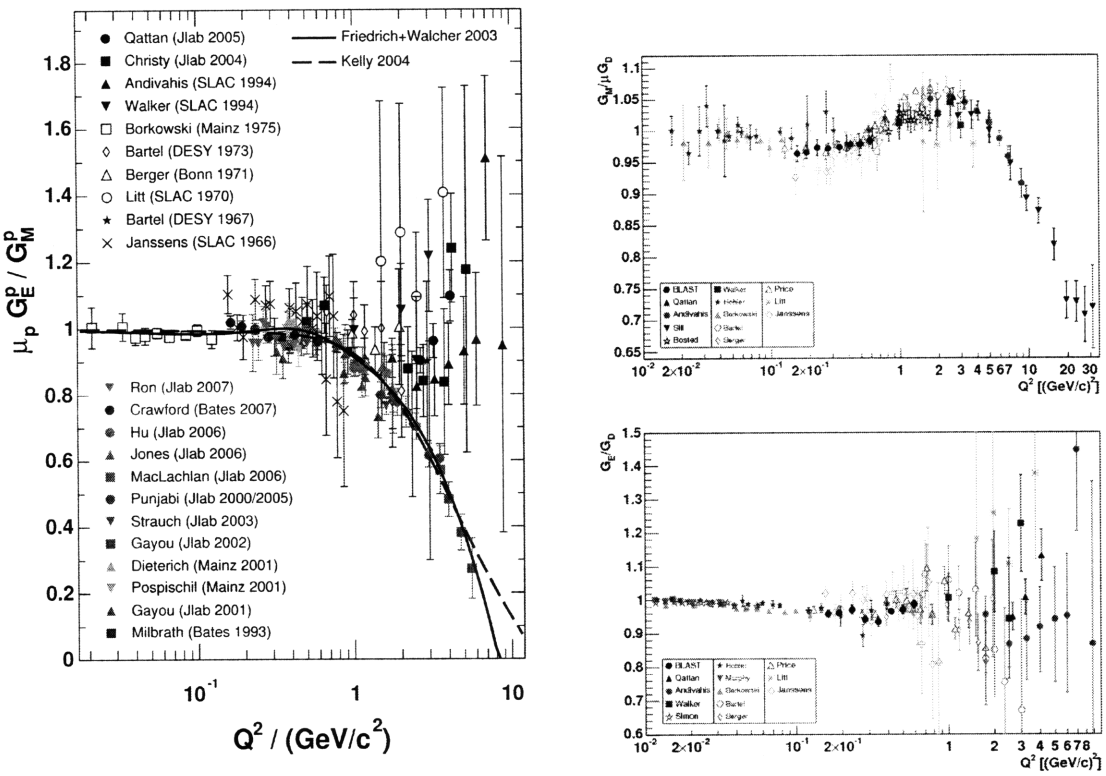


Figure 1-1: Experimentally Determined Form Factor Ratios
The Rosenbluth measurements are in black [14, 11, 26, 25, 5, 7, 8, 21, 12, 18]. The polarization measurements are in color [24, 4, 17, 16, 13, 23, 9, 10, 3, 20, 19, 22, 15, 6]. Two parameterizations are included [28, 31]. It was compiled in [34]. The image on the right contains the individual measurements of the form factors. The color separation does not exist in this image. It was compiled in [2]. The compilation procedure incorporated calibrating for a large number of different techniques used in the experimental measurements and is beyond the scope of this thesis.

[35]

The Q^2 suppression of the electric form factor in the scattering cross section means that measurements of electric form factor has increasing uncertainty with Q^2 when using the Rosenbluth method. The polarization technique does not share this limitation. This was part of the motivation for measuring the form factors with this method.

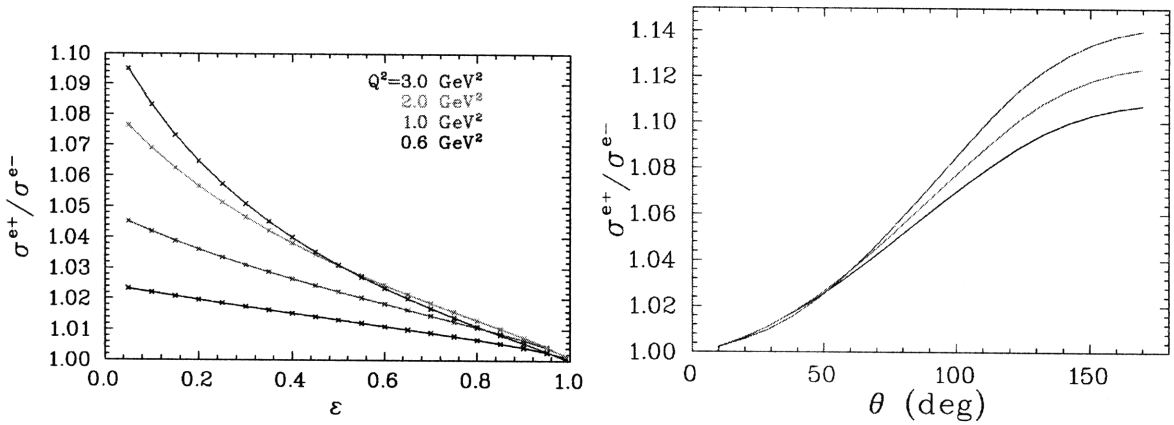


Figure 1-2: Cross Section Ratio ϵ and θ Dependence
From [34] which was calculated from[35]. On the right panel three beam energies are present (black=2.0 GeV, red=2.5 GeV, blue=3.0 GeV).

1.1.2 Experimental Discrepancies When Using the One Photon Approximation

Experimental measurements of the form factors as a function of Q^2 with the two different methods have yielded dramatically different results at high Q^2 values. A plot of these values collected from a various experiments can be seen in Figure 1-1 The polarized measurements are monotonically falling with Q^2[16, 13, 23, 9]. While the Rosenbluth method's measurements remain rather constant, though with increasing uncertainties.

The currently accepted explanation for this discrepancy is the exchange of multiple photons during lepton-proton scattering. Even though the multiple photon exchange has a small effect on the cross section, the $\approx 1/\tau$ factor affecting $G_E(Q^2)$ makes its effect on the cross section negligible. This Q^2 suppression of $G_E(Q^2)$ makes the dramatic effect of multiple photon exchange on this observable possible with current experimental results.

The real part of the multiple photon exchange affects the cross section. The observable most sensitive to effects of two photon exchanges is the ratio of the elastic scattering cross section of unpolarized electrons with unpolarized protons to that of the cross section of elastically scattered positrons. While under one photon exchanges the cross section is the same for both, when one accounts for multiple

photon exchanges there is an interference term between the one and two photon amplitudes that changes sign under charge inversion[35].

The effect is most noticeable at low ϵ values and high Q^2 values, see Figure 1-2. Unfortunately, increasing both of these is associated with lower count rates with larger statistical uncertainties. A beam energy of 2 GeV is a reasonable compromise between statistics and sensitivity and it also reduces the cost of running the experiment. Previous measurements of ϵ at low values had uncertainties of $\approx 5\%$ which is far too large to conclusively support deviations from unity[34].

1.2 Detectors

The OLYMPUS detector coordinate system is defined with the origin at the center of the hydrogen target. The z axis is down the beam line, the y axis is pointing up and the x axis is in the appropriate direction to form a right handed cartesian coordinate system. We define left and right in the detector. It follows that right is defined as the portion where x is negative. Left is for positive x.

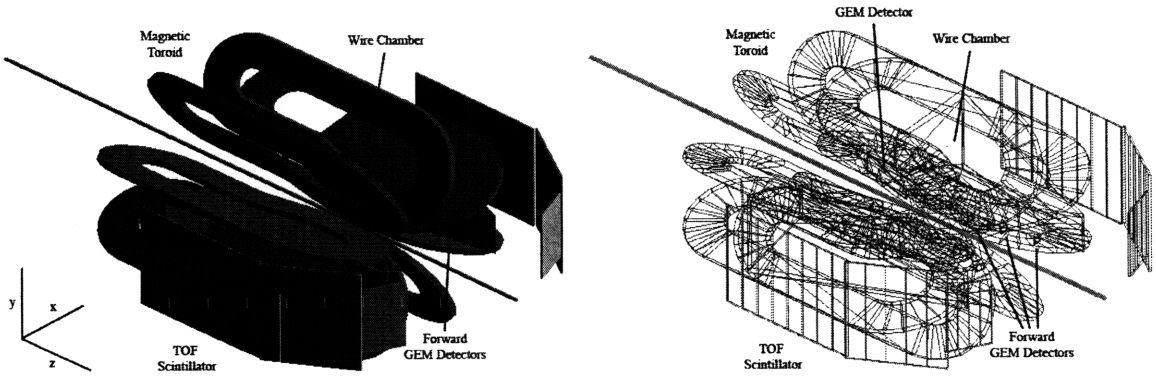


Figure 1-3: Schematic Layout of the Olympus Detector

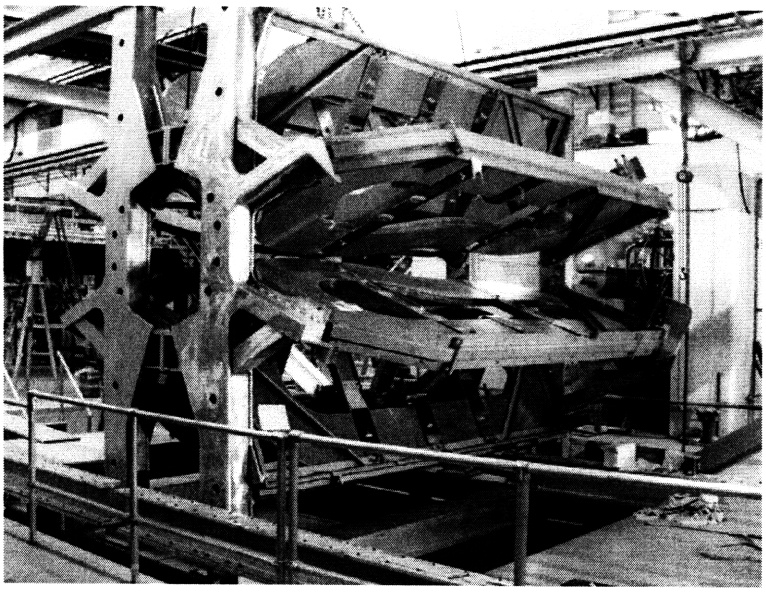


Figure 1-4: Toroidal Magnet Configuration

1.2.1 Toroidal Magnet

The toroidal magnet used in the detector consists of eight copper coils placed symmetrically around the beamline. Each coil consists of 26 turns of hollow, 1.5 inch square copper divided in two, even layers. The layers can be seen in Figure 1-4. The tubes were wrapped in fiberglass tape and potted with epoxy. During operation, they are cooled by water pumped through the hollow conductors. The toroidal configuration was chosen to maximize the magnetic field in the wire chambers to aid in momentum analysis while minimizing it around the beamline. It also helps deflect low energy particles away from the detector[34].

The monte carlo simulation and the reconstruction were dependent on knowing the value of the magnetic field generated by the coil in the space traversed by scattered particles. Inheriting the detector from the BLAST experiment improves the accuracy of the current simulation[34]. The magnetic field for the BLAST detector was measured in the horizontal sectors throughout the volume which would be occupied by the tracking detector via an EPICS controlled x-y-z table and two Sentron three-dimensional Hall probes. The position of the probes was known to 1mm and the precision of the measurement was accurate to 1%. Conveniently enough,

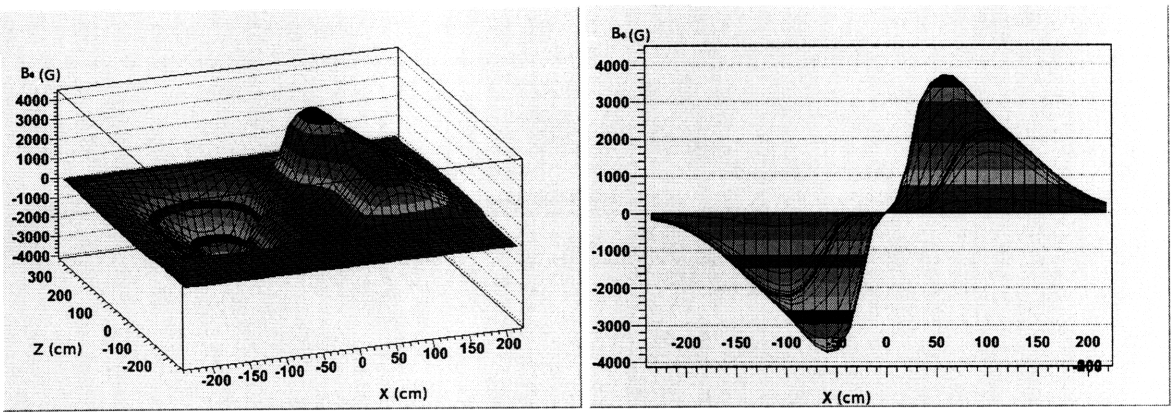


Figure 1-5: OLYMPUS Magnetic Field from Toroidal Magnetic Along Z-X Axis

this map agreed to 1% with TOSCA and Biot-Savart calculations. The simulations did not account for other magnetic materials in the detector which could lead to systematic errors[2]. The Biot-Savart calculation of the magnetic field was made more accurate by allowing the toroid to move radially, along the Z direction and in azimuthal position to provide the best match to the measured values. This fit position was then used to extend the values of the field past the measured portions to aid in reconstruction. We are currently using those values to simulate the magnetic field. For the proposed OLYMPUS experiment, the field in the tracking volume will be measured, again, at a number of points before the drift chambers are installed and the coil positions measured to aid in tracking and momentum analysis[34].

1.2.2 Time of Flight Scintillator

The Time of Flight Scintillators (ṪOFs) consist of a scintillator attached to photomultiplier tubes. Besides providing triggering information, the observed time differences for the left and right TOF scintillators is useful in particle identification by measuring the difference in the particle speeds or $\beta = v/c$. Organic scintillators produce photons when free valence electrons associated with π molecular orbitals from benzene-benzene bonding are excited by incident particles. These photons are collected by a photomultiplier tube (PMT) that amplifies the photons and then

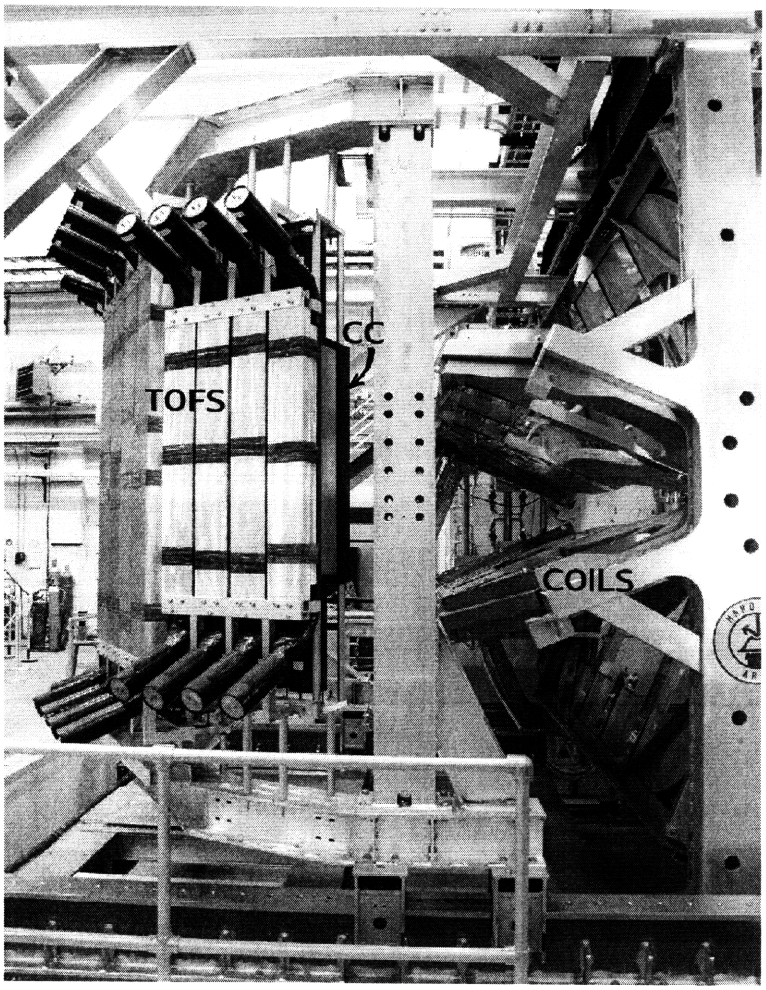


Figure 1-6: Time of Flight Scintillator
From [30]. The Cerenkov Counters in the image will not be included in the OLYMPUS detector.

transduces the signal.

Sixteen TOFs cover the scattering angles from 20° - 80° which matches the drift chamber's angular acceptance. The forward angle TOFs at $\theta < 40°$ are 119.38 cm in length, 15.24 cm wide, and 2.54 cm thick, while the backward angle TOFs at $\theta > 40°$ are 180.00 cm long, 26.2 cm wide, and 2.54 cm thick. They are constructed from Bicron BC-408 plastic scintillator. The PMTs are placed perpendicular to the magnetic field to aid in shielding[30].

They operate with greater than 99% efficiency. They were designed to have a timing resolution, FWHM, of 500 ps. Resolution measurements placed the resolu-

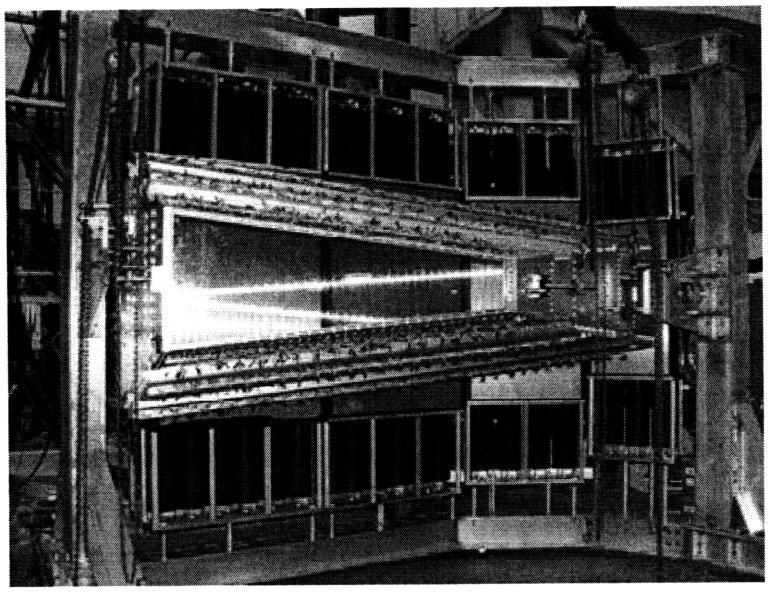


Figure 1-7: Photo of Assembled BLAST Wire Chambers
[34]

tion well within these tolerances[30].

1.2.3 Drift Chambers

The drift chambers are used to resolve the starting vertex position, momentum, and scattering angles (polar and azimuthal). The drift chambers have eighteen planes of sense wires, but effectively establish three points along the trajectory of a particle. Since the particle is traveling through a magnetic field generated by the toroid, a measurement of the momentum can be made. Tracing these points back to the beamline yields the other vertex measurements.

Wire chambers function by detecting the electron-ion pairs created by a charged particle passing through a gas. The ionized electrons are attracted to a thin anode wire with a strong electric field. The acceleration of the electrons near the wire produces a cascade of ionizations near the anode wire amplifying the electron signal greatly, making it detectable. The electrons are then collected by the wire. The pulse is measured with peak detection circuitry. The time it takes the electrons to drift a distance is known. By measuring the timing of pulses on several wires offset

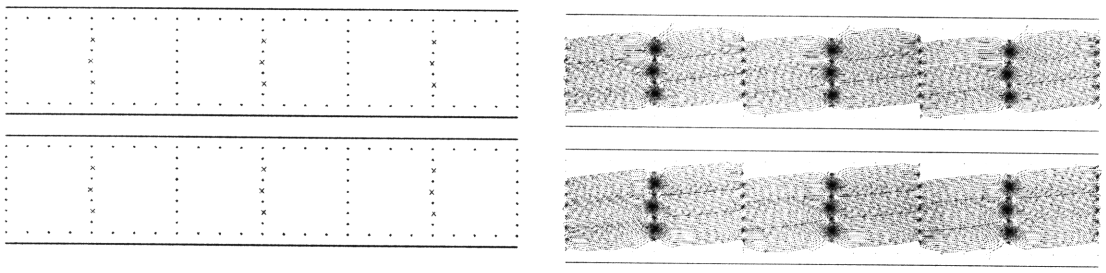


Figure 1-8: Drift Chamber Cell Wire Arrangement and Electric Field Lines [34] Electrons more or less follow the field lines to the anode.

from each other, high resolution (for large wire chambers the best resolution is on the order of $100 - 200$ μm) position measurements perpendicular to the wire and crude measurements along two other axes are generated. To compensate for this, two planes of wires set at different angles are used to provide accurate position measurements along two axes[1, 29].

The wire chambers for the OLYMPUS detector are recycled from the BLAST detector. The drift chambers fit between the toroidal coil to maximize angular acceptance, which is roughly $\theta \in (20°, 80°)$ and $\phi \in (-15°, 15°)$. The chambers are at a planar angle of $16.5°$ with respect to the beam line. The constraints of the toroidal magnet along with the desire to keep the angular acceptance of wire chambers constant with increasing distance from the beamline defined the trapezoidal shape of the wire chambers. There are two separate detectors along the left and right sides[34].

Each detector consists of three drift chambers (inner, middle, outer). Each chamber consists of two rows (super-layers) of drift cells in the 'jet' style. Each cell consists of three sense wires staggered by ± 0.5mm to resolve the left right ambiguity of position relative to drift time. This can be seen in Figure 1-8. The two superlayers are oriented such that one superlayer is offset by $+5°$ off vertical and the other is offset by $-5°$. The offset allows higher resolution position determination along both axes. The angle of the offset is small because resolution along the vertical axis is mainly useful in determining the ϕ component of the event while the other axis is useful in determining the θ, p, and z. While the two superlayers

should provide extra information about the slope, the resolution is so low it is not very useful in event reconstruction [2]. This wire chamber configuration produces a nominal resolution of 200μm in the plane of the wire chambers[2]. Even if at one point the wire chambers operated at this nominal performance, their effectiveness degrades through use. The avalanche of ionizations that occur in electron detection leads to the generation of free radicals which then polymerize and attach to the sensing wires. This reduces the conduction and increases the size of the wires decreasing the electron gain. While some means are available for restoring the wires, entropy always wins[29].

Each of the three drift chambers are joined together to form one gas chamber to reduce scattering off of potential faces for each chambers. The entrance and exits to the wire chambers are two layers of 25 μm mylar. The gas used in the chamber is 82.3% helium and 17.7% isobutane maintained at a pressure of 1 inch of water above atmospheric. As with the other design decisions, the low pressure and helium gas mixture are meant to reduce multiple scattering[34].

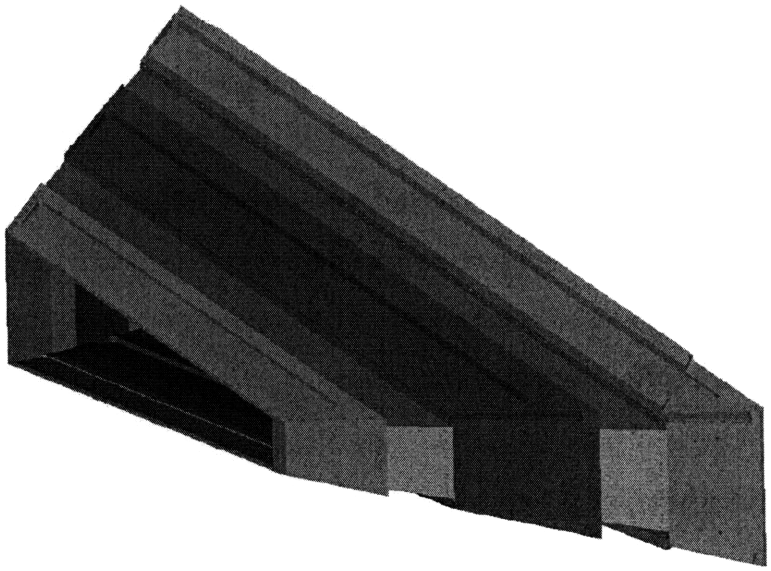


Figure 1-9: Assembled Wire Chamber

The form of the combined drift chambers can be seen in this rendering.[34]

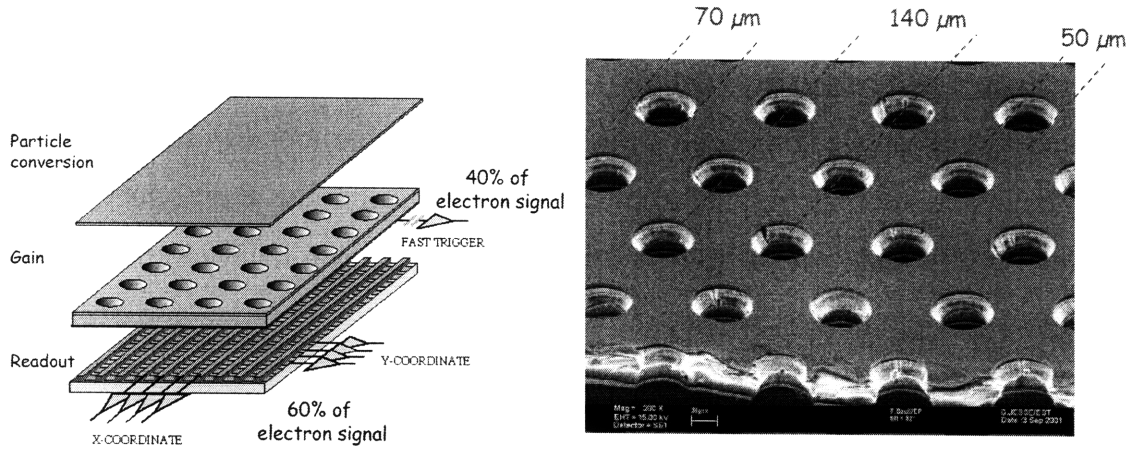


Figure 1-10: GEM Detector Construction

The left image shows the sandwich of (in descending order) cathode, the gain portion, and the detecting lattice of anodes. The right image is a picture of the amplification layer of a GEM detector[33].

1.2.4 Gas Electron Multiplier (GEM) Detector

GEM detectors operate on a similar principal to the wire chambers. Ions and electrons are formed in a gas and then attracted to detecting anodes and cathodes. The amplification method of the electrons, while similar in principal, is executed in a very different way. A very thin ($\approx 50\mu$m) insulator is coated on both sides with copper. Small holes ($\approx 50 - 70\mu$m) are etched in the copper, and then in the insulator. A potential difference is applied to the plates generating a very strong electric field in the hole. Electrons generated by the passage of charged particles follow the field lines through the holes. The strength of the field in the holes causes a cascade of ionizations, amplifying the electron signal similar to the cascade of ionizations near the anode wire in a wire chamber.

This amplification layer is sandwiched on one side by a charged cathode and on the other side by a patterned anode. The patterned anode consists of two layers of metal strips. The top layer consists of parallel rows of metal strips sitting on top of insulating kapton ridges. At the bottom of the insulating ridges is another set of metal strips set at some angle to the top row. These layers act as the readout for the detector. Multiple amplification layers (typically three) can be used to further

increase the signal coming from the ionization event[33, 36, 32].

The fabrication of the GEM foils is via photolithography reducing the distance between sensitive elements. This dramatically increases the spatial resolution relative to wire chambers which were limited by the time resolution and diffusion in the gas over the large drift distances. For wire chambers, these drift distances were constrained by the increased mechanical stress on the charged wires when moving them close together[29]. Multiple scattering is minimized by how thin the detector is. The small size also reduces the ion drift drift time leading to high count rates from GEM detectors[36].

GEM detectors are a proposed addition to BLAST detector. One proposed addition is the inclusion of three forward GEM detectors on each side of the beamlime to act as a luminosity monitor, which would greatly improve our understanding of the systematic uncertainties associated with varying count rates over time. This luminosity monitor would also be able to extract event information from high θ scattered protons. The other proposed addition is the inclusion of a GEM detector placed in front of the drift chambers. This detector would provide a high resolution point in space for use in reconstruction of the tracks through the detector. The inclusion of another point for use in reconstruction itself helps overconstrain the path of the particle through the magnetic field. This would provide a dramatic increase in the momentum and angular resolution of reconstruction[34].

1.2.5 Beam and Hydrogen Target

The detector would be relocated to the Deutsches Elektronen Synchrotron (DESY) facilities in Hamburg Germany. Three months using the DORIS ring are requested for the experiment. The beam would switch between 2 GeV electrons and positrons. To reduce systematic errors the switching time between e^- and e^+ would be minimized through possible modification to the DORIS ring[34].

The target is unpolarized hydrogen gas. The target consists of thin-walled (50μm) cylindrical, aluminum storage cell into which is fed hydrogen. The cross

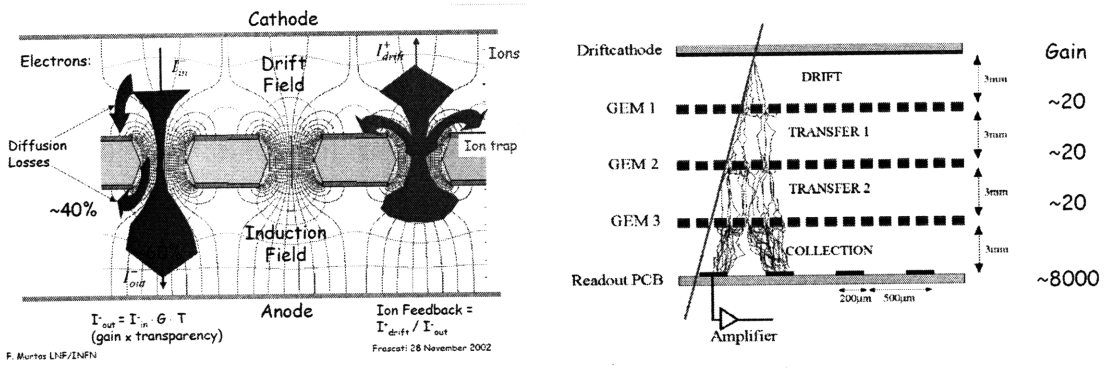


Figure 1-11: GEM Electron Multiplication

The left image shows the field lines for the amplification layer of a GEM detector. Note that $\approx 40\%$ of the electrons in the avalanche end up on the copper plate and not the anode layer. The right image shows the trail of electrons through a three stage GEM detector[33].

section of the storage cell to be 9 mm vertically by 15 mm horizontally. The gas diffusing out is pumped away by a system of vacuum pumps which are being reused from BLAST. The target is cooled to 25K giving a thickness of 3×10^{15} atoms/cm^2[34].

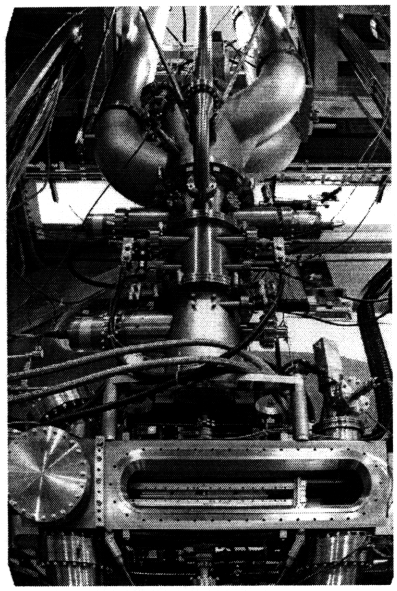


Figure 1-12: Hydrogen Target

Chapter 2

Software Methods

This section documents the software employed in the simulation and reconstruction. The first half describes the more complicated algorithms implemented, while the second portion provides a description of how the code is organized and the data flow.

2.1 Monte Carlo Methods

Event generation relies upon generating events with θ and z values that follows a certain probability density function, PDF. The inverse transform method was employed to do this[27]. Specifically, for a PDF, $p(x)$, and its cumulative density function, CDF, $P(x) = \int_{-\infty}^{x} p(x')dx'$, if we take a random variable ζ uniformly distributed on $(0, 1)$ then $P^{-1}(\zeta)$ has the same distribution as $p(x)$, where $P^{-1}(x)$ is the inverse of $P(x)$. When $P^{-1}(x)$ does not have an analytic form, numerical methods can be employed. This entails numerically integrating the PDF to generate the CDF. Finding $P^{-1}(\zeta)$ is equivalent to finding x such that $P(x) = \zeta$. Since the CDF is monotonically increasing, this can be efficiently implemented as a binary search.

2.2 Function Minimization

One cannot escape minimizing functions in experimental physics. There are a myriad of problems associated with minimization from local minima to edge cases. Fortunately, the problem of minimization has been tackled before. There are software libraries designed for this specific purpose. For problems like fitting circles to points in 3-space and fitting tracks, the Minuit minimization library[1] was used extensively. The Minuit library has a function that takes a function with n parameters that returns a real number and minimizes it. While it is computationally expensive relative to a possible custom algorithm for track fitting[39], the possible tracks for the low energy experiment are rather simple and the increased computational time is greatly offset by the decreased time in development. The use of Minuit reduces the problem of function minimization to the much simpler two fold problem of creating the function to be minimized and providing an accurate estimate for starting parameters. While the former is well defined, the second problem is, by its nature, filled with sloppy heuristics.

2.3 Fitting a Circle to a Collection of Points in 3-Space

Fitting a circle to the hits on the wire chamber provides a computationally inexpensive (relatively) way to estimate the starting parameters for fitting tracks to this collection of points.

2.3.1 Function to Be Minimized

We need some value that parameterizes how well a circle fits to a collection of points. For both fitting a circle and fitting a path to a collection of points we defined this as follows. For m points in our set S, let d_i be the Euclidean distance of the ith point from the curve then we define this value as $\sum_{i=1}^{m} f_i(d_i)$. The set of functions

[1] `http://lcgapp.cern.ch/project/cls/work-packages/mathlibs/minuit/home.html`

$f_i(x)$ are positive, monotonically increasing functions. Each point is allowed to have a different $f_i(d)$ because when fitting paths to hits on a detector there are variations in the detector resolution. In implementation, $f_i(d) = R * d^2$ where the constant R depends on the detector.

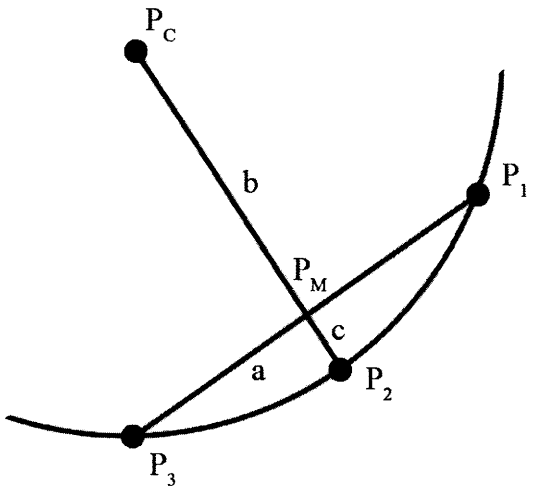
We need a way of finding the distance between a point and a circle in 3 space. This problem can be made easy with the correct representation of the point and the circle. Points are defined with the traditional Euclidean coordinates. We represent the circle with six parameters. Its radius, r, its center point p_c, and then two euler angles, θ and ψ. To generate our circle we apply a transformation, T, to every point on a circle of radius, r centered at the origin in the x-y plane. For a point, p, on the circle T is defined as:

$$T(p) = p_c + R_z(\psi)R_x(\theta)p \tag{2.1}$$

Where R_i is a rotation about the i^{th} axis. Note that the distance between our transformed circle and a point is the same as the distance between an untransformed circle and the inverse transformation applied to the point. After the inverse transformation is applied we just need to find the distance from $T^{-1}(p)$ and a circle of radius r centered at the origin on the x-y axis. The symmetries of this reduced problem make it fairly trivial.

2.3.2 Starting Parameters

For our particular task we are only interested in fitting circles to points located in the wire chambers. The small ϕ angular acceptance of the wire chambers coupled with the fact that the magnitude of the magnetic field is largest in the y direction means that a good estimate for the starting euler angles are $\theta = \pi/2$ and $\phi = 0$. Now we just need to find the center point and the radius. To do this we exploit a few facts about the circles we are trying to fit. First the center of the point is very far away from the points we are fitting the circle to. The points are also spaced fairly regularly. Using these approximations finding the center of the circle is a relatively straight forward task which is detailed in Figure 2-1.



With our approximation P_M is roughly in the center, so for $a = d(P_3, P_M)$ and $c = d(P_2, P_M)$, we have $b = a/\tan(2\tan^{-1}(a/c))$. From this we can find the center of the circle and the radius.

Figure 2-1: Estimation of Circle Starting Parameters

2.4 Fitting a Particle Track to a Collection of Points in 3-Space

For a particle of known mass and charge traveling from the beamline we are trying to find the starting conditions that best fit the path of the particle through a varying magnetic field to a collection of points. This means we are trying to find the starting z position, scattering angle off the beamline, $\theta = \tan^{-1}\left(\sqrt{p_x^2 + p_y^2}/p_z\right)$, its momentum p, and, to a lesser extent, $\phi = \tan^{-1}\left(p_y/p_x\right)$.

2.4.1 Function to Be Minimized

To give a numerical value estimating how well a path defined by a set of starting parameters fits the collection of points, we simulate the path of a particle through the magnetic field, find the distance of the path to the points and then assign a value as prescribed in section 2.3.1. Because this is reliant on numerical techniques, there are inherent inaccuracies in the fit. With proper choice of step size and integration technique, these errors can be minimized.

We generate a set of differential equations for the position and velocity using the standard $E\&M$ force relation with no electric field, $\frac{d\vec{P}}{dt} = q(\vec{v} \times \vec{B})$ and the relativistic relation between momentum and velocity. We have two main choices of integration technique, Runge-Kutta and Burlisch-Stoer[38].

Burlisch-Stoer was avoided because of how the magnetic field at a point is evaluated. To save computational time, the magnetic field was evaluated using numerical methods at a regular intervals on the detector and then stored in a table. The values of the surrounding points are then interpolated to find the magnetic field throughout all space. Burlish-Stoer does poorly on non-smooth problems such as when the right hand side of the differential equation is evaluated by table look-up and interpolation[38].

This leaves an n^{th} order Runge-Kutta method. We chose the classic workhorse of lazy and naive physicists, the fourth order Runge-Kutta method with a fixed step size. As the integrator is stepping through the path each point in the path is stored. This breaks the curve up into a series of line segments. The line segment closest to the point is found and the distance between the segment and the point is returned. This introduces another numerical inaccuracy since the path deviates slightly from the segment. Since the curves we care about all have a relatively low curvature, this method is accurate enough. The accuracy of the distance measurement could be improved by taking the starting conditions of the first point in the segment and integrating over the path with a smaller step size over that segment. In practice this was found to do very little to improve the accuracy of curves fit to the path because the majority of error in fitting procedure comes from inaccuracies in the integration and not the distance measurement.

An n^{th} order Runge-Kutta method while using an adaptive step size technique can provide excellent accuracy while keeping computational efficiency high, so it seems a bit odd that a fixed stepsize routine was used[38]. In order to improve accuracy, a variable step size routine (the step doubling technique[38]) was attempted along with an implementation that varied the step size based off of the starting velocity of the particle. Both attempts performed very poorly during re-

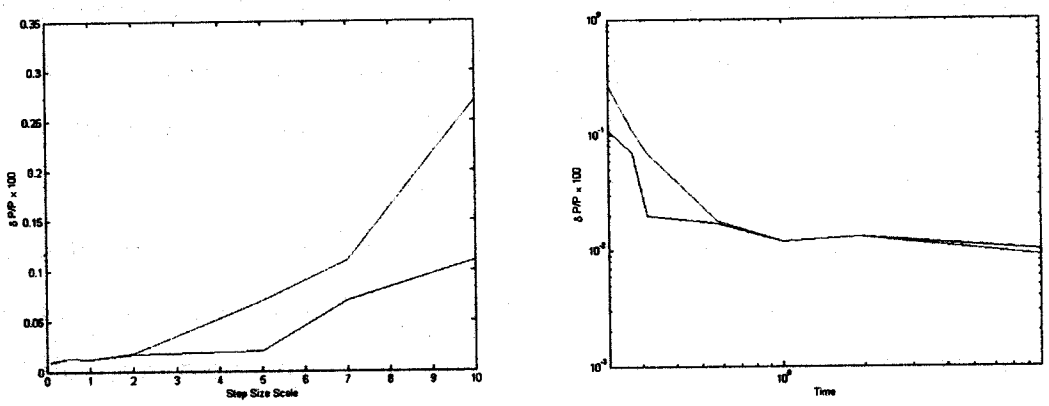


Figure 2-2: Run Time Analysis of Reconstruction with Varying Step Sizes
A quick run time analysis was done to determine an optimum time step size when integrating the paths of the particles. During integration testing, the electron time step size was set to 6.66×10^{-11}s and the proton step size was set to 3.11×10^{-11}. The left plot shows the momentum uncertainty as a function of step size. The step size the x-axis represents is the values listed above for testing multiplied by the scale factor. The right plot is a graph of the momentum uncertainty as a function of the normalized run time of the reconstruction program. For both plots, the green line is the proton uncertainty. The blue is the electron. Significant reductions in runtime can be gained by sacrificing a fraction of a percent of accuracy in the fit.

construction. The suspected reason is related to the minimization routine having trouble coping with irregularities. The distance from a hit to the path with a set of starting parameters is dependent on the location of the points used during integration. Part of this is because of the line approximation for measuring the distance, the other part is from inherent inaccuracies in numerical integration. While these variations are small, with irregular point locations, the errors vary irregularly from one path to another with very slight differences in starting parameters. These irregularities cause hell for the minimization procedure. Having a fixed step size means that these errors vary regularly with changing starting parameters. The minimization routine has no problem overcoming this.

Since we are tackling the same problem many, many times the ideal step size can be found beforehand. This helps offset the possible computational gains from using a variable step size routine. The fourth order Runge-Kutta method provides accuracies well within tolerances without a terrible runtime.

2.4.2 Starting Parameters

Initially a circle is fit to the points in our collection. A weighted average of the magnetic field over the points in the collection is taken with slightly more weight placed on the largest magnetic field value. Combining the radius of the circle and the magnetic field average generates a starting momentum value that's accurate to first order in the magnetic field. To find the initial z position, we exploit the fact that the curvature of our path is relatively small. We find the point on the fit circle closest to the z axis and then take its z value module a correction for the curvature as the initial z position. We then find the point in our set closest to the z axis and use the line between it and the initial z position to estimate the initial θ and ϕ values modulo a slight correction for curvature.

2.5 Module Organization

2.5.1 Monte Carlo Simulation

The Monte Carlo simulation was centered around the GEANT4 library. The use of the GEANT4 library basically determined the structure of the program. Understanding the program is highly dependent on understanding GEANT4. The program is dependent on the GEANT4 library, both of which are available from Cern. GEANT4 version 4.92[2], root version 5.22[3], and gcc version 4.2.4[4] was used.

The monte carlo simulation setup entails giving GEANT4 the detector information, physical processes to be used and the magnetic field in space. Events are then generated. This entails creating a list of particles with their species, momentum and position. These events follow the e-p scattering kinematics and cross section along with noise consisting of pion generation and Møller scattering. GEANT4 then handles simulating the path of these particles, and any other particles that might be generated during their flight through the detector. The physical processes

[2] http://geant4.web.cern.ch/geant4/
[3] http://root.cern.ch/
[4] http://gcc.gnu.org/gcc-4.2/

used by GEANT4 are selected by the user. GEANT4 reports hits on the detector which are then stored for use during reconstruction. How the events are stored depends on the detector.

Module Description

OLYMPUS_Generator handles the generation of a primary events. It inherits the G4VUserPrimaryGeneratorAction class.

OLYMPUS_Detector handles the construction of all the detectors in the GEANT4 simulation.

OLYMPUS_Physics contains the list of physical processes used during the simulation. It inherits G4VUserPhysicsList.

OLYMPUS_Mag_Field stores the magnetic field used during the simulation. It inherits G4MagneticField.

***_Hit & *_SD** handles hits and making a detector sensitive. Hits on the detector are stored in EventInfo.

OLYMPUS_UserEventAction Signals to the EventInfo class that the event has started and the event has ended. It inherits G4UserEventAction.

EventInfo stores hits on the detectors for use with reconstruction. EventInfo stores all of the hits in root's data class TTree.

Testing

Testing the overall functionality of the system is difficult, because if we knew how the system will behave in all cases, there would be no need for the simulation. This means that the simulations is very reliant on unit testing to insure proper functionality.

Operating under the philosophy that code that has not been tested does not work, all the parts written during development have been tested. GEANT4 handles all of the transport of particles and the physical processes. Fortunately, GEANT4's testing of $E\&M$ processes is well documented and repeatedly performed. [5] This means that ensuring the correctness of the Monte Carlo simulation is dependent on making sure the event generation, the hits on the detector, the physics list used and the detector layout have all been implemented properly. GEANT4 includes a standard physics list, QGSP,[6] which covers the energies we care about. For any simulation that was heavily dependent on the physics, this was used. The detector layout and basic hits on the detector can be verified via visual inspection with GEANT4's visualization library. Determining the materials the detectors are constructed from is reliant upon being careful while coding, a slight weakness in the testing. Grievous errors in the selection of the material could be detected via rapid attenuation of the particles during transport. For testing that the gaussian smearing of the hits on the detector was being done properly, a large number of numerical values of the smearing was compared to expected. The magnetic field value was ensured to not return unreasonable values by plotting the magnetic field as a function of space.

This leaves the event generation. The event generation code was unit tested by someone who did not write the code. Testing unit generation entails: making sure the kinematics are done properly, making sure θ, ϕ, z, and p generation are done properly, and checking for the proper selection of particles. At this point, only the elastic event generation has been tested thoroughly. Testing proper type selection of the particles was done via text output in GEANT4. The elastic kinematics were tested by comparing expected values with those generated by the program and previously computed values.

The angle of the scattered proton as a function of the scattering angle of the

[5] http://geant4.web.cern.ch/geant4/results/results.shtml

[6] http://geant4.cern.ch/support/proc_mod_catalog/physics_lists/hadronic/QGSP.html

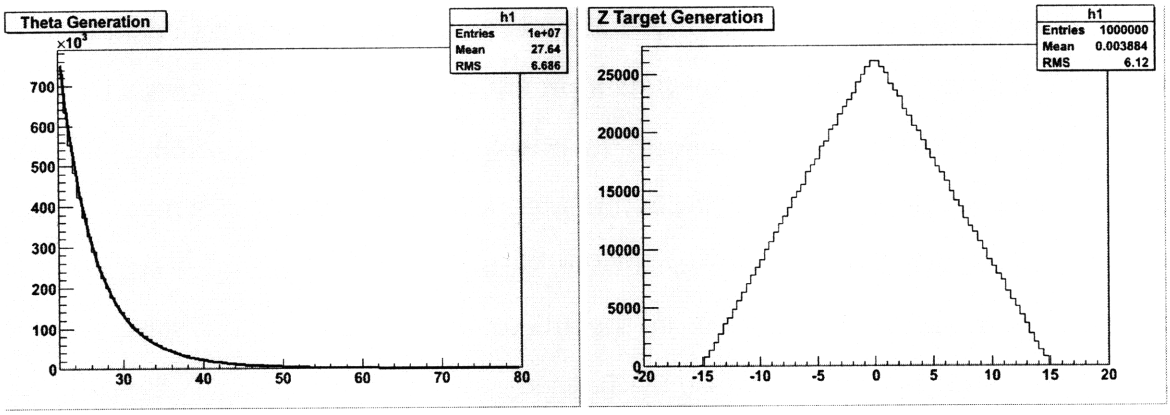


Figure 2-3: Elastic $e - p$ Event Generation z and θ Dependence
The line through the histogram represents the fit of the expected differential scattering cross section.

electron is:

$$\sin\theta_p = \sqrt{\left(\frac{1}{1+\tan^2\left(\frac{\theta_e}{2}\right)\left(\frac{E}{M_p}+1\right)}\right)} \tag{2.2}$$

[30]

The momentum of the scattered electron as a function of the scattering angle of the electron is:

$$k' = \frac{E}{\left(1+\frac{2E\sin^2(\theta_e/2)}{M_p}\right)} \tag{2.3}$$

[30]

Testing proper generation of the other kinematic variables involves generating a large number of events and then comparing the distribution of events to the expected distribution. For ϕ and z generation this was done via visual inspection. Visual inspection is fairly accurate for triangular and flat distributions. For the θ distribution, the expected PDF scaled by a constant was fit to the generated distribution. Since p is related to θ via the kinematic constraints that were already tested, fitting to the p distribution is not necessary.

2.5.2 Reconstruction

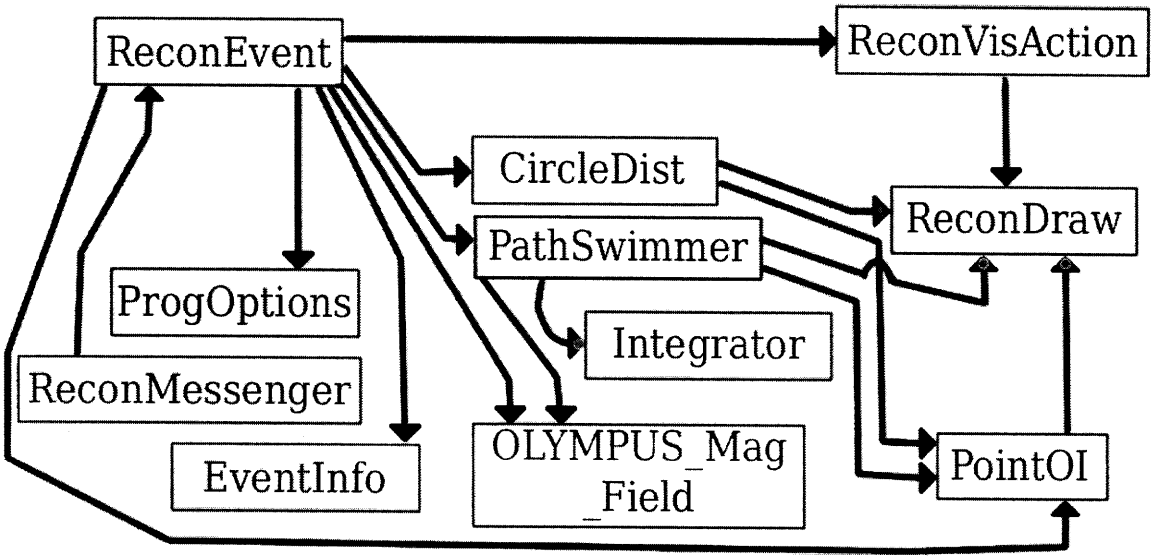
The reconstruction software converts hits on detectors into the most likely starting configuration for a scattering event. The software used in reconstruction are dependent on the root library specifically for storing information and performing minimizations. Visualizations of the reconstruction rely on the GEANT4 library. Visualizations are used primarily during debugging of the reconstruction software. CircDist and PathSwimmer coupled with the Minuit2 library are used to implement the minimization problems described above.

The reconstruction process works by first loading hits. Circles are fit to the hits generated by a possible track to estimate starting parameters for the tracks. Tracks are the fit to the hits on the detector. From these tracks the starting parameters of each events are extracted. During each step filtering is performed to remove any tracks that are not useful for reconstruction. Cuts are based off of kinematic constraints. A visualization of the process for an elastic e-p scattering event can be seen in figure 2-5.

Module Descriptions

ReconEvent is the main class during reconstruction. The class handles accessing the event information stored in EventInfo and storing the reconstructed event info. The fitting procedures calling the Minuit library and the estimations of the starting parameters are handled by this class. The filtering of events and tracks are also handled by this class. Basically every step in the reconstruction of events from hits on detectors are handled by this class.

EventInfo stores event information generated by the Monte Carlo simulation or the experimental results. The event information consists of the hits on the detectors for each event. In the case of events generated with the Monte Carlo simulation the generated event parameters are also stored. It also stores the reconstructed event info. The reconstructed info consists of the estimated starting parameters for each event.



In the module dependency diagram a solid arrow indicates dependence while an arrow with a red dot in the center indicates the class at the base of the arrow inherits the class at the point of the arrow.

Figure 2-4: Module Dependency Diagram for Reconstruction

ProgOptions provides post-compile time control for the program. It stores the command line options and options specified in a text file. It is implemented as a singleton.

ReconDraw is an abstract class meant to be inherited by any object during reconstruction that might be drawn. It exists to take advantage of polymorphism in C++.

ReconVisAction inherits G4VUserVisAction. It handles drawing objects during reconstruction using the GEANT4 visualization classes. It stores a list of objects that have inherited ReconDraw to be drawn.

Integrator stores a number of routines for doing numerical integration of ordinary differential equations. These methods include Runge-Kutta and Euler integration techniques.

OLYMPUS_Mag_Field loads the precalculated magnetic field values a set points

in space. It interpolates these values when returning the magnetic field at points in space.

PointOI represents a point of interest. It stores a position and a distance function. The distance function represents $f_i(d)$ from section 2.3.1. It determines the weight this point has during the fitting procedure.

PathSwimmer handles swimming a particle through the magnetic field with some set of starting conditions. It finds the distance from a point to this path for use with the fitting procedure called from ReconEvent. It also handles drawing the path.

CircleDist is our representation of a circle. Its main purpose is to find the distance from a circle to a point. It also handles the drawing procedures for a circle.

ReconMessenger implements the GEANT4 messenger class for use in controlling reconstruction while visualizing the reconstruction. Its mainly used in debugging.

General Notes

The Minuit2 library imbedded in root made the wonderful decision of declaring some of the arguments of the minimization library as *const*. Unfortunately with C++, the *const* keyword is rather viral. Any method needed to be called by a method declared *const* must be also be *const*. This has the unfortunate consequence that, for example, while trying to fit a path to a collection of points, the need to store path information during the swim means that the *mutable* keyword was abused heavily. Basically, when reading this code the *const* keyword is more a semantic formality that should not be taken to mean anything.

Testing

Unit testing for many of the classes was done with the CxxTest [7] unit library. Unit testing was done on all of the smaller modules downstream of ReconEvent. All of the functionality of PointOI, CircDist, Integrator, ProgOptions, ReconMessenger, ReconVisAction were tested via unit testing. Parts of the functionality of the PathSwimmer was tested via unit testing. Testing that the path matches the path of a particle was done while testing the overall functionality of the system.

The Monte Carlo simulation provided the means of testing the reconstruction. By comparing the simulated event information to the reconstructed event information we know if the reconstruction is happening properly. Particularly, if we turn off physics and smearing of hits, we can test that the reconstruction of the events is happening perfectly, where perfectly is defined as within tolerances. The use of floating point numbers, numerical integration techniques, and fitting procedures with finite accuracy means that even for the ideal case the event will not be reconstructed perfectly. Visualizations of the reconstruction proved invaluable during the debugging process.

2.6 Things To Be Done

The software is a work in progress. Another undergraduate will be working to improve the reconstruction and monte carlo simulation. This is why there is such an emphasis on documenting the software and the testing in this paper. The software work needed for OLYMPUS is not near over. Here is a list of needed improvements.

Filtering of tracks does not currently incorporate information from the TOFs. The TOFs are very useful for filtering based on estimated particle velocity. The reconstruction does not incorporate hits on the forward luminosity monitors for use in beam intensity estimates and reconstruction of events for high angle scattered protons.

[7] `http://cxxtest.tigris.org/`

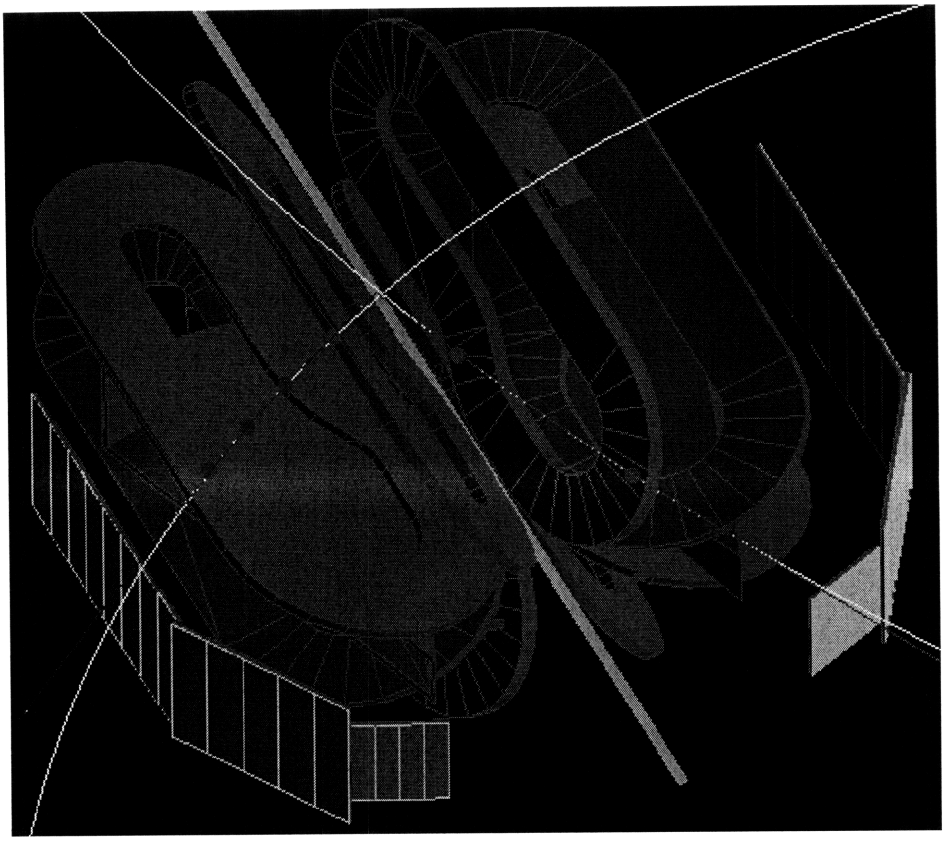


Figure 2-5: Reconstruction Elastic $e - p$ Scattering Event

The path of the proton is blue. The path of the electron is red. The hits on the drift chambers are purple, on the GEMS they are pink. The circles fit to the hits on the wire chambers to estimate the starting parameters for the path fit are yellow. The paths fit to the hits on the left and right detectors are teal/light blue. They are difficult to see in this image.

The model of the wire chambers is in need of improvement to provide a better simulation of the experiment and reconstruction. Using the actual wire chamber layout coupled with experimentally determined tests of the resolution of the wire chambers will provide a much better understanding of the resolution and hit rates. Modeling the structure of GEM detectors might also provide this insight. The small scale of the GEM detector means that modeling the structure would be about as useful as understanding the form of the position errors (possible deviations from a gaussian form) and the width of these errors.

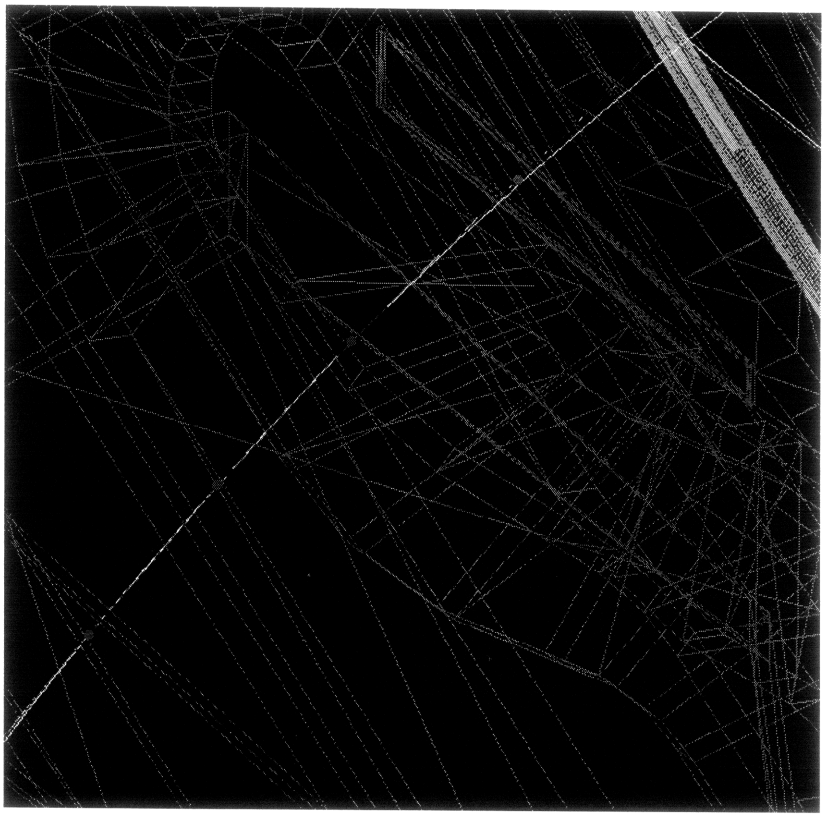


Figure 2-6: Close Up of Reconstruction Elastic $e - p$ Scattering Event

Chapter 3

Simulation Results

A high precision measurement of the electric and magnetic form factors depends on high resolution measurements of the momentum and scattered angle, specifically θ. It also depends on understanding the systematic errors associated with the count rates as a function of θ or p. Radiative corrections modify the Rosenbluth cross section. The count rates per bin (either p or θ) are scaled by the luminosity of the beam, the efficiency of the detector and the angular acceptance of the detector as a function of θ[14]. Future studies with the simulation software, will measure the performance of the proposed luminosity monitor. The momentum resolution and angular acceptances were estimated with the current simulation.

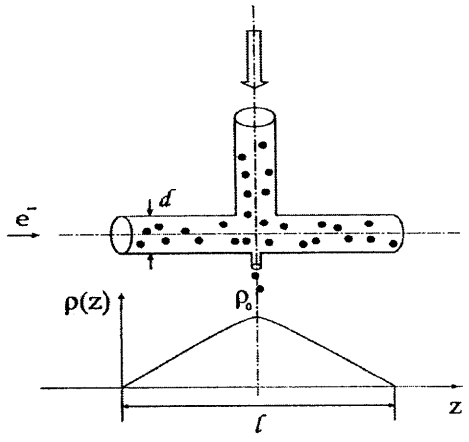


Figure 3-1: Hydrogen Target
Borrowed from[34].

Recon. Variable	Design Value	Measured Value
k'	2%	3%
$\Delta\theta_e$	.30°	.45°
$\Delta\phi_e$	.50°	.56°
Δz_e	$1.0cm$	$1.0cm$

Table 3.1: BLAST Drift Chamber Reconstruction Resolution [30]

3.1 Target and Detector Simulation

3.1.1 Hydrogen Target Simulation

The density of hydrogen in the beam path is roughly triangular around the feed system. Since the number of events is proportional to the density of the target, the scattering events have a triangular distribution along the z axis. See figure 3-1. Because of the narrow beam width, the x and y coordinates of the events are fixed at 0.

The electron θ for scattering events is generated by numerically integrating the Rosenbluth cross section and then performing a binary search to generate the inverse of the function. The ϕ distribution for the electrons generated is flat. The rest of the starting parameters are generated through kinematic constraints. Pion generation and $e - e$ scattering are included as noise.

3.1.2 Wire Chambers and GEM Detectors

Many different levels of complexity can be used in modeling the wire chambers. This can run from modeling the wires in the chamber and using their spatial resolution based on the ion drift time and then combining the hits on various wires to recreate the hit, to just storing where the particle hit the detector. We chose a middle path. Hits of charged particles registered on a detector were then smeared with a gaussian of a set σ. For the wire chambers, the the gaussian smearing along the vertical axis, y, and that normal to it in the plane of the detector, x, had different values, σ_y and σ_x. Unfortunately, the momentum and angular resolution of the

σ_x (μm)	σ_y (μm)	$\sigma_{\theta_{e^-}}$ (deg)	$\sigma_{p_{e^-}}$ ($\sigma_p/100p$)
100	500	$.21 \pm .01$	$2.3 \pm .1$
200	400	$.27 \pm .01$	$2.5 \pm .1$
300	700	$.23 \pm .01$	$3.1 \pm .1$
400	600	$.26 \pm .01$	$3.2 \pm .1$
500	1000	$.30 \pm .01$	$3.8 \pm .1$
200	1000	$.27 \pm .01$	$3.1 \pm .1$
300	1500	$.31 \pm .01$	$4.0 \pm .1$

Table 3.2: Monte Carlo Wire Chambers Resolutions for 850 MeV Beam

wire chambers is highly dependent on the spatial resolution of the hits.

Hits on the GEM detectors in the plane of the wire chambers also underwent a gaussian smearing. The spatial resolution for purposes of simulation was assumed to be 100 μm. The resolution of a GEM detector can reach about $30 - 40$ μm. Since the purpose of the study was to determine the usefulness of a GEM detector, setting the resolution higher than expected makes the justification of their use more valid.

Since the wire chambers have been used before, we have a means of calibrating our resolutions against the experimental resolutions found. Our goal was to match the resolutions found experimentally in BLAST with the wire chambers by varying the σ_x and σ_y. The experimental values are listed in table 3.1. The simulated wire chamber resolutions are listed in table 3.2.

While the momentum resolution matched the experimental values with reasonable assumptions about the position uncertainty of the wire chambers, the θ and z resolution was significantly higher than the experimentally determined values. This is either a problem with the simulation, our wire chamber simulation is a bit naive, or a difference in how the resolutions were determined. Either way, it means we have two variables and one degree of freedom. The asymmetric, $\sigma_x = 200\mu$m & $\sigma_y = 1000\mu$m, uncertainties had the highest angular uncertainty which matches the experimental results the best without exceeding the ideal resolution[2]. Increasing the beam energy to 2.0GeV caused the momentum resolution to decrease to $5.4 \pm .1\%$ and the θ resolution to to $.24° \pm .01°$. The straighter path of the higher

energy electrons are responsible for this shift, along with some of the physical process occurring at higher energies. If GEM detectors are included the momentum resolution improves to $1.56 \pm .02\%$ and the θ resolution improves to $0.081^\circ \pm .02^\circ$.

3.2 Angular Acceptance

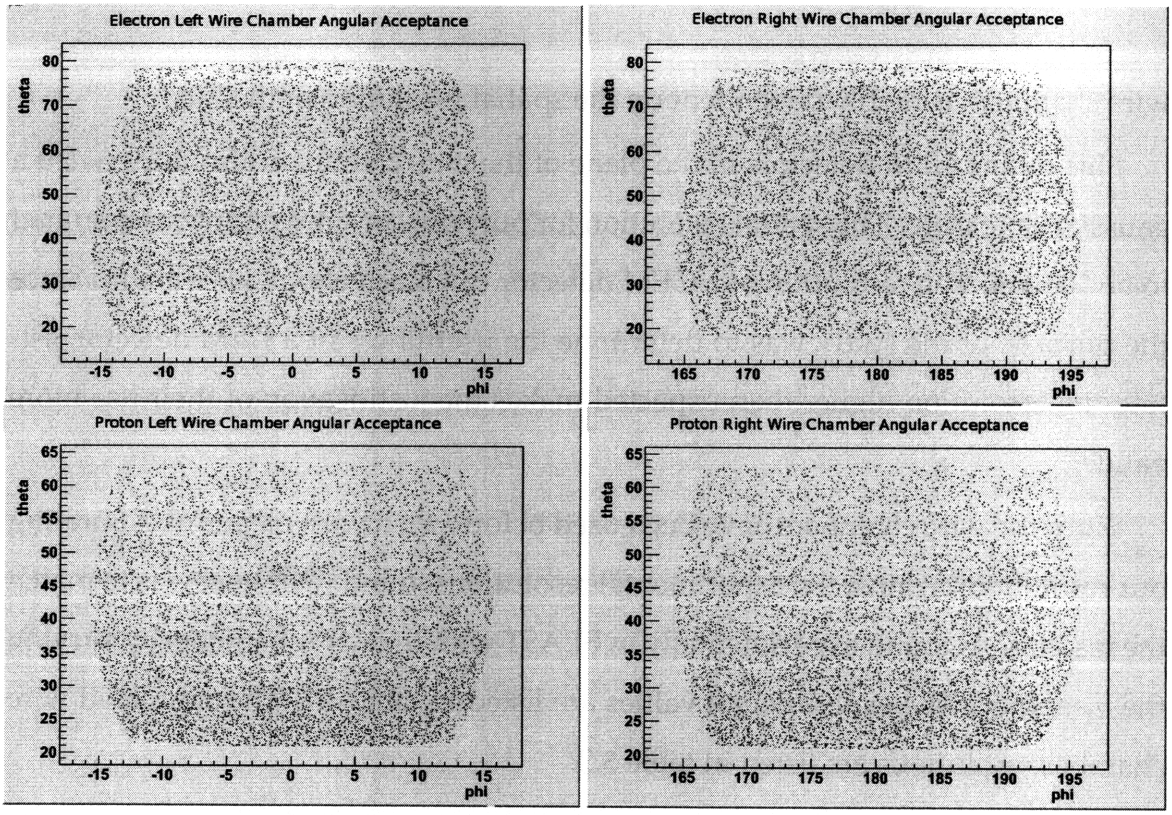


Figure 3-2: Angular Acceptances of e^- and p Where Reconstruction is Possible The top row features the electron's angular acceptance. These angular acceptance plots were generated with the physics turned off. That is to say, other than the particle traveling through the magnetic field, no other physical processes were simulated.

The angular acceptance of the detectors needs to be understood to account for systematic errors during data analysis. Normalizing the count rates as a function of theta is vital for getting the proper cross section out of the count rates. Also, during planning of the experiment, understanding the angular acceptance gives an understanding for the θ, and thus ϵ and Q^2 values we can probe.

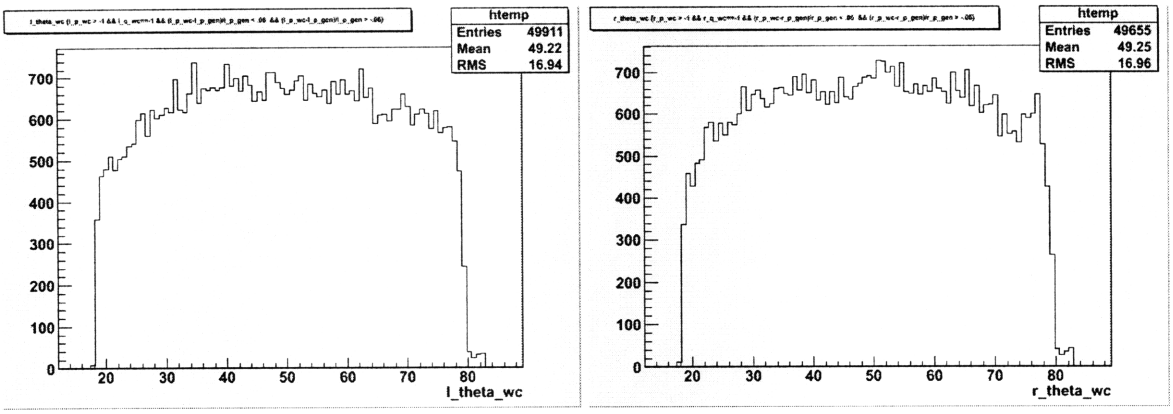


Figure 3-3: Angular Acceptances of e^- on Left and Right Detectors
These plots were generated using all the standard E&M processes accounting for the stray counts at θ values beyond the angular acceptance with no physics.

The plots of the angular acceptances were generated by using events with a flat θ_e distribution and the elastics scattering kinematics for $2.0GeV$ electrons. Only events that could be reconstructed with the wire chambers, that is to say where a scattered electron and proton each hit the wire chambers of the left and right sides.

3.3 Usefulness of Proposed GEM Detector

The forward GEM detectors are proposed as luminosity monitors to help reduce systematic uncertainties. This section, however, is focused on how beneficial a GEM detector in the plane of the wire chambers would be. For it to be worth the monitory investment it would need to have an angular acceptance comparable to the wire chambers while increasing the momentum and theta resolution of the OLYMPUS spectrometer.

For scattering events occurring at z values near the origin, the angular acceptance of the GEM detector is superior to that of the wire chambers. See figure 3-4. Because the GEM detector is so thin, the angular acceptance is highly affected by the z position of the event. Fortunately the angular acceptance only becomes inferior to the wire chambers angular acceptance in regions where almost few counts are expected.

Since GEM detectors are thin, it should not adversely affect the path of the

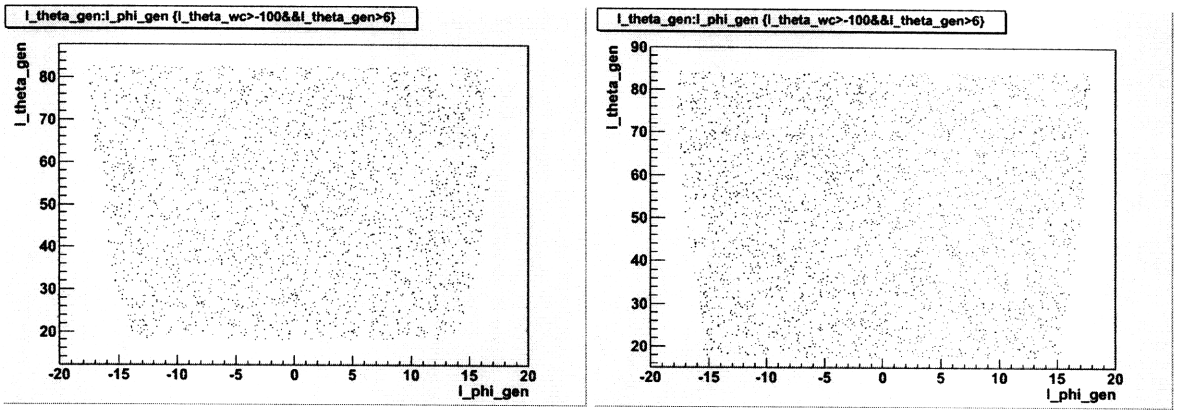


Figure 3-4: Angular Acceptance of e^- for WC and GEM Detectors

particle via scattering. This coupled with the extra high resolution information about the track means the GEM detectors will provide a dramatic increase in the momentum and theta resolution. For purposes of testing this, a large variety of wire chambers resolutions were chosen. In all cases the GEM detector provided at least a factor of two improvement in the momentum resolution. See table 3.3.

Momentum and theta resolution measurements were made by first subtracting the generated p and θ values from the ones garnered from reconstruction. A gaussian was then fit to the peak and the uncertainty was taken to be the σ of the fit.

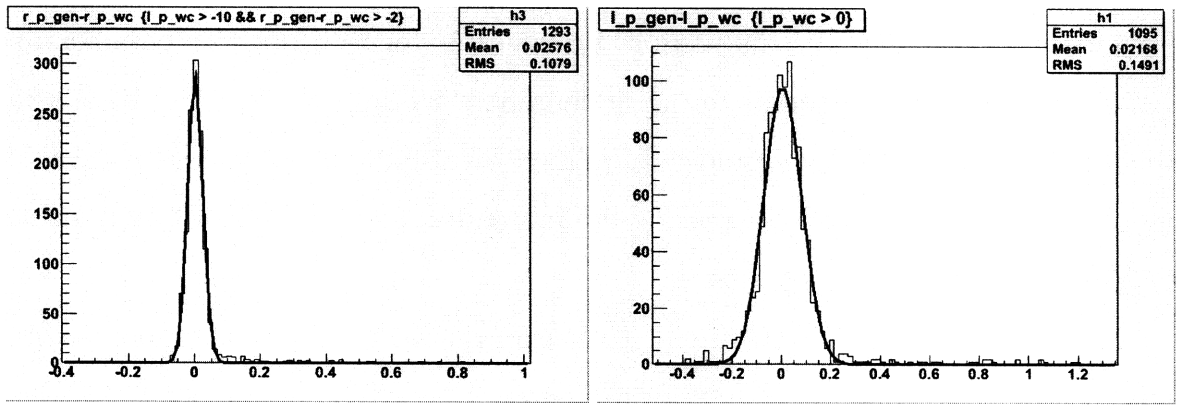


Figure 3-5: Momentum Resolution Fit

Particle	With GEM Det.	WC σ_x (μm)	WC σ_y(μm)	Mom. σ(MeV)
e^-	y	500	2000	44 ± 1
p	y	500	2000	37 ± 1
e^-	n	500	2000	163 ± 5
p	n	500	2000	126 ± 3
e^-	y	200	1000	$23.5 \pm .6$
p	y	200	1000	$23.4 \pm .4$
e^-	n	200	1000	77 ± 2
p	n	200	1000	62 ± 1
e^-	y	100	500	$19.6 \pm .5$
p	y	100	500	$20.5 \pm .4$
e^-	n	100	500	47 ± 1
p	n	100	500	41 ± 1

Table 3.3: Momentum Resolution Measurements
Momentum resolution for elastic $e^- - p$ scattered at $45°$. Note the significant improvement of momentum resolution with the inclusion of hits on the GEM detector.

Chapter 4

Conclusion

The intricacies of modern nuclear physics experiments necessitate simulating the experiment in order to compensate for systematic errors and to fully understand the operation of the detector. The need for these simulations drove the development of the GEANT library, which handles all the physical processes needed for a comprehensive model of many experiments. Using the GEANT4 library, we created a simulation to aid our understanding of the proposed OLYMPUS experiment. While the simulation of some of the detectors is rather simple at this point in time, we have laid the groundwork for iterative improvements to the simulation software. The strong emphasis placed on testing and documentation will aid in future development.

Extracting the e-p scattering cross section requires software that takes hits on detectors, filters for relevant scattering events, and then determines the most likely kinematic parameters for the scattered particles. We wrote code relying the root libraries to accomplish these goals. The simulation provides a means of testing this reconstruction code because with software, unlike real life, we know the scattering angles and momenta for the particles. While some of the functionality of the reconstruction software needs to be modified. It needs to include all the detectors in the experiment and provide better filtering of events. These are all relatively easy tweaks to the software already written. While there is a lot of work to be done with the simulation and reconstruction software, it is mostly down hill from here.

Bibliography

[1] A. Cattai and G. Rolandi. Wire chambers. *Journal of Physics G, Nuclear and Particle Physics, Review of Particle Physics*, 1999.

[2] Christopher Blair Crawford. *Precision Measurement of the Proton Electric to Magnetic Form Factor Ratio with BLAST*. PhD dissertation, MIT, Department of Physics, May 2005.

[3] B. Hu et al. *Phys. Rev. C*, 73(064004), 2006.

[4] B.D. Milbrath et al. A comparison of polarization observables in electron scattering from the proton and deuteron. *Phys. Rev. Lett.*, (80), 2005.

[5] C. Berger et al. *Phys. Lett.*, (87), 1971.

[6] C.B. Crawford et al. *Phys. Rev. Lett.*, 98(052301), 2007.

[7] F. Borkowski et al. *Nucl. Phys.*, (269), 1974.

[8] F. Borkowski et al. *Nucl. Phys.*, (461), 1975.

[9] G. MacLachlan et al. *Phys. Rev. Lett. C*, 73(261), 2006.

[10] G. Ron et al. *Phys. Rev. Lett.*, 99(202002), 2007.

[11] I.A. Qattan et. al. Precision rosenbluth measurement of the proton elastic form factors. *Phys. Rev. Lett*, 94(142301), 2005.

[12] J. Litt et al. *Phys. Lett.*, (40), 1970.

[13] M. Jones et al. *Phys. Rev. Lett.*, 84(1398), 2000.

[14] M.E. Christy et. al. Measurements of electron-proton elastic cross sections for $0.4 < q^2 < 5.5(gev/c)^2$. *Phys. Rev.*, C70(015206), 2004.

[15] M.K. Jones et al. *Phys. Rev. C*, 74(035201), 2006.

[16] O. Gayou et al. *Phys. Rev. Lett.*, 88(092301), 2002.

[17] O. Gayou et al. Measurement of g_{E_p}/g_{M_p} in $ep \rightarrow ep$ to $q^2 = 5.6gev^2$. *Phys. Rev. Lett*, 88(092301), 2002.

[18] R.C. Walker et al. *Phys. Rev. D*, 49(5671), 1994.

[19] S. Dieterich et al. *Phys. Lett.*, B500(47), 2001.

[20] S. Strauch et al. *Phys. Rev. Lett.*, 91(052301), 2003.

[21] T. Janssens et al. *Phys. Rev.*, 142(922), 1966.

[22] T. Pospischil et al. *Eur. Phys. J.*, A12(125), 2001.

[23] V. Punjabi et al. *Phys. Rev.*, C71(069902(E)), 2005.

[24] V. Punjabi C.F. Perdrisat et al. Proton elastic form factor ratios to $q^2 = 3.5$ gev^2 by polarization transfer. *Phys. Rev.*, C71(055202), 2005.

[25] W. Bartel et al. *Phys. Lett.*, (236), 1967.

[26] W. Bartel et al. *Nucl. Phys.*, (429), 1973.

[27] Eduardo Acosta Francesc Salvat, José M. Fernández-Varea and Josep Sempau, editors. *PENELOPE - A Code System for Monte Carlo Simulation of Electron and Photon Transport*, Issy-les-Moulineaux, France, November 2001.

[28] J. Friedrich and Th. Walcher. *Eur. Phys. J. A*, 17(607), 2003.

[29] Claus Grupen. *Particle Detectors*, volume 2 of *The Art of Computer Programming*. Cambridge University Press, New York, New York.

[30] Peter Joseph Karpius. *Vector Polarization Observables of the Deuteron and A New Measurement of the Magnetic Dipole Form Factor G_M*. PhD dissertation, University of New Hampshire, Physics, 2005.

[31] J.J. Kelly. *Phys. Rev. C*, 70(068202), 2004.

[32] P. Sievers M. Ziegler and U. Straumann. A triple gem detector with two-dimensional readout. *Nuclear Instruments and Methods in Physics Research Section A*, 471(1-2):260–263, 2000.

[33] F. Murtas. Development of a gaseous detector based on gas electron multiplier (gem) technology. Slides from a presentation given by a member of Instituto Nazionale di Fisica Nucleare, November 2002.

[34] The OLYMPUS Collaboration. *A PROPOSAL TO DEFINITIVELY DETERMINE THE CONTRIBUTION OF MULTIPLE PHOTON EXCHANGE IN ELASTIC LEPTON-NUCLEON SCATTERING*, 2008.

[35] J.A. Tjon P.G. Blunden, W. Melnitchouk. Two-photon exchange in elastic electron-nucleon scattering. *Phys. Rev. Lett.*, C72(034612), 2005.

[36] M.T Ronan. Micro-pattern gas detectors. *Journal of Physics G, Nuclear and Particle Physics, Review of Particle Physics*, 2005.

[37] M. N. Rosenbluth. High energy elastic scattering of electrons on protons. *Phys. Rev.*, 79(4):615–619, Aug 1950.

[38] William T. Vetterling & Brian P. Flannery William H. Press, Saul A. Teukolsky. *Seminumerical Algorithms*. Numerical Recipes in C: The Art of Scientific Computing. Cambridge University Press, New York, second edition.

[39] Fred James & Matthias Winkler. *MINUIT User's Guide*. Cern, Geneva, June 2004.