

University of Louisville

ThinkIR: The University of Louisville's Institutional Repository

Electronic Theses and Dissertations

5-2021

Study of neutral hadron production in a high intensity particle physics experiment.

Jake W. Berg
University of Louisville

Follow this and additional works at: <https://ir.library.louisville.edu/etd>



Part of the [Elementary Particles and Fields and String Theory Commons](#)

Recommended Citation

Berg, Jake W., "Study of neutral hadron production in a high intensity particle physics experiment." (2021). *Electronic Theses and Dissertations*. Paper 3632.
<https://doi.org/10.18297/etd/3632>

This Doctoral Dissertation is brought to you for free and open access by ThinkIR: The University of Louisville's Institutional Repository. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of ThinkIR: The University of Louisville's Institutional Repository. This title appears here courtesy of the author, who has retained all other copyrights. For more information, please contact thinkir@louisville.edu.

STUDY OF NEUTRAL HADRON PRODUCTION IN A HIGH INTENSITY
PARTICLE PHYSICS EXPERIMENT

By
Jacob W. Berg
B.S., University of Wisconsin Stevens Point, 2011
M.S., University of Louisville, 2013

A Dissertation
Submitted to the Faculty of the
College of Arts and Sciences of the University of Louisville
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy
in Physics

Department of Physics and Astronomy
University of Louisville
Louisville, Kentucky

May 2021

Copyright 2021 by Jacob W. Berg

All rights reserved

STUDY OF NEUTRAL HADRON PRODUCTION IN A HIGH INTENSITY
PARTICLE PHYSICS EXPERIMENT

By

Jacob W. Berg
B.S., University of Wisconsin Stevens Point, 2011
M.S., University of Louisville, 2013

Dissertation approved on

April 21, 2021

by the following dissertation Committee:

Dissertation Director
David N. Brown, Ph.D.

Swagato Banerjee, Ph.D.

Gamini Sumanasekera, Ph.D.

Thomas Riedel, Ph.D.

DEDICATION

This dissertation is dedicated first and foremost to my wife, Kayla, and my daughter, Emmeline. Kayla, you've stood by my side and supported me along this entire journey. We've built an incredible life together. I love you. Emme, I love being your daddy, thank you for keeping me sane by being the funniest, weirdest kid there is.

This dissertation is also dedicated to my grandfather, William H. Pichette, Ph.D., for being my inspiration to explore the universe and sparking my interest in new technology, setting forth my trajectory in pursuing my degrees.

ACKNOWLEDGMENTS

Thank you to my advisor, David N. Brown, Ph.D. for your patience and for having faith in me, especially in the last two years. Your advice on this study and in life have been invaluable.

Thank you to Chris Verzani, Ph.D. and Hai Nguyen, Ph.D. who were my professors at the University of Wisconsin Stevens Point. You introduced me to the worldwide Physics community and taught me the fundamentals of academic research.

I would also like to thank my wife for her assistance in editing this dissertation and for having the patience to read through it all.

Erik, Andrew, James, Jeff, Nate, Adam, and Brian: you guys don't know how much your friendship means to me, thank you for being an outlet in stressful times and for being there to talk about all the ups and downs in life. (R2R)

Finally, to my dad, Jim, and my mom, Mary Kay (and step-dad Randy), thank you for everything you've ever done for me and for your unquestioning support in my disciplines of choice. I've always appreciated your cooperation with each other that provided a (thankfully) boring upbringing.

ABSTRACT

STUDY OF NEUTRAL HADRON PRODUCTION IN A HIGH INTENSITY PARTICLE PHYSICS EXPERIMENT

Jacob W. Berg

April 21, 2021

In this dissertation I present independent measurements of the multiplicities, R , of π^0 and η mesons produced in annihilation interactions of electrons and positrons at a center-of-mass energy of 10.54 GeV. Data were collected using the *BABAR* detector at the PEP-II storage rings located at the SLAC National Accelerator Laboratory in Menlo Park, California. Both mesons have a decay mode to two photons, which is used to identify them and measure their production per event. I find $R(\pi^0) = 2.8 \pm 0.3$ per event and $R(\eta) = 0.25 \pm 0.03$ per event. I also present my contributions to software development for the Mu2e Experiment at the Fermi National Accelerator Laboratory.

TABLE OF CONTENTS

DEDICATION	iii
ACKNOWLEDGMENTS	iv
ABSTRACT	v
LIST OF TABLES	viii
LIST OF FIGURES	ix
INTRODUCTION	1
An Overview of Particle Physics	1
Examples of Studies in Particle Physics using Simulated Data and Simula- tion Software	5
THE <i>BABAR</i> EXPERIMENT	19
Background	19
The <i>BABAR</i> Detector	24
PRODUCTION OF INCLUSIVE NEUTRAL HADRONS	31
Introduction	31
Event Selection	33
π^0 ANALYSIS	37
Background	37
Cut Identification	37
π^0 Fitting	42
Uncertainty Analysis	46

η ANALYSIS	49
Background	49
η Cutting	50
η Fitting	51
Uncertainty Analysis	55
CONCLUSIONS	59
REFERENCES	61
APPENDIX A: COMMONLY USED ACRONYMS	64
APPENDIX B: MODEL CODE	65
Example of histogram drawing and fitting code	65
Example of <i>BABAR</i> run time cutting code	81
Example of π^0 veto code	82
CURRICULUM VITA	85

LIST OF TABLES

1	Cuts identified for π^0	40
2	Signal Increase per π^0 Cut	40
3	Detector Efficiency Contribution to Uncertainty	47
4	π^0 Uncertainty	48
5	η Uncertainty	58
6	π^0 Results	60
7	η Results	60

LIST OF FIGURES

1	The Standard Model	2
2	Feynman Diagrams for Possible Muon to Electron Conversion Processes .	9
3	The Mu2e Detector	9
4	Production Solenoid Area Plans	15
5	Production Solenoid Area Geometry	16
6	Remote Handling	17
7	Bottomonium Spectrum	21
8	Feynman diagram for $\Upsilon(4S) \rightarrow B^0\bar{B}^0$	22
9	<i>BABAR</i> Integrated Luminosity	23
10	SLAC Overview	25
11	<i>BABAR</i> Detector	26
12	Resonance peaks of Υ	32
13	Feynman diagram for $e^+e^- \rightarrow \gamma^* \rightarrow q\bar{q}$	34
14	Feynman diagram for $\pi^0 \rightarrow \gamma\gamma$	38
15	Comparison of the π^0 mass peak	41
16	π^0 Identification Detector Efficiency	43
17	π^0 Monte Carlo Histograms	44
18	π^0 Data Histograms	45
19	π^0 Multiplicity per Momentum Bin	46
20	Separated Novosibirsk and Gaussian functions	53
21	η fit example	53
22	η Identification Detector Efficiency	54
23	η Multiplicity per Momentum Bin	55

24	η Monte Carlo Histograms	56
25	η Data Histograms	57

CHAPTER I

INTRODUCTION

1 An Overview of Particle Physics

Throughout the history of physics, there has always been a concept of a “fundamental” particle. The fundamental particle was thought to be indivisible and the building block that made up all other matter as it was known. At one point, the atom was thought to be the fundamental particle - until it was found that each atom itself is made up of protons, neutrons, and electrons. Currently, it is understood that the universe is not built from a single fundamental particle, rather it is built from a conglomerate of twelve different types of “building block” particles, four other types of particles that bind them together, and the Higgs boson. This collection is contained in what is known as the Standard Model of Particle Physics, or just Standard Model (SM) for short. A representation of the current Standard Model can be seen in Figure 1, as taken from reference [1]. Each of these particles also has an anti-matter twin, which means they have the same mass and spin properties. The other quantum numbers, such as charge, lepton/baryon number, etc. will have the opposite sign. Like the atom before them, these particles are each believed to be structure-less.

The particles in the Standard Model consist of four force-mediators, six quarks, six leptons, and the Higgs boson. The manner in which each of these particles interacts with each other and other rules for the behavior of the particles are also described by the Standard Model. Basically, the entirety of the observations and understanding of matter at the fundamental particle level makes up the Standard Model. New theories and ideas are added to the Standard Model as they are proven or an observation is

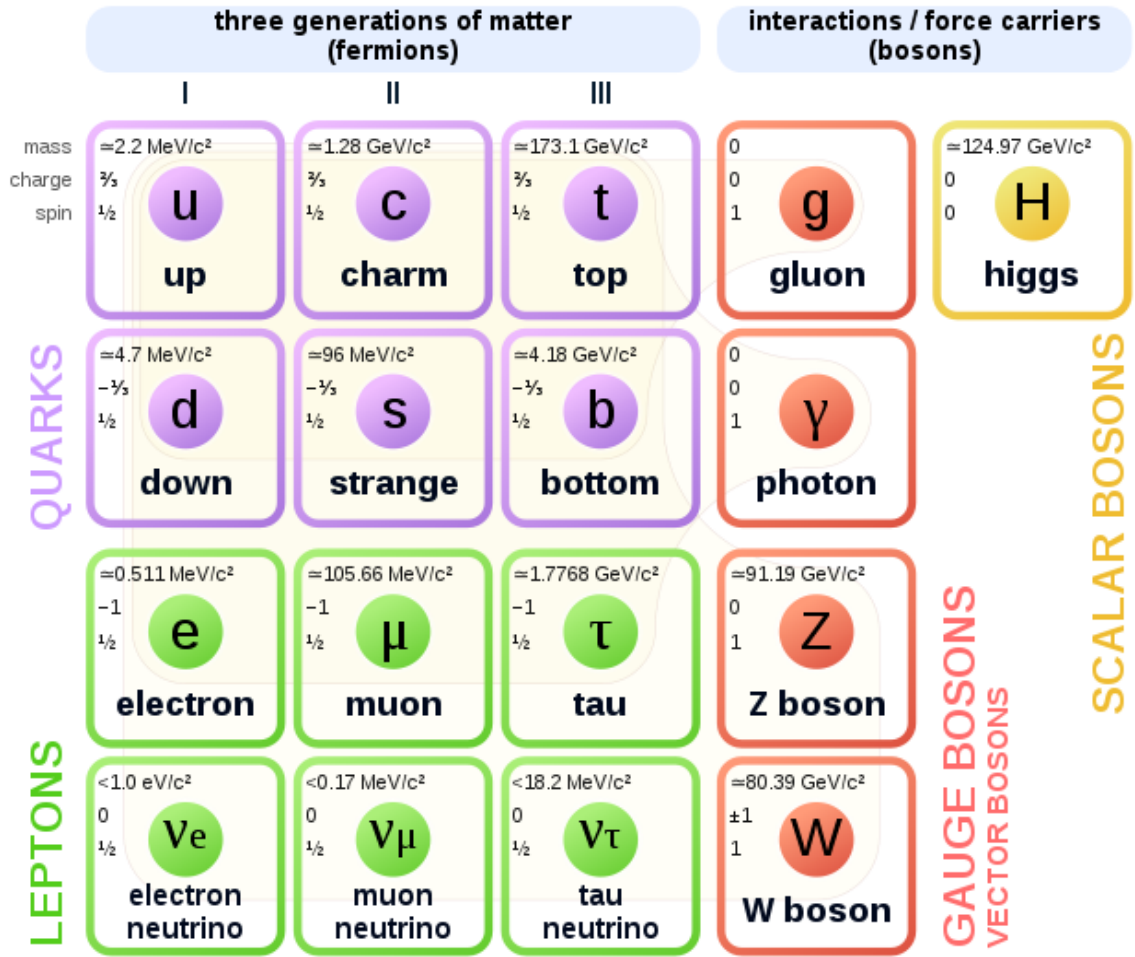


Figure 1. A representation of the currently accepted Standard Model. Image courtesy [1].

made to confirm them. A recent example is the Higgs boson – despite being a widely accepted idea long before its observation, it was not officially added as a part of the Standard Model until the confirmation of its existence in 2012. The force-mediating particles are created and exchanged between two other particles, or emitted and reabsorbed by a single particle during an interaction. The electromagnetic force is mediated by the photon (γ), the weak force is mediated by either the W^\pm or the Z^0 , and the strong force is mediated by the gluon (g).

The six quarks can be grouped many different ways to form new particles. The quarks are named, in order of lowest to highest mass, up (u), down (d), strange (s),

charm (c), bottom (b), and top (t). The next set of particles, the leptons, are split into two groups. The first group is the electron (e), the muon (μ), and the tau (τ). These particles have a charge of -1.602×10^{-19} Coulombs (usually expressed as $-e$; most fundamental particles' charges are usually expressed in fractions of e) and are very similar to one another except for their mass. Each of the three particles from the first group of leptons has a counterpart particle in the second group. Particles in the second group are known collectively as neutrinos. Neutrinos have zero charge and have very small mass compared to the other particles and come in three varieties, each of which is related to a charged lepton in the first group: the electron neutrino, the muon neutrino, and the tau neutrino. The final particle in the Standard Model is named the Higgs. This particle is responsible for the other particles in the Standard Model having mass via the excitation of the "Higgs Field," which permeates throughout the entire universe.

The quarks are very interesting particles with many different qualities that define them. As well as familiar properties such as charge and mass, quarks also have another type of characteristic called color charge. The Standard Model says that all observable particles must be "colorless," meaning they have no net color charge. Since quarks each have a color, they are not individually observable; they must combine with other quarks to form a completely colorless particle. They then are experimentally observable. Color charge has a total of six possible values. These are red (r), green (g), and blue (b), and the antimatter complement of each: anti-red (\bar{r}), anti-green (\bar{g}), and anti-blue (\bar{b}). Typically, they can form a group of three quarks, called a baryon, or a pair containing a quark and an anti-quark, called a meson. As an example, a proton is a baryon consisting of two up quarks and a down quark. For the particle to be observable, one quark must be red, one quark must be green, and one quark must be blue. For a meson, one of the quarks might be red, requiring its partner to be anti-red. These combinations of color charge are equivalent to the particles being

colorless. Other exotic assemblies may also exist, but they are very rare. It is worth noting that when a particle is said to be made up of certain quarks, this refers to the valence quarks that give the particle its quantum numbers. Baryons and mesons also contain a sea of quarks throughout them that arise from the splitting or annihilation of gluons. Quark-antiquark pairs can form indefinitely within a particle. These pairs typically recombine shortly after creation.

As previously stated, the leptons can be split into two categories. The first category contains the electron, muon, and tau, and the second category contains the neutrinos. The neutrinos mirror the first category and so they are named after their counterpart. Thus, they are named the electron neutrino (ν_e), muon neutrino (ν_μ), and tau neutrino (ν_τ). Up to this point in time, the leptons are believed to be structureless particles. Additionally, it is believed that both categories of leptons may actually consist of a single particle that can oscillate between resonances that present as one of the three particles in that category. When some interaction containing leptons occurs, in all but the rarest cases, the amount and type of leptons must be consistent throughout the interaction[2]. This is called lepton number conservation.

Conservation of quantum numbers (any assigned property of a particle) is a fundamental exercise of particle physics. This is important because it is a powerful tool to compare theoretical assumptions to experimental observations. Based on theory, certain types of quantum numbers, such as charge and total angular momentum, should be conserved throughout any process e.g. $A + B \rightarrow X + Y$. Similar to conservation of quantum numbers, symmetry laws also dictate which processes are allowed and disallowed. Three important types of symmetry are charge, parity, and time. Charge symmetry says that the laws of physics work the same for a particle and its charge conjugate i.e. an electron and a positron. Symmetry of parity refers to “mirrored” processes, meaning that you would see the same results from a process if you inverted any of its spatial coordinates. Lastly, time symmetry means that a

process happening one way in time occurs the same with the process reversed. For example, we should see the same rate of production of a process like $A + B \longrightarrow X + Y$ and its inverse $X + Y \longrightarrow A + B$.

However, certain observations have led to the discovery of processes that break this “law.” In some cases, processes that fundamentally break conservation laws have been observed. In other cases, theoretical models that correctly predict observed production rates for many processes over- or under-estimate production rates of a small number of processes. This again points to new physics, with conservation or symmetry laws being broken[3][4].

2 Examples of Studies in Particle Physics using Simulated Data and Simulation Software

Possibly the most important tool for Particle Physicists - or any type of physicist for that matter - is the ability to simulate the study they are interested in before or alongside the experiment they are attempting to perform. Simulation can help identify uncertainties, explore feasibility, and test hypotheses using computational methodologies.

Simulation and simulated data have played a part in the majority of high energy physics analyses to date. A typical study will use both simulated data and collected data to assist the measurement of whichever process the researchers are interested in. Methodologies utilizing simulated data include, but are not limited to: projecting detector inefficiencies onto a real measurement, tuning the parameters and specifications of your experiment, estimating the safety of your experiment once it’s built in the real world, informing the re-design of physical components so your experiment succeeds, and testing physical processes under different conditions. In general, an analysis of simulated data can inform researchers on what to expect from real data analysis and formulate corrections for the real data analysis. In this section, two

different experiments to which I contributed will be discussed to exhibit the utility and versatility of simulation in connection with particle physics experiments.

The experiment used for this dissertation research, the *BABAR* Experiment, will be discussed at greater length in a later section. The other experiment, named Mu2e, gave me the opportunity to expand my perspective of how teams of physicists solve problems and formulate their experiments. To describe my experience on both projects, I compare and contrast the simulation strategies of the two experiments I worked on.

Summary of Simulation Methods Used in the *BABAR* Experiment

The *BABAR* Experiment was located at the SLAC National Accelerator Laboratory at Stanford University in Palo Alto, California. This experiment collected one of the world's largest data sets at the time for collisions of a high-energy beam of electrons with a high-energy beam of anti-particles of the electron, the positron. The detector for the *BABAR* experiment halted data collection in 2008, but Physicists are still analyzing the data to this day.

The *BABAR* Experiment produced sets of simulated data (referred to as Monte Carlo or simply "MC"). These were generated to represent events and phenomena as close to reality as possible for Physicists to study before analyzing the real data. The studies done on MC are then used to calculate and derive multiple aspects of the actual measurement being studied. After a set of MC is selected, the researchers then write a program that will hunt for the process they are looking for. The program searches the MC for particles that it thinks represent the process for which it is programmed. These are called "candidates." Since these particles are only candidates, they may not actually be the particle intended to be found. Because of this, the signal of the desired process is likely swamped with false identifications of other particles. Further study can be done in an attempt to clean the set of candidates of

the falsely identified particles.

These preliminary studies on MC can help identify properties within the processes whose values can help us distinguish events of interest from background. These are called “cuts.” A cut is supposed to eliminate a portion of the candidate particles that were identified by investigating the MC. A common way of finding good cuts is to take one of the properties of particle decay (say, momentum or center of mass energy) and make a series of cuts at many different values of the same property. A script can be written to programmatically do this and then analyze the resulting data to determine if the signal became more pronounced by cutting on a particular value. Once a cut is determined, either the events containing particles measured with values violating the cut are discarded or the candidate itself is discarded from the event.

In this manner, my dissertation was completed using data collected by the *BABAR* detector and data simulated throughout the life of the *BABAR* experiment. Precise details and results will be presented in chapters III through V. The *BABAR* Detector attained excellent resolution and the collaboration completed many studies because of the high volume of simulated data and, at the time, one of the largest datasets ever collected.

Corollary to the Discussion of Simulation Methods – The Mu2e Experiment

Background

The second experiment, the Mu2e experiment, is an in-development project at Fermi National Accelerator Laboratory (Fermilab) in Batavia, Illinois. Construction of the facility is under way and software development is ongoing. The Mu2e experiment will measure the rate of neutrinoless conversion of a muon to an electron in the process $\mu^- + N \longrightarrow e^- + N$. The processes which may be observed in Mu2e are shown in Figure 2. Typically when a muon decays into an electron, it must conserve lepton

number[5]. As a result, a muon neutrino and an electron anti-neutrino will also be created. In the Standard Model, direct conversion between lepton types is allowed. It is known that neutrinos are allowed to oscillate among their three different types. However, this phenomenon has not been observed yet for the heavy, charged leptons. Mu2e aims to make the first measurement of direct muon to electron conversion. In the current theory of the Standard Model this process is forbidden or, at best, extremely suppressed ($\sim 10^{-50}$). However, since there is no observation to confirm this, an opportunity to observe new physics is presented. The Mu2e experiment will be sensitive to a rate as low as 10^{-17} conversions per event. Many New Physics (NP) models compute a rate of conversion within the sensitivity of the Mu2e detector, meaning that *any* detected conversion immediately signifies New Physics. If Mu2e is unable to measure any muon-to-electron conversions, at the very least, the Mu2e Experiment will have made the highest sensitivity upper limit measurement of the conversion process. This will also be a useful result because some of the New Physics models predict a conversion rate higher than the new upper limit and will either be forced to accommodate the new discovery or be ruled out entirely.

The Mu2e Detector

Introduction The Mu2e detector is sectioned into three areas each defined by their major component: the Production Solenoid area, the Transport Solenoid area, and the Detector Solenoid area (Figure 3 taken from [7]). These three major components will work together to deliver over 10^{18} muons onto the stopping target over the lifetime of the experiment. The process searched for in this experiment is so rare that such a high intensity is required to see only a few signal events. This means the instrumentation must be highly specialized and exceptionally sensitive. Descriptions of the different aspects of the Mu2e detector follow[7][8].

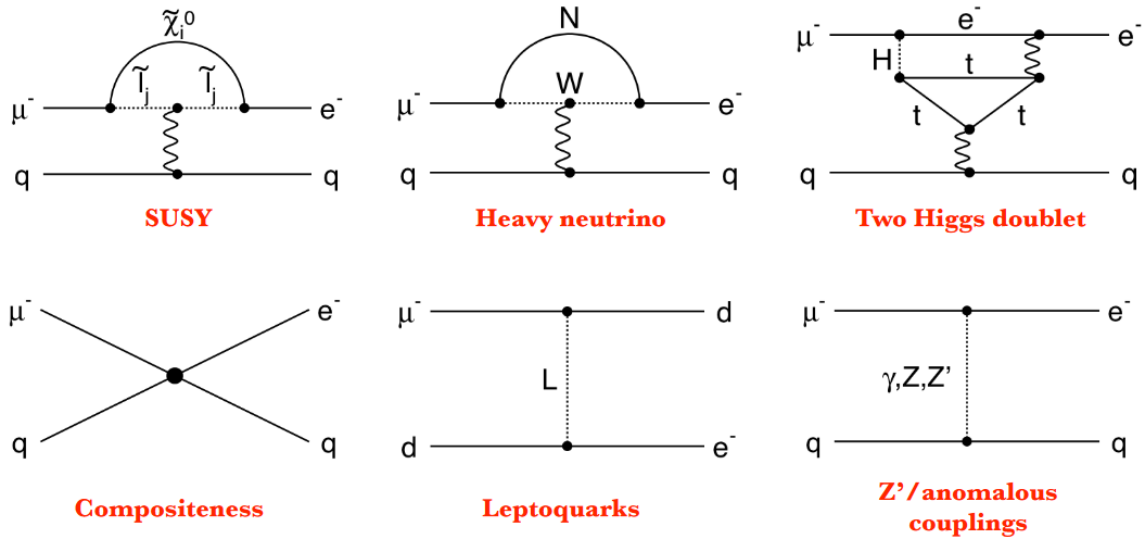


Figure 2. Potential processes in which a muon can convert directly into an electron. A signal observation of any of these processes in Mu2e would indicate new physics. Image courtesy of [6].

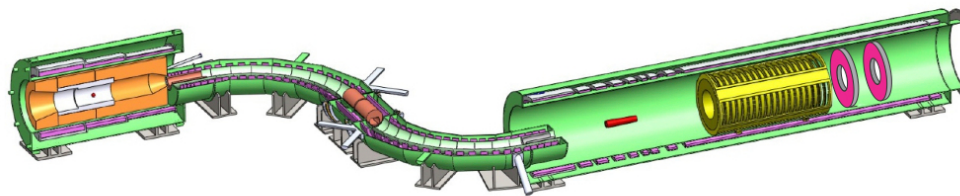


Figure 3. A computer generated representation of the detector components for the Mu2e experiment. From left to right the areas of the detector are: Production Solenoid Area, Transport Solenoid Area (S-shaped solenoid), and Detector Solenoid Area. Image courtesy of the Mu2e Collaboration[7].

Mu2e World Coordinates The Mu2e system has its own set of coordinates that are useful when describing geometry. In the Mu2e Offline software system, which is used for simulation of events, the origin of the coordinates is at the center of the Transport Solenoid (Figure 3). The y -coordinate describes the elevation of the experiment and is positive in the vertically upward direction. The z -coordinate is along the axis of the production and detector solenoids and the x -coordinate is perpendicular to the yz -plane points approximately north.

Production Solenoid Area The Mu2e production target is a pencil sized piece of Tungsten (W). Protons will be delivered to the production target via the evacuated beam pipe from upstream accelerator components. The proton beamline is in the opposite direction of the Mu2e downstream direction. Protons that miss the production target will hit an absorber at the far end of the production solenoid area. Pions created by the collision of protons on the Tungsten target will mostly decay into muons as they are pushed downstream towards the transport solenoid by a gradient in the magnetic field from the solenoid.

Transport Solenoid Area The transport solenoid area is composed of toroidal magnets laid in an S-shape co-axial with the beam stream. Each magnet is gradually angled away from the y -axis. As both positively and negatively charged muons travel downstream, they are separated in the y direction by the magnetic field. Collimators are placed along the transport solenoid with geometry designed to block positive muons and windows designed to block anti-protons. The body of the collimators contain a large volume of high-density, copper-rich alloy to optimally stop positive muons and absorb their energy with minimal bleed-through radiation.

Stopping Target and Detector Solenoid Area The stopping target is a series of 37 aluminum foils. The foils are annular rings with inner diameter 4.25 cm and

outer diameter 15 cm and a thickness of 0.2 mm. The foils are aligned with the centers of their axes co-axial with the flow of particles and are separated by 2.2 cm. When muons hit the stopping target they have a chance to enter the orbit of an aluminum atom and be captured by it, thus ejecting an electron from it. While in orbit around the aluminum atom the muon can decay normally to two neutrinos and an electron, but there is a very small chance (at least in the current Standard Model) that the muon will directly convert to an electron without the presence of neutrinos. Because of the low mass compared to the muon, the electron will have significant and specific amount of energy and will be jettisoned out of the aluminum orbit toward the detector solenoid area.

Building and Shielding The Mu2e building sits on the Muon Campus at Fermilab. The building is behind Wilson Hall and next to the building for the $g-2$ Experiment.

Because of the sensitivity required for the Mu2e experiment and the fact that the very common electron is the signal particle, it is possible for sources outside of the detector to enter the experiment and create electrons. The electrons created by external sources are practically unrestricted in the energy they might have so it is possible for some of these electrons to have an energy at the signal energy for muon to electron conversion. To combat this a complicated series of shielding needs to be created.

Structure of an Event The average event in the Mu2e Experiment begins by injecting protons into the Mu2e detector. Protons will be accelerated up to 8 GeV in the Fermilab Booster and then delivered to the Production Target area of Mu2e via the Delivery Ring. Those protons will collide with a tungsten target in order to create pions. The pions then move towards the transport solenoid due to a large

magnetic gradient. During the push towards the transport solenoid, the pions will decay into, among other particles, muons.

Once particles enter the transport solenoid, a collimator is used to select negatively charged muons to be delivered to the stopping target area. The muons that live long enough to reach the aluminum target have a chance of being captured in the orbit of the aluminum atoms. About half of the muons arriving at the stopping target are stopped in the aluminum target. To achieve this, they knock out an existing electron in the orbit of the aluminum atom.

At this point, the muons in orbit around an aluminum nuclei will mostly decay in the typical form into an electron and two neutrinos, to conserve lepton flavor, or be absorbed by the aluminum nucleus. It is possible, however slighty, that the captured muons will directly convert, without a neutrino, into an electron.

Any electrons originating near the stopping target, regardless of the process that created it, will continue towards the experiment's primary sensitive detectors. Data will be collected from these electrons as they pass through the tracker and deposit energy into the calorimeter.

Background Events Since the signal for the Mu2e experiment is a lone electron (one of the most abundant and naturally occurring particles in the universe) and the process is so rare, great consideration must be given to eliminate any background events.

The largest background in Mu2e comes from muons that decay normally while orbiting the aluminum nucleus. About 39% of the muons that are stopped in the orbit of the aluminum nucleus will decay naturally into an electron and two neutrinos. This is called the decay-in-orbit (DIO) background. The electron that is produced from muon DIO will be ejected from the orbit and can travel towards the tracker and calorimeter. The spectrum of the energy of the electrons coming from muon decays

peaks at about 58 MeV. However, a long tail on the upper end of the spectrum reaches out to the signal energy with a production rate on the order of the signal event. This process accounts for slightly more than half of all of the anticipated background for Mu2e at a rate of 0.20 events over the lifetime of the experiment.

The second largest background is anticipated to come from cosmic rays. Cosmic particles entering the detector area have a chance of interacting with detector components and producing an electron or a photon that can fake a signal event. An intricate shielding system encloses the detector in order to block cosmic rays and also to sense cosmic rays that get through the shielding. When a cosmic ray is identified, the event coinciding with it, the one before it, and the one after it are thrown out. This “Cosmic Ray Veto” (CRV) is expected to have an efficiency of 99.99%. The total background contribution due to cosmic rays is expected to be 0.08 events.

Among the other background contributions are electrons coming from muons or pions captured by the aluminum nucleus and electrons produced by the decays of muons or pions that are not stopped by the aluminum target or that decay in-flight before being stopped by the aluminum target[9].

Software Development on the Mu2e Experiment

Introduction An essential part of the Mu2e experiment is the software that aids the scientists in planning, developing, and running the experiment. In a particle physics experiment, it is paramount to create and understand a simulation of the target processes involved. The simulation is developed long before an actual building or detector is built or implemented. As part of my work, I helped develop the geometry of the detector building in the software for Mu2e.

Mu2e Building Geometry

Introduction The Mu2e building located on-site at Fermilab will be a very specialized building with many complex pieces working together to contain and deliver a successful high-intensity particle physics experiment. To make the experiment successful, the collaboration will have to create a supremely accurate representation of the real world building and experiment components in software form. One component of the Mu2e software is the simulated model of the building. The model of the building has to be very precise, down to the representation of wires and pipes. Equally important is the concrete that makes up and supports the building, and the supplementary concrete that will be placed to block the high amount of radiation throughout the experiment.

One hurdle to overcome in the creation of the Mu2e geometry is the ever-changing plans of ongoing real world construction. As a project in development, a practical model of the Mu2e building was created in the software. This model was also a representation of the early drawings of the building. The model was used to do the first simulations of the physics processes involved in the Mu2e experiment. The largest development on geometry at this time was focused on the components of the detector and the shielding closest to the detector.

Eventually, as the building was constructed, the plans were revised to accommodate extra instrumentation, reflect structural engineering considerations, add stairwells and hallways for personnel access, and design the part of the building from where the scientists will control the experiment. As the building progressed closer to completion, the software development needed to adapt by creating a more accurate digital representation of the building. Studies on the radiation delivered to certain areas of the building (particularly to where the scientists will be located) were desired and so a realistically accurate representation of the building in the simulation was of utmost importance.

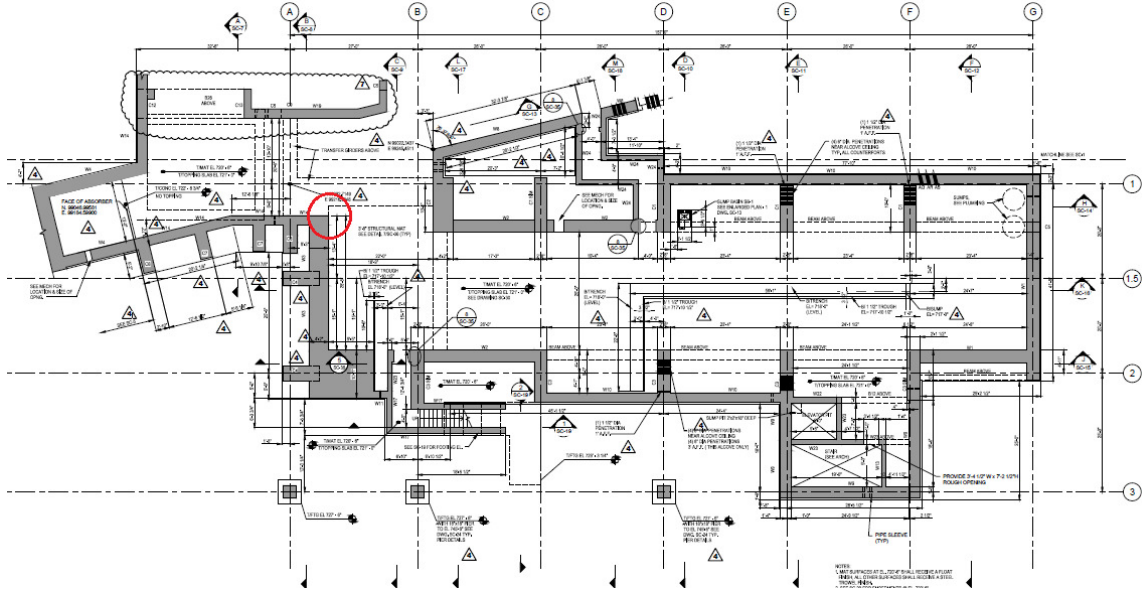


Figure 4. Plans for the lower level of the Mu2e building in the Production Solenoid-Transport Solenoid-Detector Solenoid areas. Circled is the origin for the geometry building software. Image courtesy of Fermilab.

Conventional Construction Geometry As a part of my work on the Mu2e project, I helped convert the geometry used for Mu2e simulations from the bare-bones initial design into the as-built version of the Mu2e site. This included interpreting the building plans and encoding them into the software. Figure 4 is a page of the plans for the detector level of the Mu2e building in the Production Solenoid, Transport Solenoid, and Detector Solenoid areas. For creating the geometry, an origin position was chosen as the corner of two walls at the transition of the Production Solenoid and Transport Solenoid areas at the floor of the detector level of the building (circled in red in Figure 4). The geometry was made by plotting points in the xz -plane with respect to the origin position and then extruding the piece in the y -direction by some distance.

A nomenclature to perform geometrical operations was written by the collaboration to feed into software separate from the simulation software. This software interprets the nomenclature to build the individual geometrical pieces by taking their as-built drawing dimensions and automatically creating the pieces in the Mu2e world

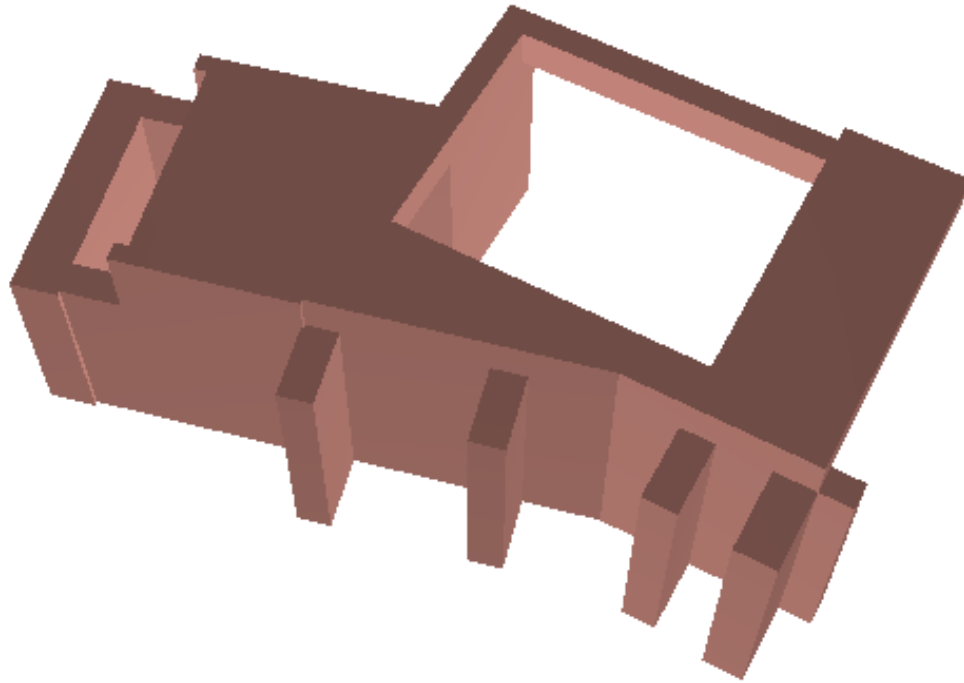


Figure 5. Building geometry pieces in the Production Solenoid Area produced by specialized software for interpreting drawing plan coordinates and extruding to create the individual pieces. Image created by the author using ROOT[10].

coordinates. The software works by walking along the perimeter of the piece vertex by vertex and extruding in one direction to give the piece thickness.

Each piece was given its own file in the software. These pieces then could easily be inserted into the comprehensive Mu2e simulation software. One novelty of using the nomenclature software to build these pieces is that the software can automatically create adjacent features to the piece being coded. For example, a piece of exterior wall might be adjacent to earth extending to the edge of our modeled space. We could create the dirt model at the same time as the wall by indicating the directions in which the dirt extends from the wall.

Because the final geometry needs to be an accurate representation of the physical

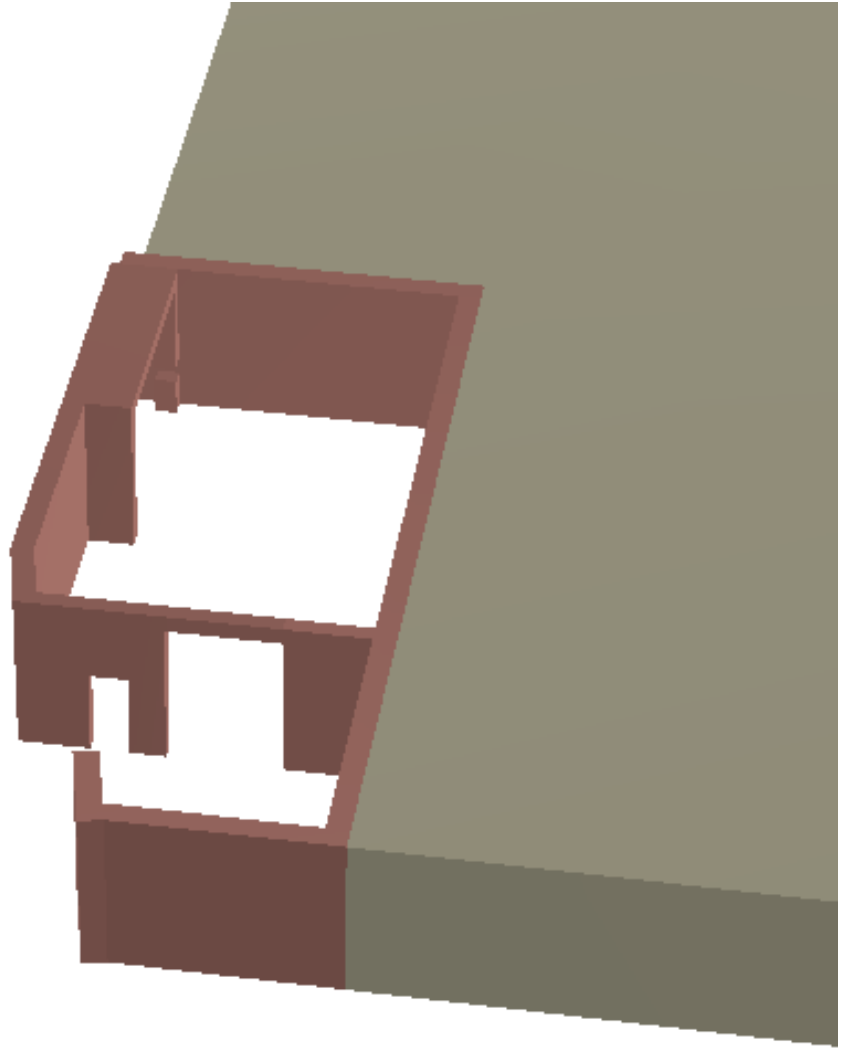


Figure 6. A demonstration of creating Mu2e geometry pieces with automatically created dirt pieces. The reddish-brown piece is the exterior remote handling area wall. A dirt piece has been generated in grey-brown. Image created by the author using ROOT[10].

building, all of the dirt and backfill surrounding the building needed to be as detailed as the building itself. Using the software to create a large amount of dirt pieces simultaneously with the building pieces was very efficient and avoided the problem of creating pieces that inadvertently overlapped. Figure 5 and Figure 6 show examples of a few geometry pieces that I created with the software using the developed nomenclature. The reddish-maroon colored pieces are parts of the Remote Handling area on the bottom floor of the detector building. The grey-brown piece is a dirt piece that was automatically generated.

As a complement to my study of neutral hadron production on *BABAR*, my experience with Mu2e helped diversify my breadth of knowledge and let me collaborate in another environment on an experiment with very different motivations from *BABAR*.

CHAPTER II

THE *BABAR* EXPERIMENT

1 Background

The *BABAR* Experiment was undertaken by a multinational collaboration based at the Stanford Linear Accelerator Center (SLAC – now known as the SLAC National Accelerator Lab). SLAC is located in Palo Alto, California, on land owned by Stanford University. The university also serves as the managing body of the laboratory. *BABAR* was proposed as part of a “*B*-Factory” project by Jonathan Dorfan, David Hitlin, and Per Oddone. It was approved by President Bill Clinton in October 1993.

B-Factories are facilities that meet specific requirements in order to generate samples of *B* mesons that allow for measurements of processes that violate *CP* symmetry. Specifically, these facilities must produce an instantaneous luminosity of at least $10^{33} \text{ cm}^{-2}\text{s}^{-1}$ and have a beam energy asymmetry sufficient to cause a *B* lifetime in the lab frame long enough to observe its point of decay[11]. A discussion of these concepts follows.

The *BABAR* experiment was largely concerned with the production of Υ particles derived from the annihilation of electrons with positrons at center-of-mass energies between 9.4 GeV - 10.6 GeV. Of particular interest to the experiment is the $\Upsilon(4S)$ resonance at a center-of-mass energy of 10.58 GeV. $\Upsilon(4S)$ is a member of the bottomonium family of particles; that is, particles that contain a *b* quark and a \bar{b} quark. Figure 7 shows the spectrum of the bottomonium family[12]. By being constituents of the s-shell in the bottomonium family, Υ particles can be created in electron-positron collisions due to the quantum numbers matching. Below the $B\bar{B}$ line indicated in

Figure 7, also known as the OZI threshold[13][14][15], Υ s decay directly or indirectly through annihilation of the b and \bar{b} quarks. At the $\Upsilon(4S)$ resonance the bottom and anti-bottom quark can have enough energy such that it becomes energetically favorable to create a new quark/anti-quark pair instead of staying bound to each other. Each of the new particles creates a bound state with one of the bottom quarks and becomes a B meson. Figure 8 shows the Feynman diagram describing the $\Upsilon(4S)$ decaying into a B^0 meson and a \bar{B}^0 meson. The capability of the $\Upsilon(4S)$ to decay into B mesons, nearly 100% of the time, made it favorable to run the experiment at 10.58 GeV. In order to create many e^+e^- interactions, and therefore as many B mesons as possible, *BABAR* maximized its luminosity to gather a lot of data. Luminosity is a measure of the number of collisions over an area per second. In this case, we are referring to the fact that, though the center-of-mass running energy in *BABAR* can be tuned, the $\Upsilon(4S)$ is the perfect avenue for B meson production and so the experiment had its greatest run time at the resonance energy. Figure 9 shows the total luminosity of the *BABAR* experiment during its lifetime[16]. The processes by which the B meson is created and decays is important to answering fundamental questions about the universe.

One highly studied problem is the matter over anti-matter preference of the universe. In most particle physics processes, matter and anti-matter are created in equal amounts. If the high energies attained at particle detectors probe the conditions of the early universe, then the processes that are observed must represent in some way the evolution of the universe shortly after the Big Bang. Today, the universe is obviously matter dominated. *BABAR* sought to investigate one possible process by which matter is produced more than anti-matter, which is violation of CP symmetry. The C and P both stand for a certain type of parity - an operation done on a state that describes its symmetry. P is the parity symmetry of the spatial coordinates of a state. An operation of parity symmetry exchanges each spatial coordinate with its

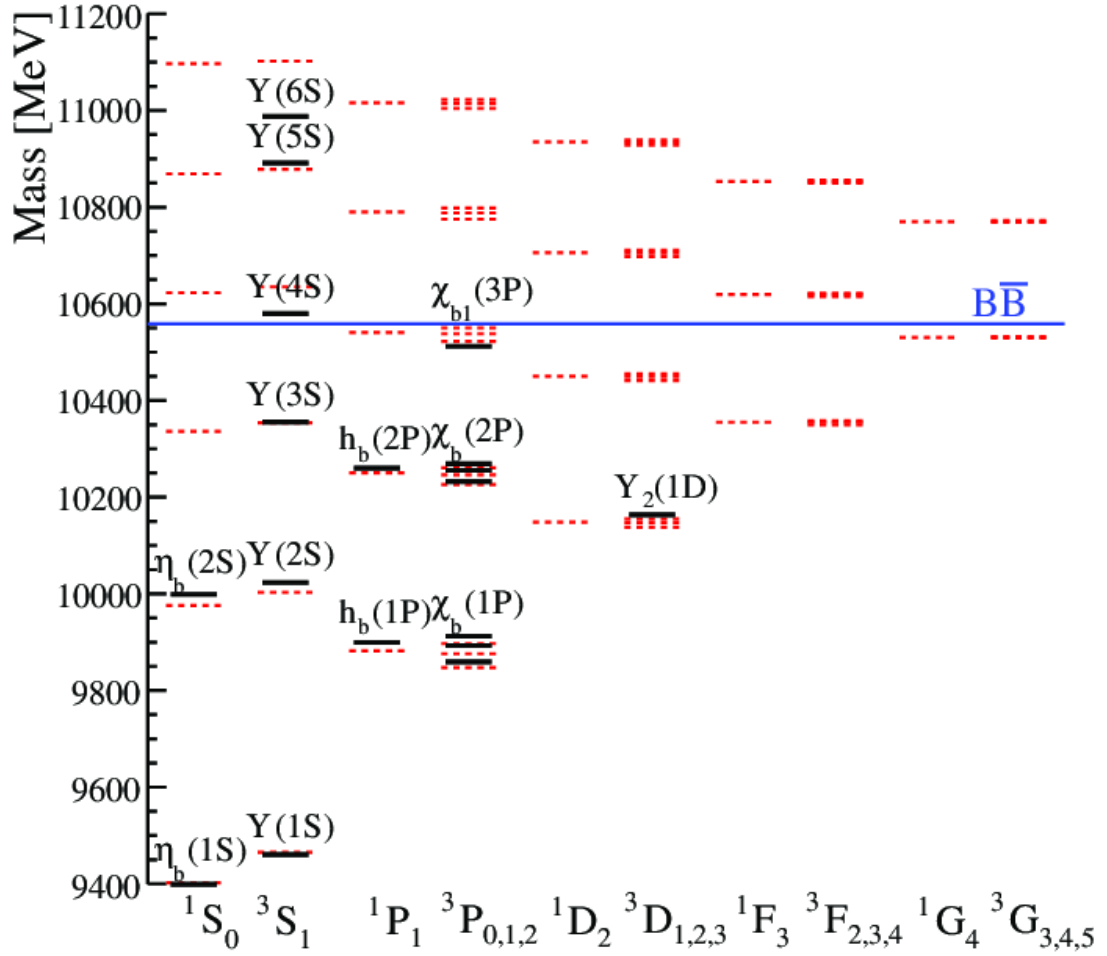


Figure 7. The spectrum of bottomonium. The dashed lines are the masses of the states calculated in theory, while the solid lines are the experimentally established masses of each state. Above the blue line B mesons live long enough to observe their decay processes. Image courtesy [12].

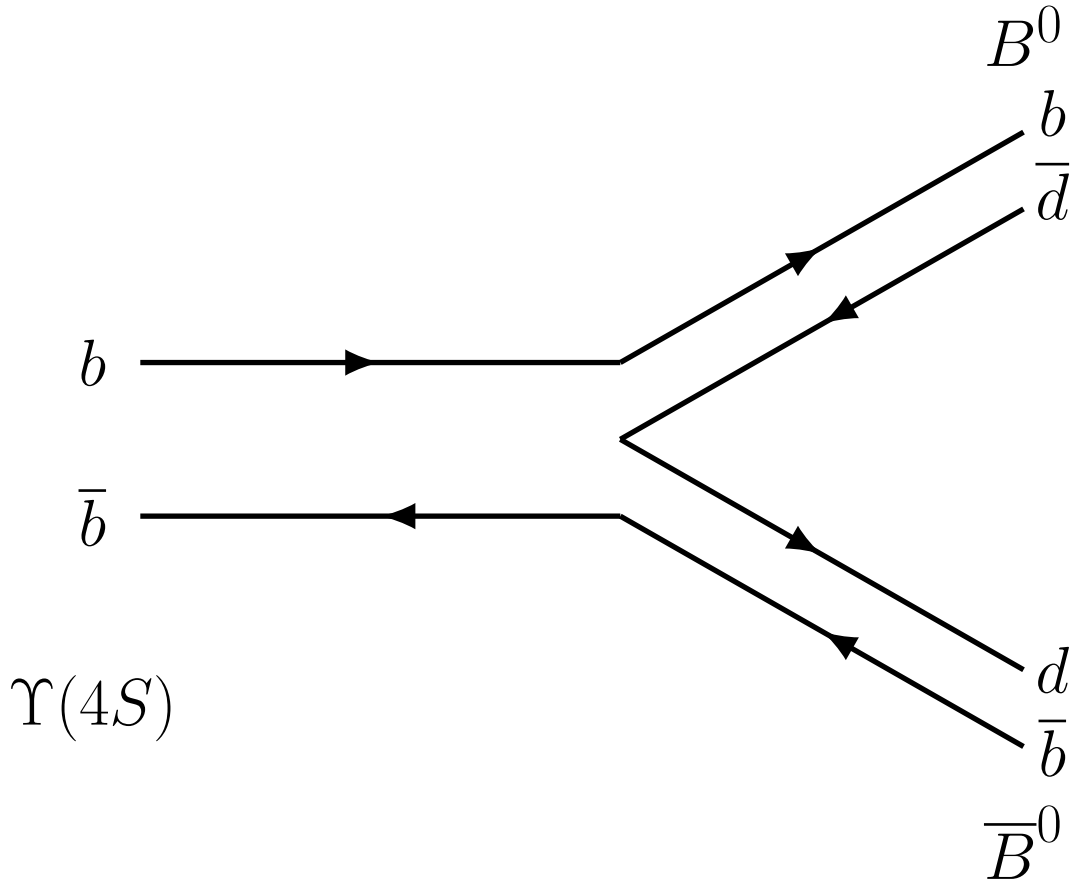


Figure 8. The Feynman diagram for $\Upsilon(4S) \rightarrow B^0\bar{B}^0$ created by the author with the “feynman” L^AT_EX package from [17].

negative, creating a “mirror image” of a state in space. C parity, or charge parity, describes what happens to a state upon the conjugation of the sign of its quantized charge values. For example, the π^+ is turned into the π^- upon charge conjugation, which can be seen by changing the up quark and the anti-down quark into their antiparticle. Combining these two operations creates a total mirror image of a physical system. The spatial coordinates are flipped as well as the quantum numbers that make up the constituents of the system. Since this is a symmetric operation, the qualities of interactions and decays should be the same for a state’s CP transformed state. This is true for both electromagnetic and strong interactions. However, the Standard Model predicts that a small amount of CP violation can happen in some

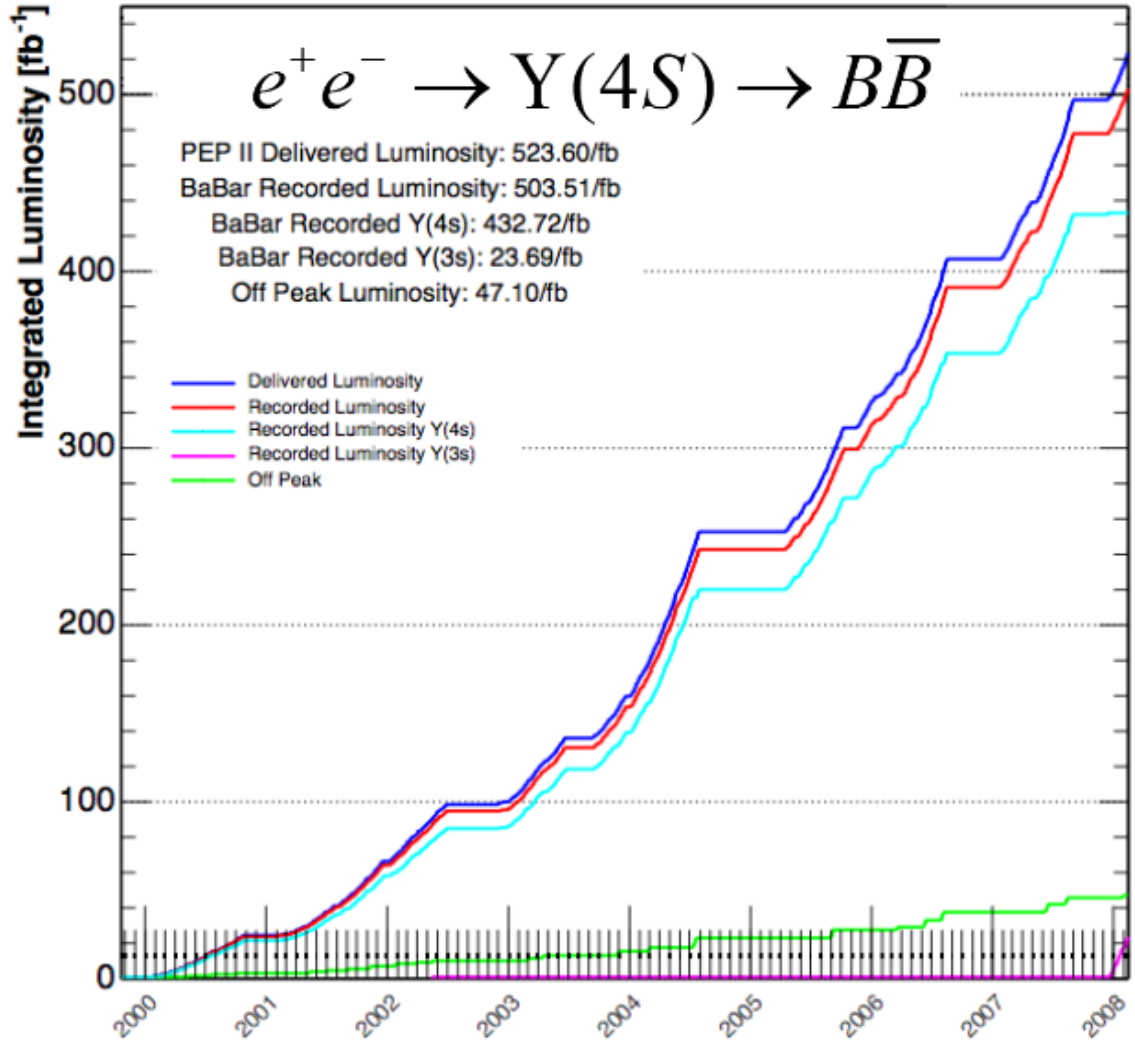


Figure 9. Integrated luminosity totals over the lifetime of the *BABAR* experiment.

specific weak interactions. It is one of the goals of the *BABAR* experiment to determine whether CP violation occurs in expected amounts or if the amount observed is lower or higher than expected[18]. In the latter case, this would indicate that new physics exists.

Existence of CP violation is the reason why B mesons are especially important for the *BABAR* Experiment. Because of their relatively high mass, B mesons have an abundance of decay modes with many of those modes being CP “eigenstates.” These are states in which the B and its anti-matter twin can access the same decay state. The high amount of possible CP eigenstates assures a likelihood of asymmetries

throughout the B system. While the B meson has a high mass, it is actually the lightest particle with a bottom quark constituent. Any decay from a B must result in particles without a bottom quark because the energy to create a bottom quark is inaccessible. This means that a quark flavor change must occur. Since quark flavor changes happen only via the weak force, *every* B meson decay is a probe of the weak force. The weak force is the only force known to violate P and CP symmetry[3], so each B created is a chance to observe CP violation.

2 The *BABAR* Detector

Introduction

The physical components of the *BABAR* Experiment comprise a large assembly of machines, buildings, and land area. A detailed summary of the particle generation, transport, collision, and detector follows. The journey of the electrons and positrons begins in the 2 mile-long linear accelerator (Linac) at the SLAC National Accelerator Lab (Figure 10)[19]. An electron gun is stationed at a point along the Linac to produce a beam of electrons that are accelerated to 25 GeV. These electrons are diverted by a dipole magnet shortly before they reach the PEP-II rings and collide with a Tungsten block, producing a shower of particles. Positrons from that shower are selected by a magnetic field and are collected in a transport beam which delivers them back to the start of the Linac. At this point, the positrons are accelerated up to 3.1 GeV out of phase with another beam of electrons, this time at 9 GeV. The electrons and positrons travel down the Linac and enter the PEP-II storage rings.

The PEP-II storage rings reused many of the components of the retired PEP-I storage ring. Whereas PEP-I's purpose was to collide electrons and positrons in a single storage ring, PEP-II was created with two rings to accommodate *BABAR*'s novel approach to generating B mesons by using asymmetric beam energies. The two rings have the same dimensions and are spaced about one meter apart. The rings are

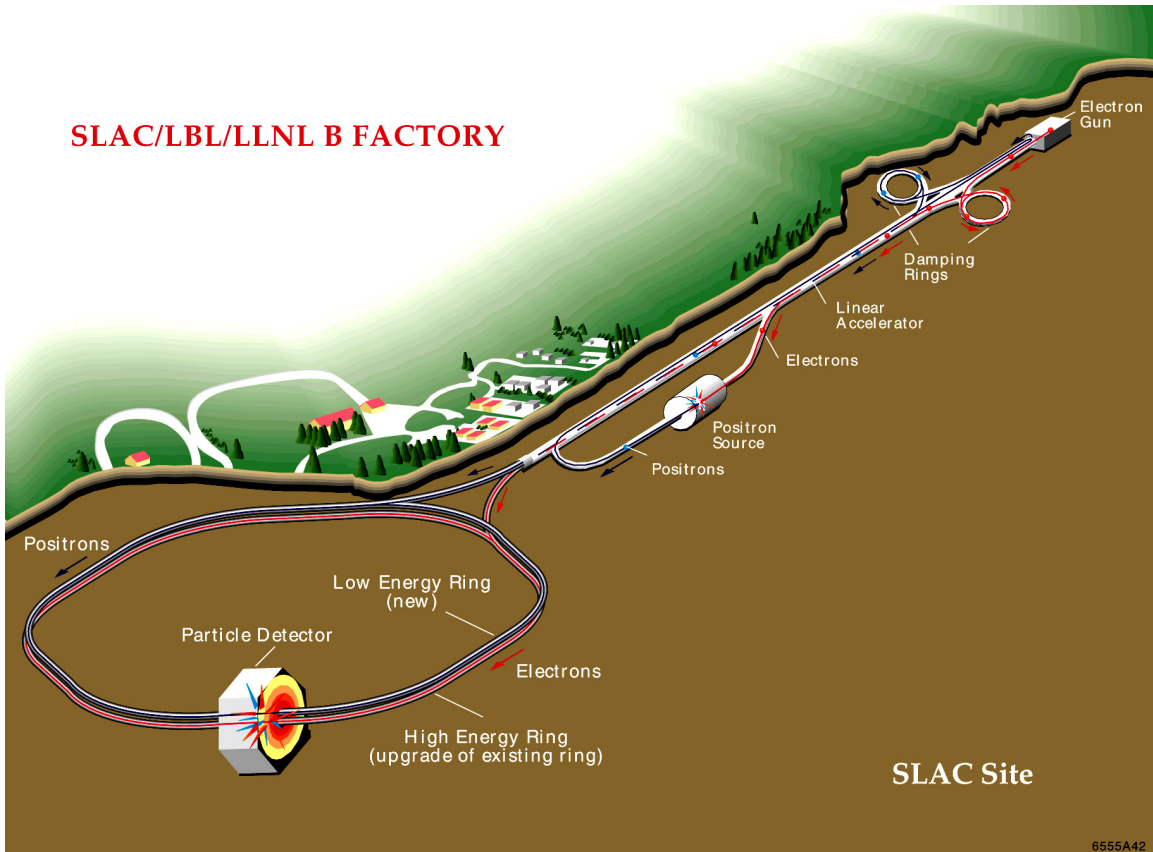


Figure 10. A diagram of the SLAC facility used for *BABAR*. Image courtesy SLAC[19].

roughly hexagonal in shape when viewed from above, rather than completely circular. The six curved portions are each 250 m in length and the straight portions are 120 m in length, for a circumference totaling 2220 meters. When the particles enter PEP-II, the positrons enter the upper Low Energy Ring (LER) and the electrons enter the lower High Energy Ring (HER). After circulating in the rings at a rate of 135 kHz, the positron beam is brought down to the HER and the two beams collide at the interaction point head-on in the detector. Once the particles have collided or scattered, the products of those interactions enter the various subsystems of the *BABAR* detector.

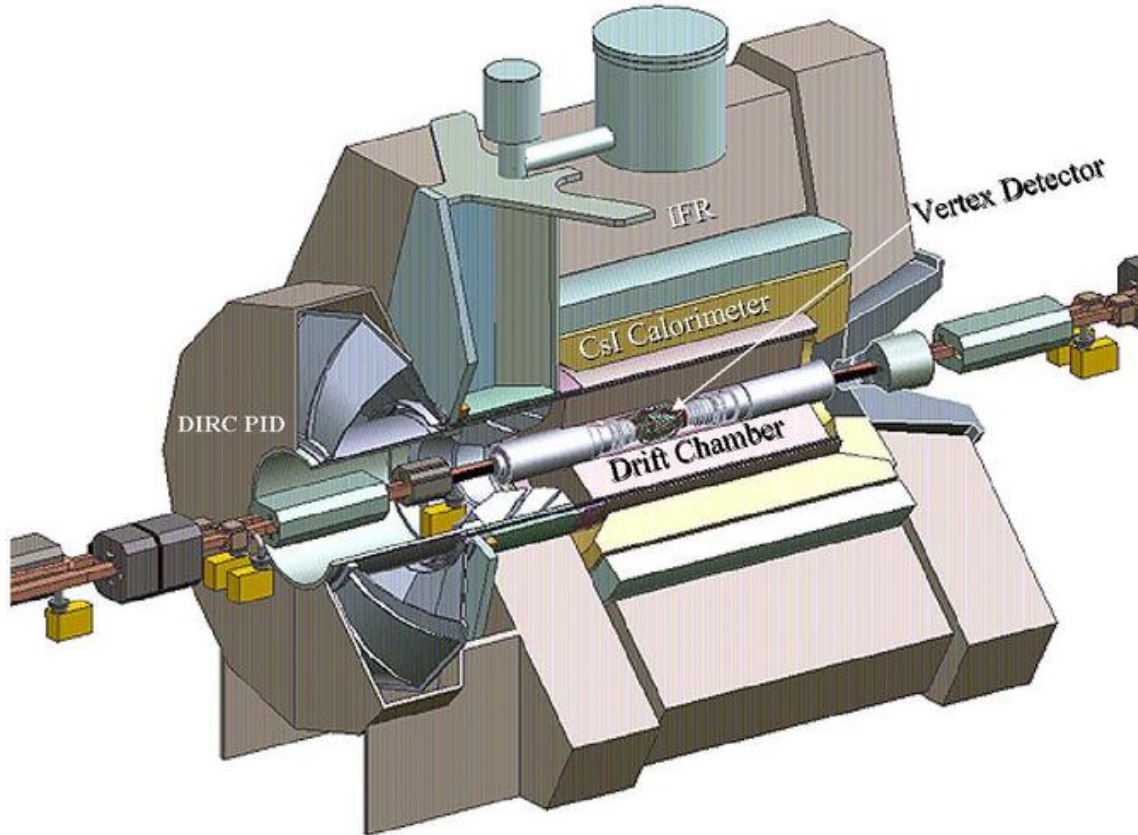


Figure 11. A cut-out view of the different components within the *BABAR* detector. Image courtesy [20].

The Subsystems of the *BABAR* Detector

The *BABAR* detector can be roughly described as the combination of five subsystems. Starting from the beamline and moving outward, they are the Silicon Vertex Tracker, the Drift Chamber, the Detector of Internally Reflected Cherenkov light, the Electromagnetic Calorimeter, and the Instrumented Flux Return. There is also a superconducting coil which provides a strong magnetic field (1.5 Tesla). Figure 11 shows a cut-out view of the different subsystems[20].

The Silicon Vertex Tracker

The Silicon Vertex Tracker (SVT) is the innermost component of the *BABAR* detector. The primary function of the SVT is to collect precise positioning data of the paths

(called “tracks”) of charged particles as they move through the component. Of specific interest, the SVT was designed to measure the distance between the vertices of the two B mesons created from the decay of an $\Upsilon(4S)$ particle. This distance can be used directly to make a measurement of CP violation. The SVT also has the duty of characterizing low momentum tracks that do not make it to the outer components of the detector.

Each panel of the SVT has five layers of silicon sensors each of which has a distinct shape and composition. The panels are then mounted cylindrically around a frame and encompass the interaction point. The SVT is the nearest component to the beamline and, as such, is the first level of detection in the *BABAR* detector. Because of this, the angular acceptance for the entire experiment is set by the dimensions of the active area of silicon in the vertex tracker. The constraints imposed by the machine components surrounding the individual detector subsystems resulted in the active silicon reaching in the forward direction all the way to about 1 cm away from the B1 magnet.

The SVT also has the duty of collecting the majority of the data to reconstruct tracks with $40 \text{ MeV}/c < p_t < 100 \text{ MeV}/c$ since those particles have tracks with small radii that do not spend a significant amount of time in the outer subsystems of the detector.

The Drift Chamber

The next component of the *BABAR* detector is the Drift Chamber (DCH). The Drift Chamber consists of 7104 $12 \times 18 \text{ mm}^2$ cells filled with a helium-isobutane gas mixture, which reduces the effect of multiple scattering and increases resolution. The cells are arranged in 40 cylindrical layers. The DCH is the primary tracking component of the *BABAR* detector. Its job is to provide high resolution momentum measurements for charged particles and can also make particle identifications by measuring dE/dx –

the energy deposited by a charged particle via ionization, normalized per unit length of its path.

When a charged track enters the drift chamber it can ionize the gas in the cell. The electrons created by the ionization drift towards wires inside the cell and a signal is recorded. The position of the charged track can then be calculated by measuring the time that the electron takes to reach the wire.

For the DCH to accurately measure the B physics that *BABAR* is interested in, it must be able to precisely record the momentum of charged tracks in a certain range, $60 \text{ MeV}/c < p_t < 2.5 \text{ GeV}/c$. Spatially, however, the inner radius of 22.5 cm of the chamber sets a lower limit on the resolution of charged particles at $100 \text{ MeV}/c$ due to the short length of the track's path inside the chamber.

The DCH is also offset from the center of the interaction point due to the boosted nature of the collisions. This reduces the dependency on the center of mass angle for resolution measurements. The Drift Chamber is able to make measurements from within the full polar angle range allowed by the design of the beamline components. This range is $-0.87 < \cos \theta_{lab} < 0.96$.

The Detector of Internally Reflected Cherenkov Radiation

After the Drift Chamber, the next detector subsystem is the Detector of Internally Reflected Cherenkov Radiation (DIRC). The primary function of the DIRC is to identify hadrons passing through the detector. This component of the detector is composed of 144 quartz bars each 1.7 cm thick, 3.5 cm wide, and 490 cm long. They are arranged radially such that they create a 12-sided polygon coaxial with the beamline. The bars are mounted inside of a support tube. At one end of the support tube sits a toroidal chamber filled with water that houses photon detectors; this is the "Standoff Box". It houses around 11 000 2.5 cm diameter photomultiplier tubes.

As such, the DIRC measures the Cherenkov angle of particles passing through

the quartz bars. Cherenkov radiation is light that is created by a charged particle propagating through a material faster than the speed of light within that material. In other words, if $\beta = v/c$ where v is the speed of the particle in the material and c is the speed of light (in a vacuum), then Cherenkov radiation will be emitted when $\beta n > 1$.

Particles traveling through the quartz bars will produce photons in this manner, sort of like a “sonic boom” of light. The photons emitted will have a certain angle associated with the speed of the particle. The photons then totally internally reflect through the quartz bars, preserving the angle (the Cherenkov angle, θ_C), and eventually reach the phototubes in the standoff box. Pattern recognition software determines the Cherenkov angle for each charged particle of sufficient momentum from the distribution of phototube signals.

The combination of momentum and Cherenkov angle measurements provides high resolution particle identification, particularly good for distinguishing particles at high momentum.

The Electromagnetic Calorimeter

Directly incident on the outside of the DIRC is the next component of the detector, the Electromagnetic Calorimeter (EMC). Its job is to identify electrons, neutral electromagnetic particles, and hadrons as well as measure their position and energy. It is constructed from 6580 CsI crystals that are doped with thallium iodide. The crystals are shaped like trapezoidal pyramids which are truncated at the top. The crystals are arranged in 48 rows of 120 crystals in the barrel region, with an inner radius of 90 cm. The crystals are also angled projectively along the beam-line direction. In the boost direction there is an endcap made of 820 crystals in 8 radial rings with the first, smallest ring starting at a radius of 55.3 cm. The coverage of the EMC is thus $-0.78 < \cos \theta_{lab} < 0.96$.

The Instrumented Flux Return

The outermost component of the detector is called the Instrumented Flux Return (IFR). This component has two purposes. First, the IFR is composed of layers of iron and steel in order to serve as a flux return to the magnetic solenoid. Second, between the layers of the flux return are active detectors that can detect muons and high-momentum pions which live long enough to reach this part of the detector[21].

CHAPTER III

PRODUCTION OF INCLUSIVE NEUTRAL HADRONS

1 Introduction

The preceding discussion of the *BABAR* experiment has, up to this point, all pertained to the study of $\Upsilon(4S)$ decays. The detector run-time that was spent on this particular phenomenon was predominantly performed at the optimum production energy for the particle. This is called running “on-peak”, as there is a spike in the production of the particle at the resonance energy. Figure 12 shows the cross-section of the resonances of the Υ at each resonance energy[22]. As well as doing studies on-peak of the $\Upsilon(4S)$, the *BABAR* experiment spent detector run-time running “off-peak” at energies around the optimum resonant energy. Studies performed off-peak produce their own bevy of phenomena and are good probes of processes of Quantum Chromodynamics (QCD) – the Standard Model theory of how the strong force works. In addition, these studies can be particularly useful to other experiments as a general “cross-check.”

The data simulated by the *BABAR* collaboration and the data collected by the *BABAR* detector for this study was generated at a center-of-mass energy of $10.54 \text{ GeV}/c^2$. Figure 12 shows that this energy is considered “off-peak” for the Υ .

One set of particles that *BABAR* isn’t directly sensitive to is neutral hadrons. The off-peak signal of these particles don’t contribute to processes potentially involving CP violation, but they are important to understanding the hadronization process of particles. Hadronization is the process by which quarks are created and coalesce into mesons and baryons from pure energy. In *BABAR*, when the electron and positron annihilate there is a chance that the energy created by that annihilation will end

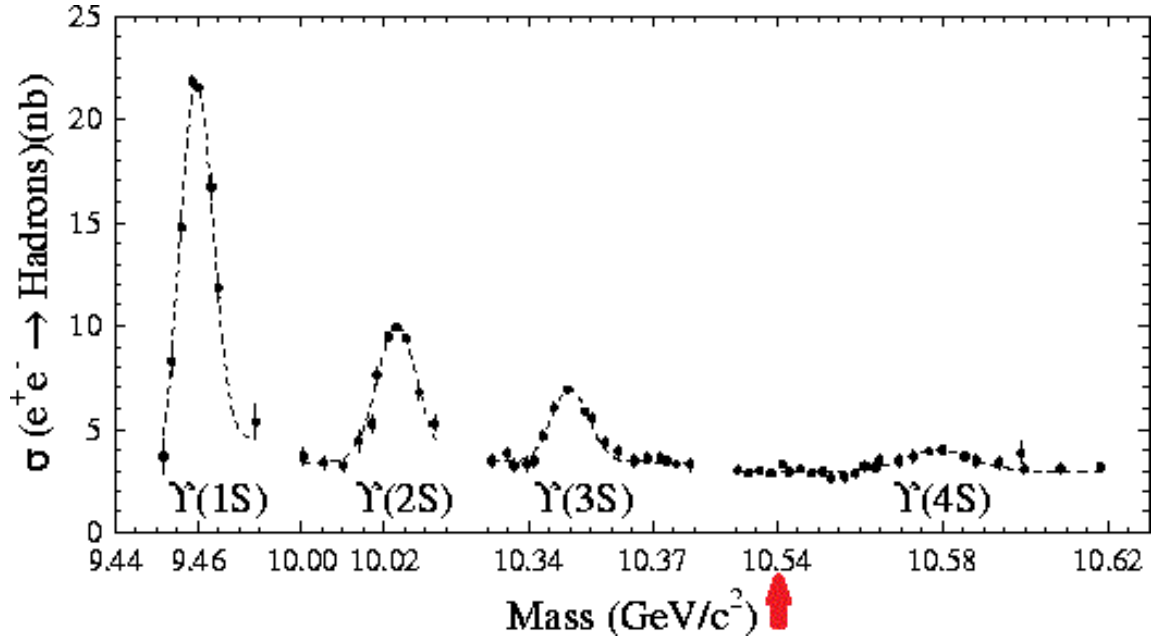


Figure 12. Resonance peaks of Υ . Red arrow indicates center-of-mass energy for this study. Image courtesy [22].

up in the form of hadrons. This is a fundamental process in particle physics, but the exact avenue for hadron production is unknown. Understanding and deriving the production of neutral hadrons will complete another piece of the puzzle to the Standard Model.

This study is a measurement of the production rate of each of a pair of neutral hadrons per event at a specific energy. These neutral hadrons have similar and related quark make-up but, due to spin state mixing, present as distinct particles. Depending on the center of mass energy of the electron-positron collision, the measurements will vary. The measurements found in this study will be useful for theorists trying to understand the physical underpinnings of hadronization and will also be useful for experimentalists trying to disentangle the products of hadronization from the signals they're interested in for their studies.

2 Event Selection

In this dissertation, I will be measuring the production rate, also known as multiplicity, of neutral hadrons, h , in events of the type

$$e^+e^- \rightarrow \gamma^* \rightarrow q\bar{q} \rightarrow n_h h X,$$

where h can be a π^0 or η , n_h is the number of the hadron produced in a given event, and X is the collection of other hadrons in the event. Figure 13 shows a Feynman diagram of this process. Multiplicity is then defined as

$$R(h) = \sum \frac{n_h}{N_{events}}.$$

To help narrow in on a specific process to be studied, the University of Louisville High Energy Physics (UofL HEP) group has used the detailed knowledge of the particle production in each dataset to develop a standard event selection as a first layer of cutting of the data. These cuts are used as a baseline for both analyses in this study. In particular, our group is interested in QCD processes (i.e. containing quarks) and so these standard cuts aim to remove as many non-QCD events as possible.

The first major cut is on charged tracks in order to cut out Bhabha and di-muon events. Bhabhas are events where an electron and positron collide with or scatter off of each other in the detector, emit a virtual photon, and the only products of the interaction are an outgoing electron and positron. Events such as this make up the vast majority of events produced by *BABAR*. Di-muon events are similar, with the difference being that in a e^+e^- annihilation, a μ^+ and a μ^- are the only products created. The same phenomena with the third generation of leptons is also allowed, but created in a very small amount – τ^+ and τ^- may also be a product of an electron

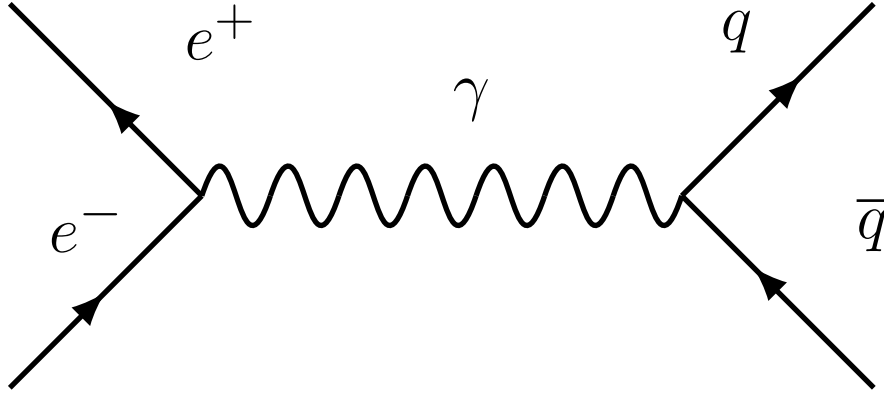


Figure 13. The Feynman diagram for electron-positron annihilation creating hadrons. Processes from $q\bar{q}$ can create additional hadronic particles. Created by the author with the “feynman” L^AT_EX package from [17].

and positron collision. In other words,

$$e^+e^- \rightarrow e^+e^- \text{ (Bhabha)}$$

$$e^+e^- \rightarrow \mu^+\mu^- \text{ (mu-mu or di-muon)}$$

$$e^+e^- \rightarrow \tau^+\tau^- \text{ (tau-tau)}$$

are all excluded in this manner

This is achieved by removing all events in which less than three charged tracks are detected. It follows that since an electron (charge -1) and a positron (charge $+1$) are the only products of a Bhabha event, they have two charged tracks and thus are excluded from the selection of events. This also applies to di-muon events for the same reason. The majority of QCD events at the experiment’s CM energy are going to have at least three charged tracks, so this is a good way to select events we’re interested in.

The second event selection ensures that the total charge of all tracks within each accepted event have $-1 \leq Q_{total} \leq 1$, that is, a total charge close to zero. This eliminates beam-pipe and beam-gas background events which come about by stray

electrons or positrons colliding with material in the beam pipe or colliding with a gas molecule inside of the beam pipe. This causes many protons (+1 charge) from the nuclei of the molecules to spray into the detector. These extra “garbage” particles get tagged to an event and the event is recorded with excess charge.

The next event selection requires the measurement of “R2” to be less than 0.9. This eliminates more Bhabha and di-muon events as well as some tau events. A minimal amount of hadronic events do get cut by this selection as well. The quantity “R2” is the ratio of the second to the zeroth Fox-Wolfram moments. The Fox-Wolfram moments, H_ℓ , can be defined as

$$H_\ell = \sum_{i,j=1}^N \frac{|\vec{p}_i||\vec{p}_j|}{s} P_\ell(\cos \Omega_{ij}),$$

where $P_\ell(\cos \Omega_{ij})$ are the Legendre polynomials. The Fox-Wolfram moments are used as a tool to analyze the geometric details of particle jets as a result of collisions. They help reduce more Bhabha, di-muon, and tau events because an R2 close to 1 corresponds with jets that leave the interaction point back-to-back ($\cos \theta = -1$)[23].

The fourth event selection is to get rid of events with a total visible energy recorded significantly far away from our center-of-mass energy, namely $E_{total} < 5.0$ GeV and $E_{total} > 14.0$ GeV. This cut has a number of purposes. This eliminates more events that may detect particles from outside not created by the collision or events where particles interact with the beam-pipe or beam-pipe gas and deposit their energy in the material’s molecules. This also rejects events that may have had poor calorimetry recorded. Additionally, the rare occurrence that multiple events happen at nearly the same time are cut by this requirement since the energy recorded for the event would be contributed by each of the simultaneous collisions.

The next two selections require that the primary vertex of the event is reasonably close to the center of the detector, where the collisions are supposed to occur. This is

another selection that reduces garbage events from being included. Events that pass this selection are close to the axis of the detector, increasing the likelihood that they were the result of a collision. Events that inadvertently record particles that were created at the beam-pipe or within the beam-gas could cause the apparent origin of the particles to differ from its actual location. The two conditions that are required of the primary vertex are that the z -position is within 5 cm and the distance radially from the axis is less than 5 mm. The z -position is the distance from the center of the detector that's along the beam path direction. The z -position has a higher tolerance as long as the origin is very close to the central axis of the beam path. Requiring the event to have a primary vertex near the center of the detector greatly increases the probability that an accepted event originated from a collision from the two beams centered on the axis of the detector.

Restricting the measured cosine of the angle of the thrust axis is the seventh event selection. The cosine of this angle should not be too close to 1 or -1 , that is to say, the products of the collision shouldn't be traveling close to the beamline axis. Specifically, we require $-0.80 < \cos \theta_{thrust} < 0.80$. As noted earlier, the SVT does not have one hundred percent full coverage of the region. This means particles near the axis may simply miss the detector components. This also further reduces Bhabha events where the electron and positron just barely scatter off of each other and have a very small scattering angle.

The final cut of the UofL HEP standard event selection removes events with three or four charged tracks where the highest momentum track is an electron. This again reduces Bhabha events by attempting to identify and remove events where one of the products radiates a photon, which pair-produces another electron-positron pair. Such events are called "radiative Bhabha" events.

CHAPTER IV

π^0 ANALYSIS

1 Background

The π^0 is a neutral meson, with a mass of $135 \text{ MeV}/c^2$, comprised of a quantum mixture of up and down quarks.

$$\pi^0: \frac{u\bar{u} - d\bar{d}}{\sqrt{2}},$$

where the notation here is indicative of a linear combination of an isospin doublet state[2][24]. With the quarks having the opposite charge of their anti-matter partner, it can be seen that the total charge for the particle is 0 in both the $u\bar{u}$ and $d\bar{d}$ states. For this study, the most common decay channel, $\pi^0 \rightarrow \gamma\gamma$, with branching fraction $98.823 \pm 0.034\%$ [24], was analyzed. Figure 14 shows the Feynman diagram for the $\pi^0 \rightarrow \gamma\gamma$ decay channel. We took advantage of *BABAR*'s excellent resolution of photons from the EMC in order to measure the production of π^0 in the off-peak energy region.

2 Cut Identification

In order to analyze the production rates of the π^0 from *BABAR*, a method was developed to systematically increase the purity of the signal. Using simulated data and software provided by the *BABAR* Collaboration, we recorded highly detailed collections of information about each simulated event and all of the particles in it. This data is recorded into what is called an nTuple. This is a file similar to a spreadsheet

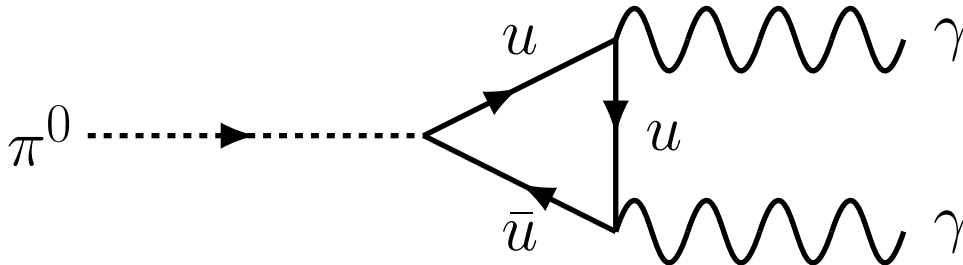


Figure 14. The Feynman diagram for $\pi^0 \rightarrow \gamma\gamma$, the particle also exists with down quarks in place of up quarks. Created by the author with the “feynman” L^AT_EX package from [17].

with deep connections between the properties recorded. These connections are akin to a tree-like structure, with branches making the relations between particles, events, and recorded properties with leaves representing the data values. In this file, we can measure many properties for each individual particle in an event. This could be charge, momentum, energy, etc., as well as information about the event as a whole. After simulation of the interactions in the event, an nTuple is produced that contains the data as if the detector was recording the information, and additionally some extra information, called “MC Truth,” which describes the initial state of the entities in the event. Using the data recorded in the file, an analysis script developed by the UofL HEP group was used to identify cuts to purify the signal. To determine which properties have a significant effect on reducing the background and identifying π^0 particles (or whichever particle you’re interested in), each property of the π^0 , the photons, and the event is analyzed. This analysis can determine whether a cut above, below, or both above and below identified values increased the signal squared to background ratio (S^2/B). Because we know the “truth” information we are able to compare prospective cuts to the true signal particles.

The determination of optimal cuts proceeds as follows. In this explanation, we

will use the mass of the π^0 candidate as an example of a variable on which we might try to cut. For each candidate π^0 particle, we record its reconstructed mass into one of two density distributions, depending on whether the MC truth information for the candidate tells us it is a π^0 . We thus end up with the number of correct π^0 as a function of π^0 mass in one distribution and the number of “fake π^0 ,” or “background,” candidates in the other. By integrating over a region of the π^0 distribution, we determine the number of signal candidates that will survive the corresponding cut, which we will call S. By integrating over the same region in the other distribution, we determine the number of background candidates that will survive the same cut, which we will call B. We automate the procedure to scan over a broad parameter space for possible cut values and cut types (“accept candidates with π^0 mass less than this value,” “accept candidates with π^0 mass greater than this value,” and “accept candidates with π^0 mass between these two values”). This is then repeated for other available measured quantities.

The resulting set of histograms provide a spectrum of cuts and allow the identification of the most effective cuts. I wrote a secondary script in order to calculate the value of the related property to cut on based on highest signal-squared-to-background ratio(S^2/B). A sample of 10 000 simulated events from *BABAR*’s database of simulated data was run while targeting the decay mode $\pi^0 \rightarrow \gamma\gamma$. The preceding process was used to calculate an increase in S^2/B for each property recorded in the nTuple file. Cuts that resulted in greater than a 5% increase in S^2/B were kept and applied in the full analysis. Table 1 lists the cuts that were implemented in the full analysis and Table 2 shows the percentage increase in S^2/B for each cut that was utilized. Figure 15 shows a comparison in the small sample set where no cuts are utilized and where all identified cuts are applied. Both sets of data are normalized to the same area for comparison purposes and displayed by using an analysis framework called ROOT. All subsequent figures in this dissertation were created during this study

Property	Cut Value
π^0 Mass Error	$0.0036525 < M_{\pi^0} Err < 0.0060225$
γ Energy	$E_\gamma > 0.138$ GeV
γ Momentum	$p_\gamma > 0.138$ GeV/ c
$\gamma \cos \theta$	$\cos \theta_\gamma < 0.9307$
γ Calorimeter Energy	$E_\gamma > 0.138$ GeV
γ Calorimeter Raw Energy	Raw $E_\gamma > 0.1305$ GeV
γ Number of Calorimeter Crystals	$\gamma n_{crystals} > 4.0425$
γ Lateral Moment	$\gamma_{lat} > 0.0125$
γ Zernike Polynomial	$ Z_4^2(\rho, \varphi) > 0.0002$

Table 1. Cuts identified for π^0

Property	Increase in S^2/B
π^0 Mass Error Cut Above	5.3%
π^0 Mass Error Cut Below	19.2%
γ Energy	19.6%
γ Momentum	19.6%
$\gamma \cos \theta$	5.7%
γ Calorimeter Energy	19.6%
γ Calorimeter Raw Energy	19.6%
γ Number of Calorimeter Crystals	17.7%
γ Lateral Moment	11.5%
γ Zernike Polynomial	5.1%

Table 2. Signal Increase per π^0 Cut

using the ROOT framework developed by [10]. Before applying cuts, the peak of the signal is approximately 1.7 times higher than the average background. After the cuts are applied, the signal peak is approximately 3 times the height of the average background; this is a relative improvement of approximately a factor of 2.

Once the optimal cuts are identified, I edited an analysis script developed by the *BABAR* Collaboration in order to program my cuts into the *BABAR* infrastructure of software. This software is advantageous over creating nTuples because the cuts are performed during runtime instead of in post-processing. The previously discussed software dumps large lists of properties for all event data into an nTuple file, whereas this software outputs simply a spectrum of candidates and records only rudimentary

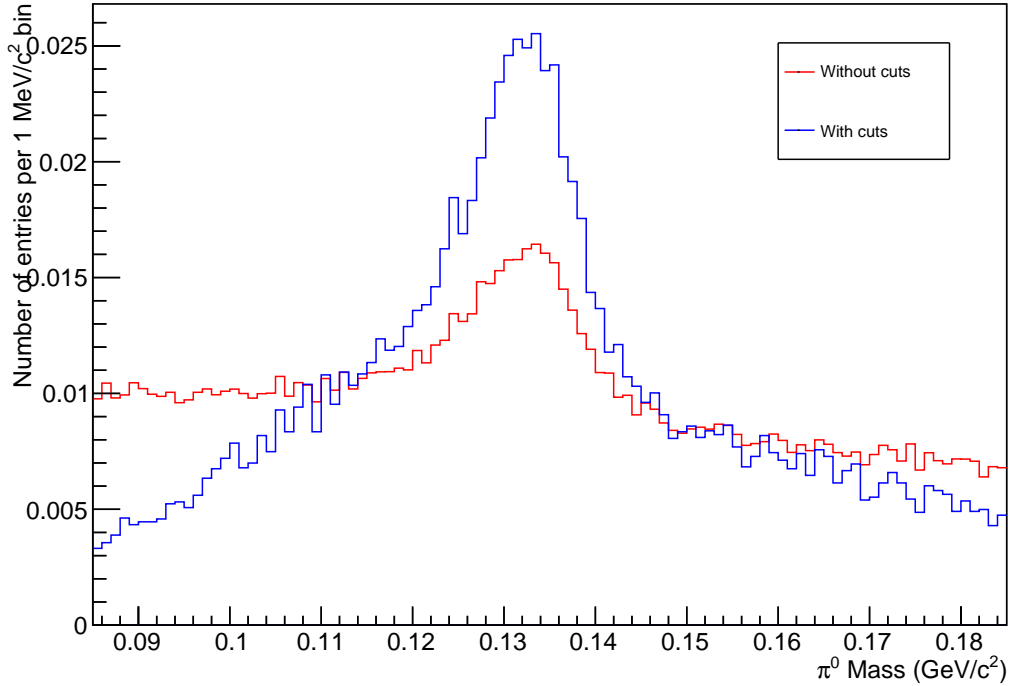


Figure 15. Comparison of the π^0 mass peak in 10 000 Monte Carlo events with and without cuts. Data is normalized to compare signal peak height.

properties of the candidates such as mass and momentum. For a Monte Carlo (MC) simulated analysis, truth information is also output.

After the cut analysis was finalized, large sets of MC and real data were run with the cuts applied. If conditions of any one event violate the specifications in the UofL HEP Group event selection, the event is discarded. Additionally, any individual candidate particle whose properties (or daughters' properties) violate one of the cuts listed in Table 1 is removed from the analysis as a candidate from events surviving event selection. We generated 8.64×10^7 MC events with a ratio of approximately 2 to 1 uds events to $c\bar{c}$ events, specifically 5.91×10^7 uds events and 2.73×10^7 $c\bar{c}$ events were used. The equivalent luminosity of the MC data was approximately 28.3 fb^{-1} for the uds events and 21 fb^{-1} for the $c\bar{c}$ events. For real data, 2.08×10^8 events from *BABAR*'s off peak Run 5 were used for the analysis. This is equivalent to 14.68

fb^{-1} of integrated luminosity at $10.54 \text{ GeV}/c^2$ CM energy. Upon completion of the processing, both sets of data were analyzed in parallel with the same method in order to account for inefficiencies of data-taking by the detector components.

3 π^0 Fitting

Both sets of data were divided into 10 $0.5 \text{ GeV}/c$ bins of π^0 candidate momentum and π^0 candidate mass spectrum was fit for signal and background contribution. Using a polynomial function for the background and a Gaussian,

$$f_G = \frac{A}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(x-\mu)^2/\sigma^2},$$

for the signal, fit parameters were obtained for each bin. Finding the area under the Gaussian function yields a measurement of the observed number of π^0 's in each momentum bin. For simulated data, the number of π^0 's that were generated in actuality, identified in the truth information, can be compared to the raw measurements made in each bin. This provides an effective efficiency that represents geometric detector efficiencies and branching fractions. The efficiency is calculated as

$$\text{MC efficiency} = \frac{N_{\text{observed}}}{N_{\text{truth}}},$$

where N_{observed} is the area under the Gaussian signal in the momentum bin and N_{truth} is the number of π^0 's recorded in the truth information. The value for each bin is stored so that it can be used to calculate the number of π^0 's from the real data events for each relative bin. Figure 16 shows the MC efficiency value for each momentum bin. Each momentum bin is plotted so that the value on the x -axis is equal to the central value of each bin.

For event data, again the area of the Gaussian signal function provided a raw measurement of π^0 's produced in each momentum bin. However, this time the efficiency

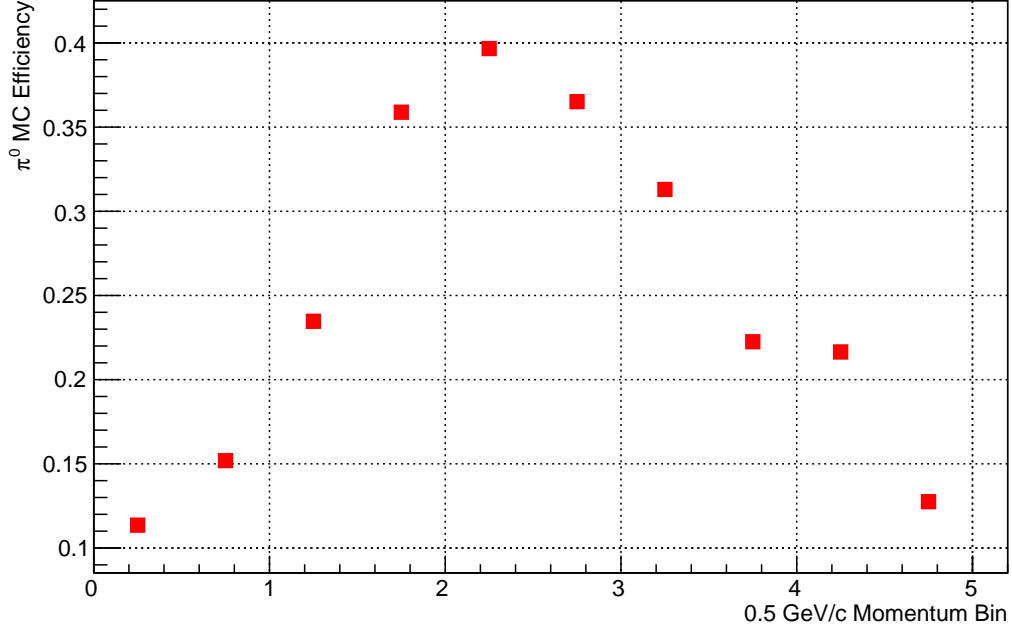


Figure 16. Detector efficiency of π^0 identification per 0.5 GeV/ c momentum bin. Each efficiency is plotted as the center value of the momentum bin.

value determined from the simulated data is used to project a measurement of the amount of π^0 's created during the events captured in the bin. A corrected number of π^0 's is calculated in each bin by dividing the area under the Gaussian signal fit by the bin's MC efficiency value. Figures 17 and 18 show the signal and background fits for each momentum bin for MC data and real data, respectively. Figure 19 shows the multiplicity of π^0 production having momentum within each momentum bin. Each momentum bin is plotted so that the value on the x -axis is equal to the central value of each bin. The corrected amount of π^0 's in each momentum bin is summed for all bins. The total number of π^0 's that were calculated is then divided by the number of events remaining in the analysis after cuts (4.02×10^7 events) to determine the multiplicity of the π^0 . I report that 2.8 π^0 's per event were created.

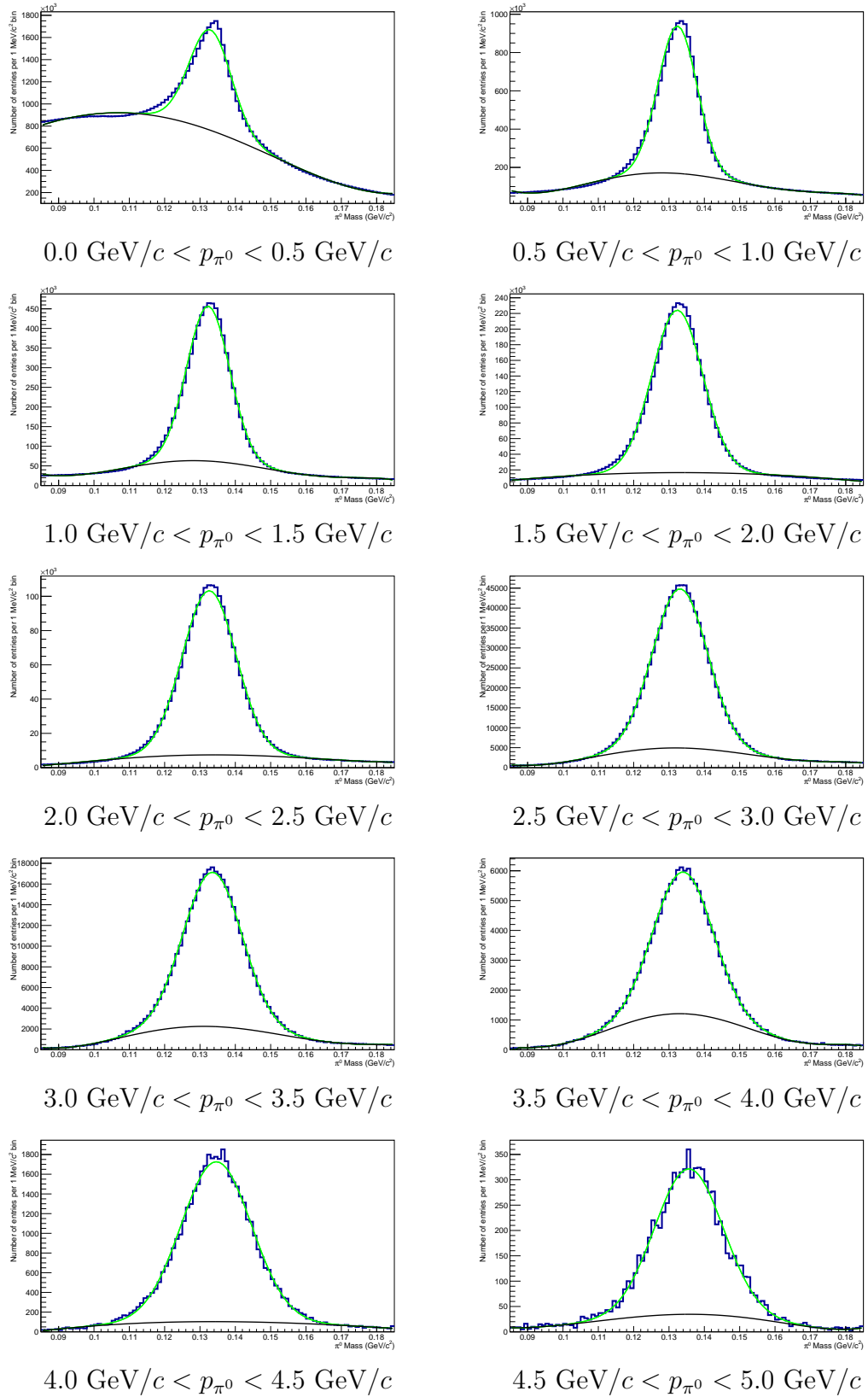


Figure 17

Histograms of π^0 mass from Monte Carlo data. Data is binned into ten bins of 0.5 GeV/c width each.

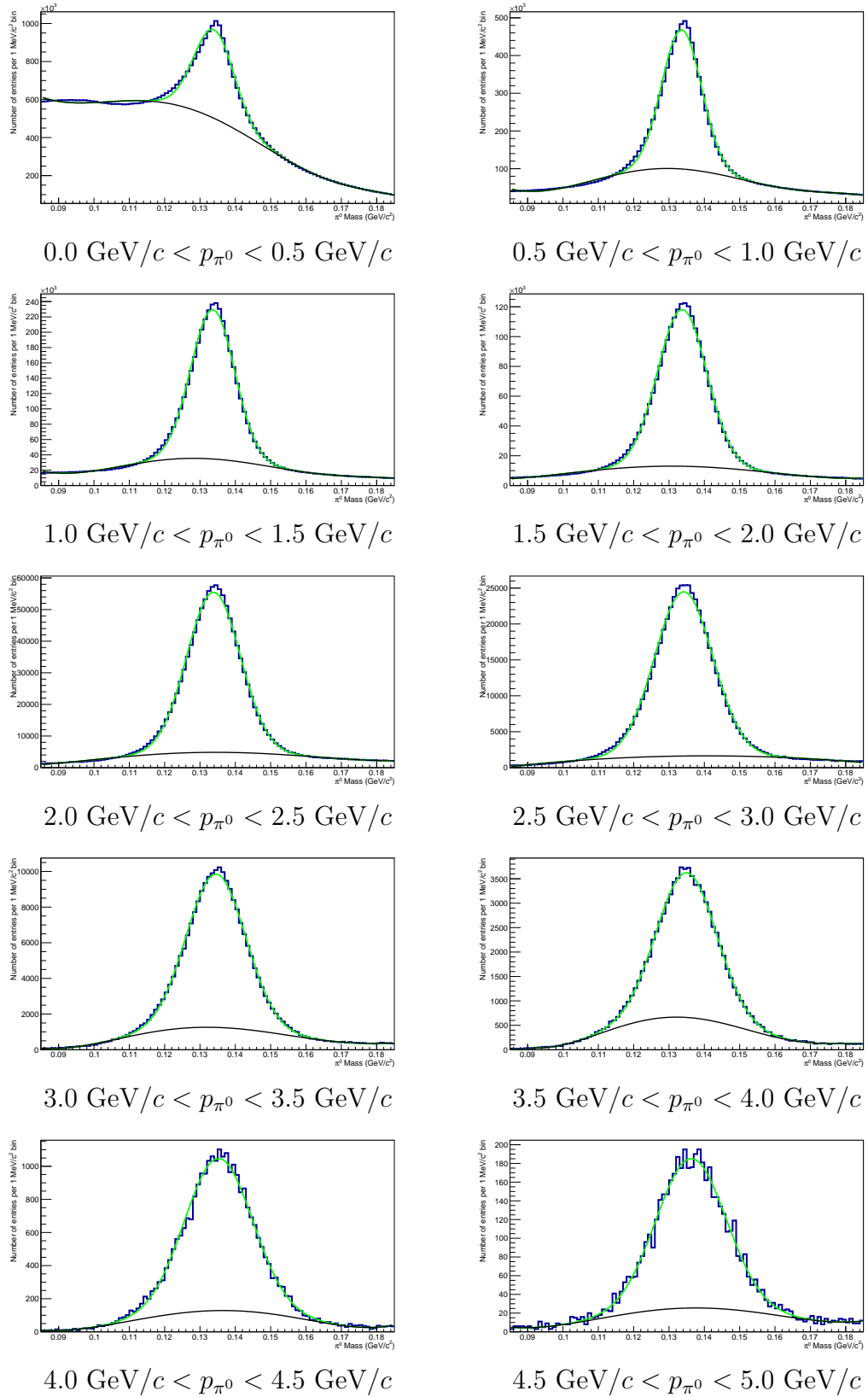


Figure 18

Histograms of π^0 mass from real data. Data is binned into ten bins of 0.5 GeV/c width each.

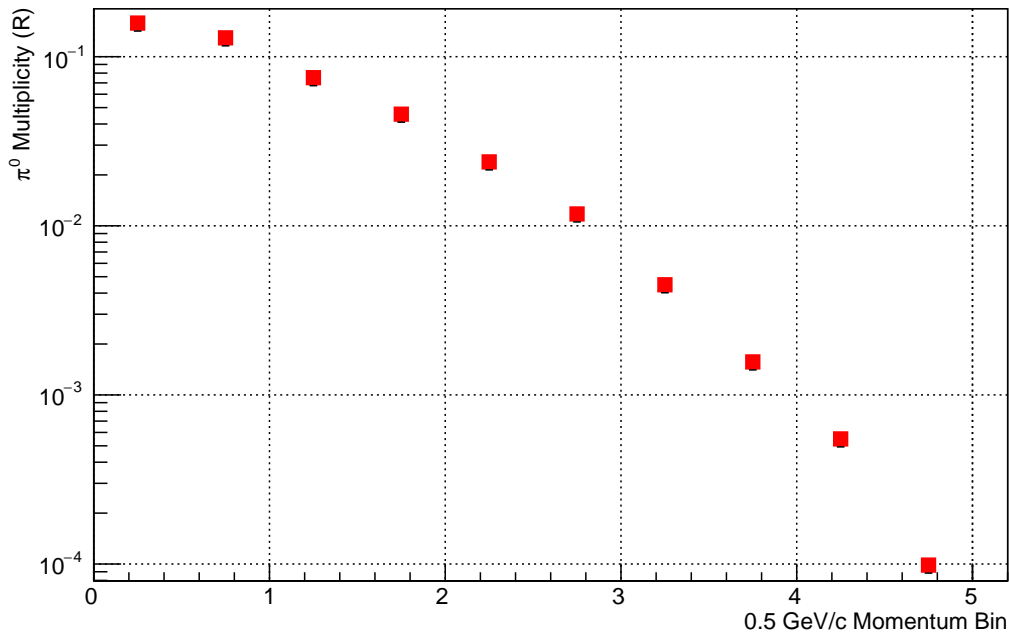


Figure 19. The multiplicity of π^0 's per 0.5 GeV/ c momentum bin. Each multiplicity is plotted as the center value of the momentum bin.

4 Uncertainty Analysis

To estimate the statistical uncertainty of our measurement, we assume a Poisson distribution for the production of the π^0 's, that is,

$$\sigma \approx \sqrt{N},$$

where N is the number of observed π^0 's. For this study, 1.8×10^7 π^0 's were found from the area of the Gaussian signal fit. This attributes to a Poisson uncertainty of about 4300 π^0 's, which is 0.02%.

There are three sources of systematic uncertainty probed in this study. Those uncertainties arise from the fitting process, detector efficiencies, and the candidate cuts.

For the complete data set, with candidate cuts applied, the Gaussian function

S^2/B	
Dataset	Value
No Cuts, No Efficiency Correction	653.5
With Cuts, No Efficiency Correction	718.2
No Cuts, With Efficiency Correction	20900
With Cuts, With Efficiency Correction	22820
Increase in S^2/B	
Dataset	Percentage increase
No Efficiency Corrections	9.912 %
With Efficiency Corrections	9.166 %

Table 3. Detector Efficiency Contribution to Uncertainty

that provides the best fit for data has a mean, μ , of 134 ± 1 MeV/ c^2 and a standard deviation, σ , of 8 ± 1 MeV/ c^2 . To understand the sensitivity of our measurements to the fitting process, we rerun our analysis, fixing the values of μ and σ in steps, with μ ranging between 133 and 135 MeV/ c^2 and σ ranging between 6.7 and 8.7 MeV/ c^2 . In no case do our results change by more than 0.4%, and we set this conservatively as the uncertainty due to fitting.

To determine the contribution of detector efficiency correction on the uncertainty, again the small set of data with and without candidate cuts was used. The Signal Squared to Background Ratio (S^2/B) was used as a proxy for our signal measurement and we examined how it was affected by bin-to-bin variation in efficiency correction. It was calculated for both sets and was also calculated with and without correcting with the MC efficiency. Table 3 shows the result of the S^2/B analysis. It can be seen in the table that the S^2/B is not significantly affected by the introduction of detector efficiency corrections. Using the small difference between the percentages of the increase in S^2/B , we assign a 7.5% uncertainty from detector efficiency correction.

For an analysis of the uncertainty due to the candidate cuts, all cuts besides the UofL HEP group standard event cuts were removed and new datasets were generated. In order to compare analyses with similar statistical relevance, 6 million data events

Source	Percentage	R
Statistical	0.02%	0.0006
π^0 Fitting	0.4%	0.01
Candidate Cuts	7.5%	0.2
Detector Efficiency	7.71%	0.2

Table 4
Absolute uncertainty on π^0 multiplicity.

with the cuts removed were processed and a corresponding set of 6 million data events were sampled from the full analysis. The truth statistics from the full analysis were used to correct for detector efficiency. The same fit procedure was utilized as the full run for both sets of data. The set with cuts removed resulted in a measurement of 3.77 π^0 's per event and the set including cuts was measured at 3.50 π^0 's per event. These values were used to calculate the deviation of the no cuts dataset from the candidate cut dataset. This was found to be 7.71%.

The sources of uncertainty were assumed independent and the contributions were added in quadrature. In total from the various sources of uncertainty, we found a variance of 10.3%, equating to 0.3 π^0 's per event. A summary of the uncertainties and their individual contributions is shown in Table 4.

CHAPTER V

η ANALYSIS

1 Background

The η particle is a neutral meson with similar quark content to the π^0 , but a mass of approximately $548 \text{ MeV}/c^2$ [24]. The difference is that the η 's composition includes a strange and anti-strange quark, as opposed to just up anti-up or down anti-down quarks.

$$\pi^0: \frac{u\bar{u} - d\bar{d}}{\sqrt{2}}$$

and

$$\eta: \frac{u\bar{u} + d\bar{d} - 2s\bar{s}}{\sqrt{6}}$$

These two particles also have the distinction that they are their own antiparticle complement.

Because of its similarity with π^0 , the η has similar decay channels. The dominant decay channel for the η is also the dominant decay channel for π^0 . Both the η and the π^0 decay into two photons as their primary decay mode, with the branching fraction for η for this decay being $39.41 \pm 0.20\%$ [24]. Since *BABAR* can't detect the neutral hadrons directly, we rely on the kinematics of their decay products and use combinatorics to attempt to identify the mothers of daughter particles. Through this strategy, photons from a π^0 could potentially be paired with a random photon and kinematically present as an η . This effect is of particular concern also because π^0 is created at *BABAR* at a much higher rate than η , meaning there are many more

photons from decaying π^0 's which could falsely reconstruct into an η . We developed a set of cuts to specifically suppress the background created by π^0 's.

2 η Cutting

We again follow a procedure similar to the π^0 analysis and start by performing the standard UofL HEP group event selection. After the event selection, the signal of $\eta \rightarrow \gamma\gamma$ is still hidden by background. We move to cutting out candidates by reducing the background swamping the signal of the η due to combinatorics of random pairings of photons produced by π^0 decay.

Each event remaining after the standard UofL HEP Group event selection is scanned for photons. *BABAR*'s datasets include measurements of photon energy and $\cos\theta$ in both the lab and center-of-mass frames. Inspired by a process described in [25], each photon that is identified as a daughter of a candidate η is scanned. The invariant mass is then calculated for all pairings with each other photon in the list of daughter photons. The invariant mass is defined as

$$M_{inv} = \sqrt{2p_1p_2(1 - \cos \theta)}$$

where θ is the angle between the two photons being considered, in the lab frame, and p is the momentum of each particle in the lab frame. If the invariant mass is consistent with the π^0 mass, $0.110 \text{ GeV}/c^2 < M_{inv} < 0.155 \text{ GeV}/c^2$, for any photon pair in the event, then the candidate η that was the parent of the photon being scanned is discarded.

In addition to the veto on accidental π^0 identification, a few cuts are made to ensure the MC matches real data. These requirements are

$$E_\gamma > 0.15 \text{ GeV}$$

$$LAT < 0.5$$

$$N_{crystals} \geq 4$$

$$|\cos \theta_{Helicity}| < 0.8,$$

where E_γ is the energy of any single photon, LAT describes the shape of the decay shower, $N_{crystals}$ is the number of crystals the photon passes through the SVT, and $\theta_{Helicity}$ is the angle between the candidate η in the CM frame and the direction of a daughter photon in the η rest frame.

3 η Fitting

After the cut identification process, another MC dataset was produced with the event selection and η candidate cuts applied. 1×10^8 events of MC data were run with approximately 2 to 1 uds to $c\bar{c}$ events, which is an equivalent luminosity of 29.5 fb^{-1} . For real data, the *BABAR* Run 5 dataset was run (2.08×10^8 events, 14.68 fb^{-1}) with event selection and candidate cuts applied. 4.03×10^7 events survived the event selection for real data

As can be seen in Figure 21, there is a small tail of excess particles on the low-momentum side of the peak. This is due to energy loss in the EMC from leakage through the crystals. Because of this feature, a Gaussian fit is not the most appropriate. We again refer to [25] to determine that a combination of a Novosibirsk function with a Gaussian function is the most appropriate fit for this curve. The Novosibirsk function is defined as

$$f(x) = e^{-\frac{(\ln q_N)^2}{2\tau^2} - \frac{\tau^2}{2}},$$

where

$$q_N = 1 + \frac{\sinh(\tau\sqrt{\ln 4})}{\sqrt{\ln 4}} \cdot \frac{x - x_0}{\sigma},$$

where τ is a fit parameter describing the shape of the curve and μ and σ serve

similar functions as in a Gaussian function, e.g. the center of the peak and the standard deviation, or RMS width. The Novosibirsk function does a good job at approximating the curve around its peak. To get a good fit on the tail of the signal a Gaussian function is added. For this analysis,

$$f_N(M_\eta) = \frac{1}{\sqrt{2\pi} \cdot \sigma} \exp \left[-\frac{\ln^2 \left(1 + \frac{\sinh(\tau\sqrt{\ln 4})}{\sqrt{\ln 4}} \cdot \frac{M_\eta - \mu}{\sigma} \right)}{2\tau^2} - \frac{\tau^2}{2} \right]$$

and

$$f_G(M_\eta) = \frac{1}{\sqrt{2\pi} \cdot \sigma} \exp \left[-\frac{1}{2} \left(\frac{M_\eta - \mu}{\sigma} \right)^2 \right]$$

so in total

$$f_{total} = N \cdot (1 - Q) \cdot f_N(M_\eta) + N \cdot Q \cdot f_G(M_\eta)$$

where N is the number of η 's and Q is a measure of the probability density relating the two functions such that total probability density equals 1. The μ and σ parameters for each constituent function are separate parameters that have individual contributions to the curves; they are denoted with a subscript N for the Novosibirsk parameters and a subscript G for the Gaussian. The Gaussian function is typically very shallow and is shifted left of the peak in order to fit the tail of the signal data in addition to the features of the tail of the Novosibirsk, which is described by the τ parameter. Figure 20 shows an example of the two separate functions before they are combined into a fit of the data. Figure 21 shows a fit of η 's found between CM momentum 2.5 GeV/ c and 3 GeV/ c with the background subtracted.

In low momentum bins, the background has a severe sloped shape. Due to this two different polynomial fits were performed on either side of the signal peak and then combined into a total background fit.

The same process is followed as the π^0 fitting by integrating the signal function and the compound background function and subtracted from each other. The width

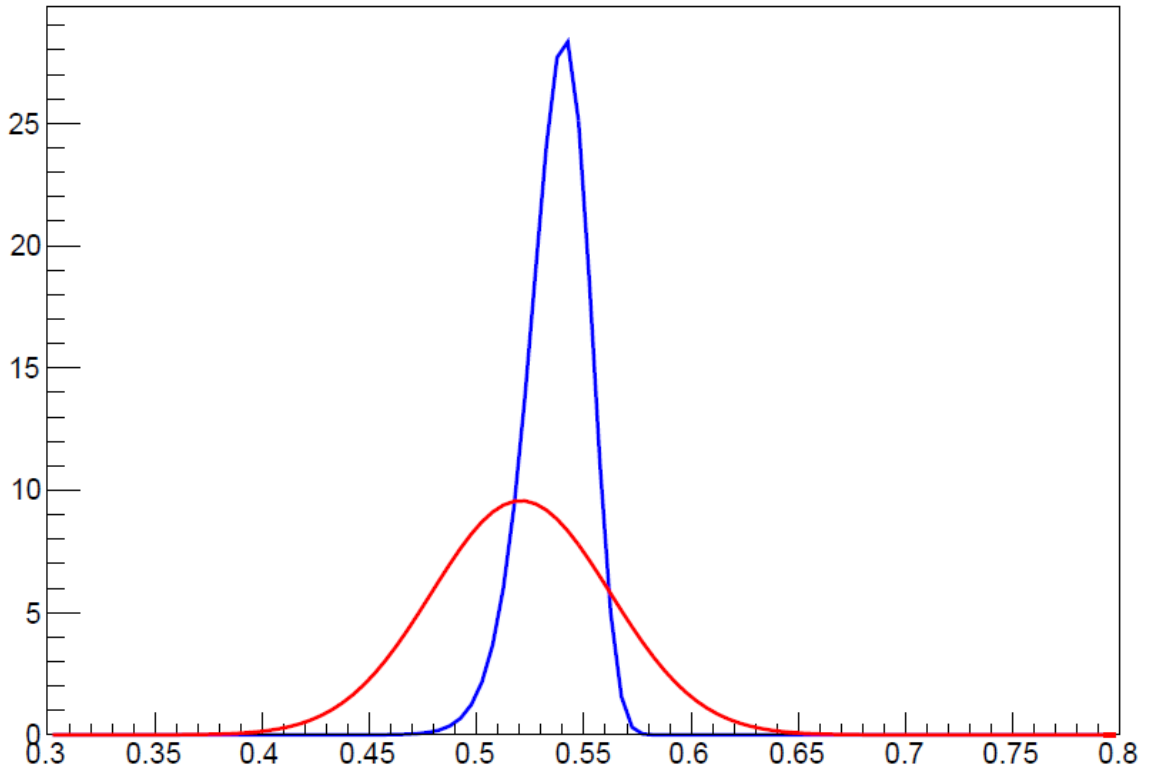


Figure 20. An example of the fit functions before they are combined.

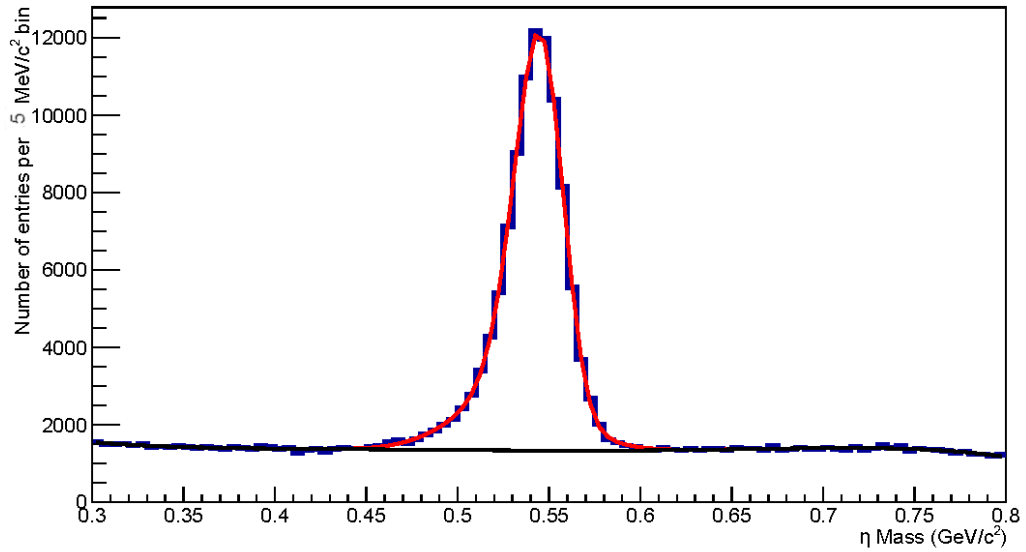


Figure 21. Background subtracted fit of η 's with $2.5 \text{ GeV}/c < p_{cm} < 3.0 \text{ GeV}/c$.

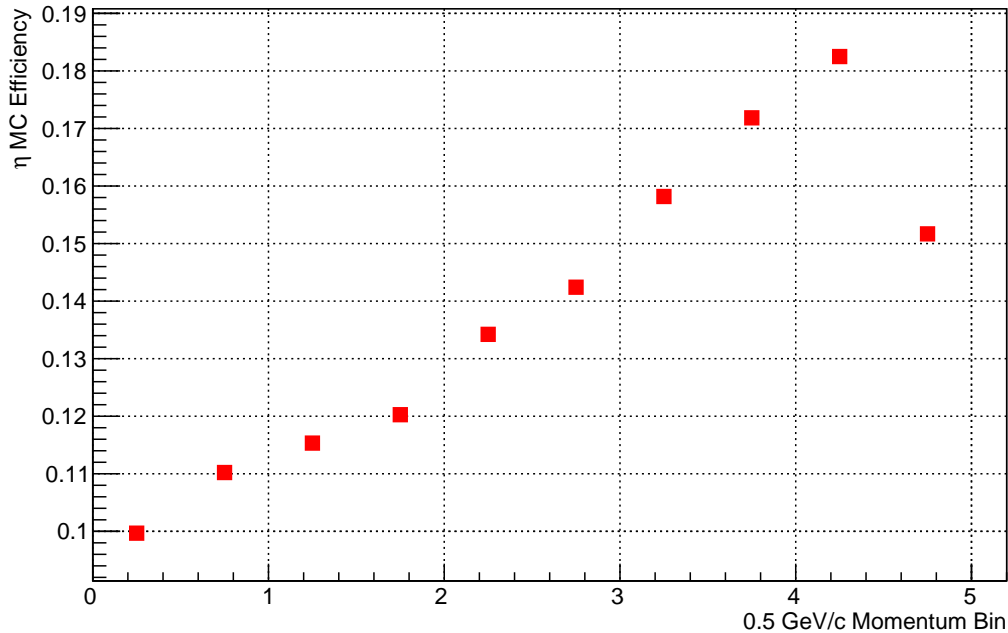


Figure 22. Detector efficiency of η identification per 0.5 GeV/ c momentum bin. Each efficiency is plotted as the center value of the momentum bin.

of each mass bin is then multiplied by the integral result to determine the number of η 's in the momentum bin. The detector efficiency for the decay is calculated from the MC fit and the truth statistics. Figure 22 shows the MC efficiency for each momentum bin. Each momentum bin is plotted so that the value on the x -axis is equal to the central value of each bin. After fitting each bin in the real data, the detector efficiency is used to calculate the amount of η 's found in the bin. Figures 24 and 25 show the signal and background fits for each momentum bin for MC data and real data, respectively. Figure 23 shows the multiplicity of η production having momentum within each momentum bin. Again, each momentum bin is plotted so that the value on the x -axis is equal to the central value of each bin. The number of η 's is totalled over all momentum bins and then divided by the number of events that survived cuts. We report that 0.25 η particles were found per event.

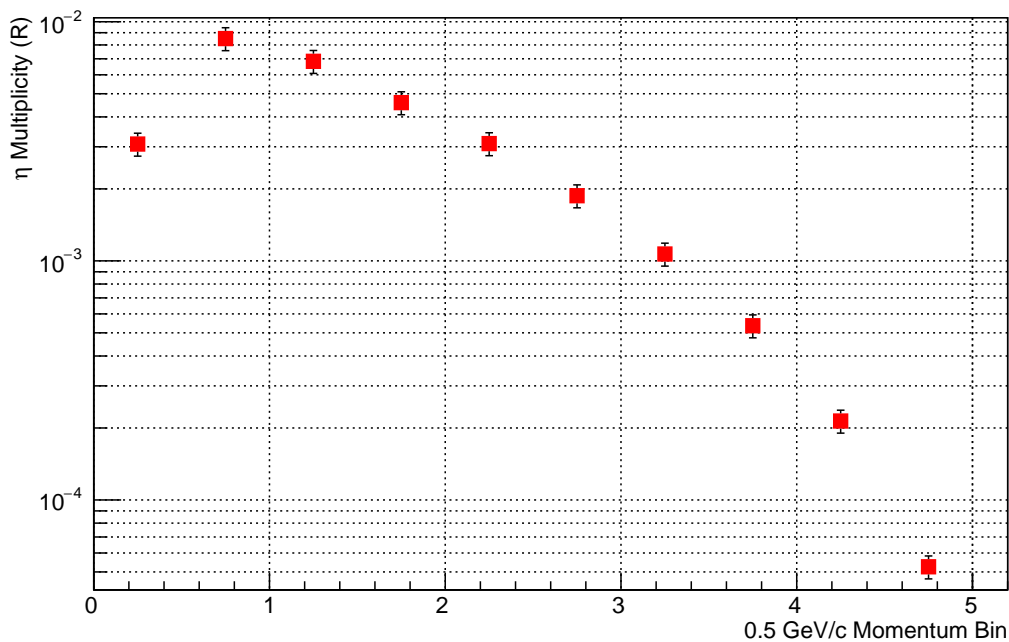


Figure 23. The multiplicity of η 's per 0.5 GeV/ c momentum bin. Each multiplicity is plotted as the center value of the momentum bin. Error bars correspond to statistical uncertainties only.

4 Uncertainty Analysis

To estimate uncertainty for the η , we again assume a Poisson distribution. With 1.2×10^6 raw η 's observed, we find a variance of approximately 1100 η 's per event. We assign a statistical uncertainty of 0.09% from this observation.

We were unable to directly calculate the uncertainty contributions from the candidate cuts and the detector efficiency for η , due in large part to the end of computing support at SLAC. Because the methodology was identical for applying the candidate cuts and because the detector efficiency should be very similar due to the η and π^0 decay modes studied, we prescribe the uncertainty found when studying π^0 for both systematic uncertainties. To be conservative, the higher of the two values is assigned for both uncertainties. The values for this are 7.71% for the candidate cuts and 7.71% for the detector efficiency.

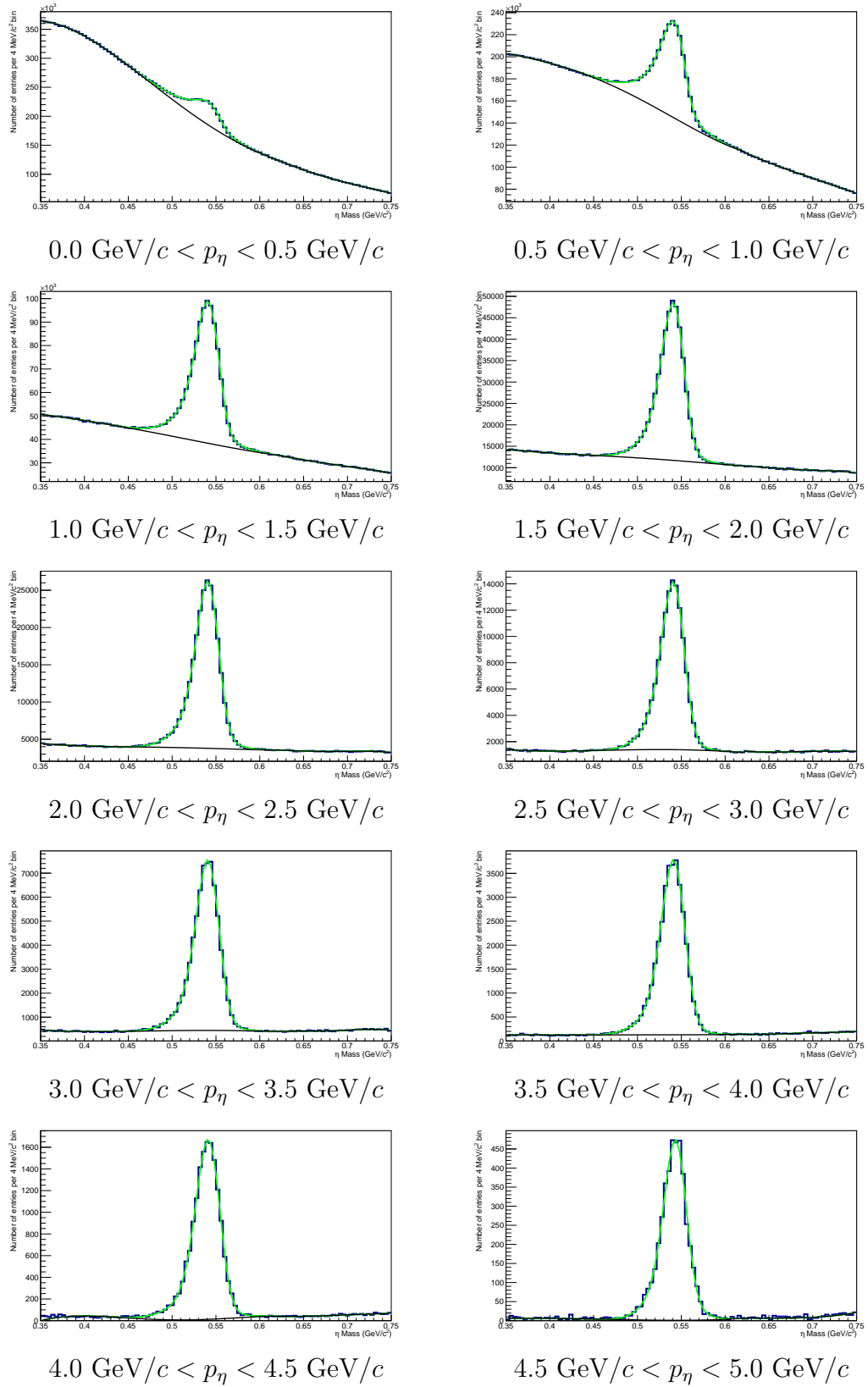


Figure 24

Histograms of η mass from Monte Carlo data. Data is binned into ten bins of 0.5 GeV/c width each.

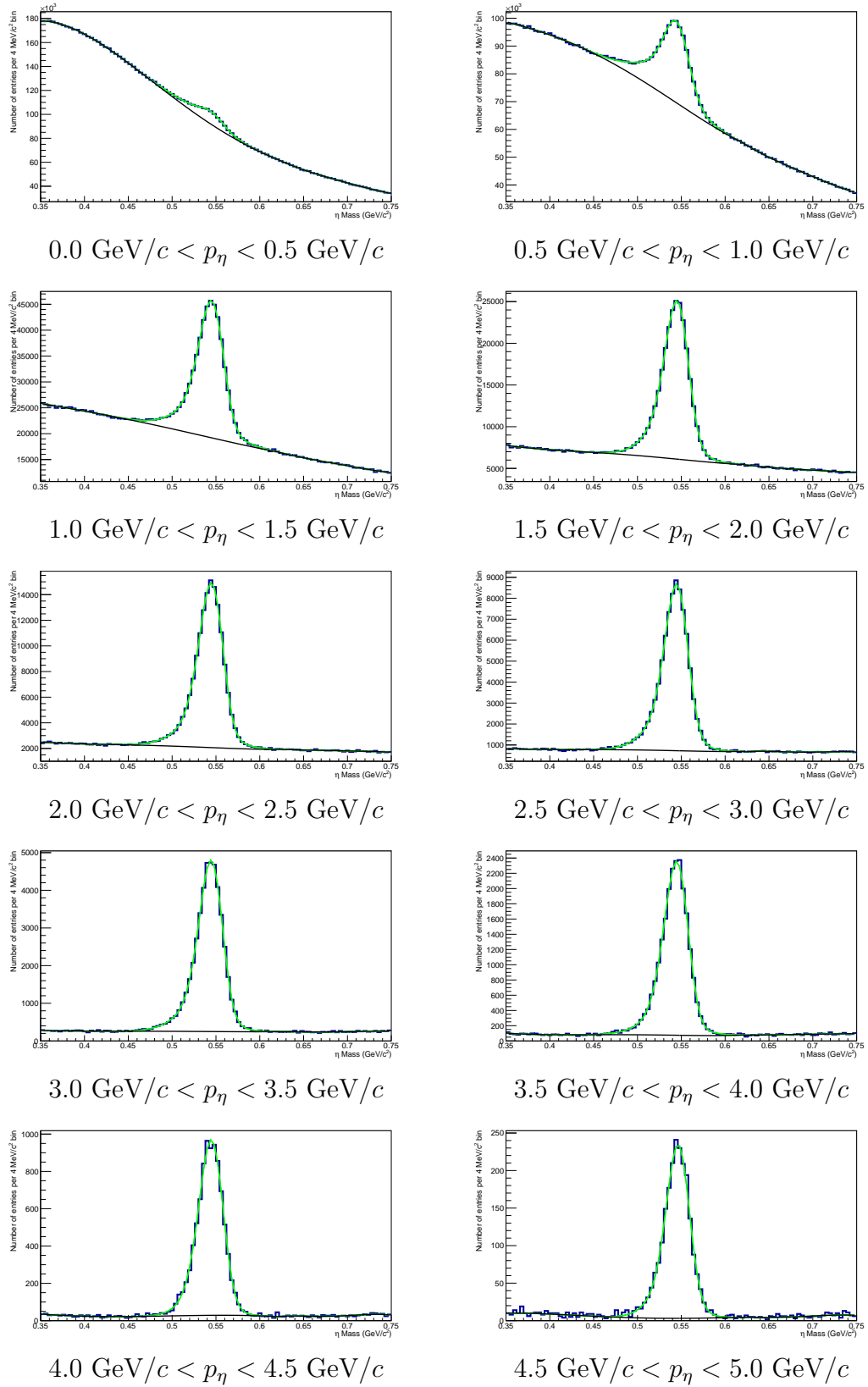


Figure 25

Histograms of η mass from real data. Data is binned into ten bins of 0.5 GeV/c width each.

Source	Percentage	R
Statistical	0.09%	0.0002
η Fitting	1.0%	0.003
MC Branching Fraction	0.5%	0.001
Candidate Cuts	7.71%	0.02
Detector Efficiency	7.71%	0.02

Table 5
Absolute uncertainty on η multiplicity.

A slightly different method was developed to estimate systematic uncertainties from the fitting of the η signal with the Novosibirsk and Gaussian combination. The data was fit again while varying the momentum bin size slightly. Measurements were taken for a multitude of momentum bin sizes. The data was fit again using the same process for all bins for each varying bin size. After obtaining sets of fit parameters, the standard deviation was calculated from across all measurements to determine the uncertainties of the properties of the analysis. Due to this analysis, we assign an uncertainty of 1.0%.

We account for a small variance of the branching fraction for $\eta \rightarrow \gamma\gamma$ used in generating the MC events. The value used to generate the MC events was approximately 0.5% off from the experimentally accepted value. We introduce an additional 0.5% uncertainty to our measurements due to this. A summary of the uncertainties and their individual contributions is shown in Table 5.

The sources of uncertainty were assumed independent and the contributions were added in quadrature. Taking into account these uncertainties, the total assigned uncertainty is 11.0%. This analysis resulted in measurements of $Q = 0.4 \pm 0.2$, $\tau_N = -7 \pm 2 \times 10^{-2}$, $\sigma_N = 13.3 \pm 0.4 \text{ MeV}/c^2$, $\sigma_G = 27 \pm 2 \text{ MeV}/c^2$, $\mu_N = 545.3 \pm 0.5 \text{ MeV}/c^2$, and $\mu_G = 532 \pm 3 \text{ MeV}/c^2$. This resulted in a total uncertainty of 0.03 η particles per event.

CHAPTER VI

CONCLUSIONS

Throughout this study, no consideration was taken to the parent particles or processes that created the π^0 's and η 's. This is what is meant by an “inclusive” study. We are not concerned with the specific processes creating the particles, but rather we are interested in the total amount of occurrence of the particle. In this study, we used the process $\pi^0 \rightarrow \gamma\gamma$ to determine that π^0 occurs 2.8 ± 0.3 times per event in e^+e^- annihilations at a center-of-mass energy of 10.54 GeV. It was also determined, by using the process $\eta \rightarrow \gamma\gamma$, that η occurs 0.25 ± 0.03 times per event in e^+e^- annihilations at a center-of-mass energy of 10.54 GeV.

The π^0 spectrum was fit with a polynomial background and a Gaussian signal. Table 6 summarizes the multiplicity result and fit parameters for the π^0 spectrum.

The η spectrum was fit with a split polynomial background and a compound signal function of a Novosibirsk function plus a shallow Gaussian function. Table 7 summarizes the multiplicity result and fit parameters for the η spectrum. The measurements for η agree with an independent measurement done in [25] by another *BABAR* group in 2001. Our measurement was done using data collected in a different detector run, which collected nearly four times the amount of integrated luminosity worth of data.

The measurements found in this study are available to theorists in order to test or build QCD-based models of hadronization. This is important because QCD is a difficult theory to do calculations for and is generally handled with elaborate models. Other experiments, particularly experiments at the LHC, depend on knowing the details QCD processes to accurately understand the backgrounds for their measurements. Our measurements will help accomplish this.

	Result
R	2.8 ± 0.3 per event
μ	134 ± 1 MeV/ c^2
σ	8 ± 1 MeV/ c^2

Table 6
Results for π^0 analysis

	Result
R	0.25 ± 0.03 per event
Q	0.4 ± 0.2
τ_N	$-7 \pm 2 \times 10^{-2}$
σ_N	13.3 ± 0.4 MeV/ c^2
σ_G	27 ± 2 MeV/ c^2
μ_N	545.3 ± 0.5 MeV/ c^2
μ_G	532 ± 3 MeV/ c^2

Table 7
Results for η analysis

REFERENCES

- [1] Users MissMJ and Cush - Own work by uploader, PBS NOVA, Fermilab, Office of Science, United States Department of Energy, Particle Data Group, Public Domain, <https://commons.wikimedia.org/w/index.php?curid=4286964> (2019).
- [2] D. Griffiths, *Introduction to Elementary Particles* (John Wiley & Sons, 1987), pp. 74, 168–170.
- [3] C. S. Wu, E. Ambler, R. W. Hayward, D. D. Hoppes, and R. P. Hudson, “Experimental Test of Parity Conservation in Beta Decay,” *Phys. Rev.* **105**, **1413** (1957).
- [4] J. P. Lees *et al.*, “Observation of time-reversal violation in the B^0 meson system,” *Phys. Rev. Lett.* **109**, 211801 (2012).
- [5] R. J. Abrams, D. Alexander *et al.*, “Mu2e conceptual design report,” (2012). ArXiv:1211.7019v1 [physics.ins-det].
- [6] G. Pezzullo, “Mu2e: A Search for Charged Lepton Flavor Violation in $\mu^- N \rightarrow e^- N$ conversion with a Sensitivity $< 10^{-16}$,” in “The 3rd International Conference on Charged Lepton Flavor Violation,” (2019).
- [7] A. Baldini, D. Glenzinski, F. Kapusta, Y. Kuno, M. Lancaster, J. Miller, S. Miscetti, T. Mori, A. Papa, A. Schoning, and Y. Uchida, “A submission to the 2020 update of the european strategy for particle physics on behalf of the comet, meg, mu2e and mu3e collaborations,” (2018). ArXiv:1812.06540v1 [hep-ex].
- [8] L. Bartoszek *et al.*, “Mu2e technical design report,” (2015).

- [9] A. Gaponenko, “CD3 Backgrounds,” (2017). Mu2e-docdb 7464 (Internal Document).
- [10] R. Brun and F. Rademakers, “Root - an object oriented data analysis framework,” Proceedings AIHENP’96 Workshop, Lausanne, Sep. 1996, Nucl. Inst. & Meth. in Phys. Res. A 389 (1997) 81-86. See also ”ROOT” [software], Release v5.34/25, 12/01/2015.
- [11] A. J. Bevan, B. Golob, T. Mannel, S. Prell, and B. D. Yabsley, “The Physics of the B Factories,” EPJC, SLAC-PUB-15968, KEK Preprint pp. 3–9 (2014).
- [12] S. L. Olsen, T. Skwarnicki, and D. Zieminska, “Nonstandard heavy mesons and baryons: Experimental evidence,” Reviews of Modern Physics **90** (2018).
- [13] S. Okubo, “ ϕ -meson and unitary symmetry model,” Physics Letters **5**, 165–168 (1963).
- [14] G. Zweig (1964). CERN Report No. 8419 / TH412 (Report).
- [15] J. Iizuka, “A Systematics and Phenomenology of Meson Family*,” Progress of Theoretical Physics Supplement **37-38**, 21–34 (1966).
- [16] J. Kroseberg, “Electroweak Penguin Decays of B Mesons,” in “DESY Particle and Astroparticle Physics Colloquium,” (2008).
- [17] A. Aivazis, “A package for drawing feynman diagrams in latex,” <https://feynman.aivazis.com/>.
- [18] D. Boutigny *et al.*, *The BABAR Physics Book: Physics at an Asymmetric B Factory* (Report of the BABAR Physics Workshop, 1998).
- [19] SLAC National Accelerator Laboratory, “Slac/lbl/llnl b factory design,” (2018).

- [20] R. Cowan, “The BABAR Detector,” (2008). https://www-public.slac.stanford.edu/babar/The_BaBar_Detector.htm.
- [21] D. Boutigny *et al.*, “BaBar technical design report,” InspireHEP (1995). SLAC-R-0457, SLAC-R-457, SLAC-0457, SLAC-457, SLAC-R-95-457.
- [22] Cornell Laboratory For Accelerator-based Sciences And Education, “The Upsilon System,” (2001). <https://www.classe.cornell.edu/public/lab-info/upsilon.html>.
- [23] C. Bernaciak, M. S. A. Buschmann, A. Butter, and T. Plehn, “Fox-wolfram moments in higgs physics,” *Physical Review D* **87** (2013).
- [24] P.A. Zyla *et al.* (Particle Data Group). *Prog. Theor. Exp. Phys.* **2020**, 083C01 (2020).
- [25] D. Altenburg and B. Spaan, “Measurement of inclusive η momentum spectra and multiplicities using the decay mode $\eta \rightarrow \gamma\gamma$,” (2003). *BABAR Analysis Document #250* (Internal Document).

APPENDIX A: COMMONLY USED ACRONYMS

CM - Center of Mass

CP - Charge and Parity

DCH - Drift Chamber

DIRC - Detector of Internally Reflected Cherenkov Radiation

EMC - Electromagnetic Calorimeter

HEP - High Energy Physics

HER - High Energy Ring

IFR - Instrumented Flux Return

LER - Low Energy Ring

LHC - Large Hadron Collider

MC - Monte Carlo

P - Parity

QCD - Quantum Chromodynamics

S^2/B - Signal-Squared to Background Ratio

SM - Standard Model

SVT - Silicon Vertex Tracker

UofL - University of Louisville

APPENDIX B: MODEL CODE

Example of histogram drawing and fitting code

```
#include <iostream>
#include <cmath>

#include "TROOT.h"
#include "TFile.h"
#include "TTree.h"
#include "TBrowser.h"
#include "TH2.h"
#include "TH1.h"
#include "TRandom.h"
#include "TPave.h"
#include "TPaveText.h"
#include "TPaveStats.h"
#include "TStyle.h"
#include "TVirtualPad.h"
#include "TLatex.h"

Double_t Background( Double_t *xPosition , Double_t
    *parameter )
{
    Double_t fitValue = 0.0;
    Double_t xValue = xPosition[0] - parameter[0];

    if ( xValue > 0 )
    {
        Double_t polynomial = parameter[1] * ( parameter[2] *
            xValue * xValue * xValue * xValue + parameter[3] *
            xValue * xValue * xValue + parameter[4] * xValue *
            xValue + parameter[5] * xValue + parameter[6] +
            parameter[7] * xValue * xValue * xValue * xValue *
            xValue + parameter[8] * xValue * xValue * xValue *
            xValue * xValue * xValue );

        fitValue = polynomial;
    }
}
```

```

    }

    return fitValue;
}

Double_t MultiBackground( Double_t *xPosition , Double_t
    *parameter )
{

    return Background( xPosition , parameter ) + Background(
        xPosition , &parameter [9] );

}

Double_t Gauss( Double_t *xPosition , Double_t *parameter )
{

    // parameter 0 = mean position
    // parameter 1 = standard deviation
    // parameter 2 = coefficient
    // This should return a normalized Gaussian value -
    // overall normalizing constant relegated
    // to the calling function
    Double_t xValue = xPosition [0] - parameter [0];

    Double_t gaussValue =
        parameter [2] / sqrt ( TMath :: TwoPi () ) / parameter [1] *
        TMath :: Exp( -( ( xValue * xValue )
            / 2.00
            /
            parameter [1]
            /
            parameter [1]
            ) ) );

    return gaussValue;
}

Double_t Novosibirsk( Double_t *xPosition , Double_t
    *parameter )
{

    // parameter 0 - mean position

```

```

// parameter 1 - "standard deviation" for Novosibirsk
// relative and absolute normalization relegated to
// calling function.

Double_t xValue = xPosition[0] - parameter[0];

Double_t hypSin = TMath::SinH(parameter[2] * 1.17741);

Double_t arg1 = 1.0 + (hypSin / 1.17741) * (xValue /
    parameter[1]);

Double_t epsilon = 0.000000001;

if(arg1 < 0.0) arg1 = epsilon;

Double_t arg2 = TMath::Log(arg1);

Double_t novoValue = 1.0/sqrt(TMath::TwoPi())/parameter[1]
    * TMath::Exp( - (arg2 * arg2) / (2.0 * parameter[2] *
    parameter[2]) - (parameter[2] * parameter[2]) / 2.0 );

    return novoValue;
}

Double_t NovoGauss(Double_t *xPosition, Double_t *parameter)
{

    //parameter 0 - overall normalization
    //parameter 1 - relative normalization of Gaussian

    return parameter[0] * ((1.0 - parameter[1]) * Novosibirsk(
        xPosition, &parameter[2] ) + parameter[1] * Gauss(
        xPosition, &parameter[5]));
}

Double_t fitFunction( Double_t *xPosition, Double_t
    *parameter )
{
    return MultiBackground( xPosition, parameter ) + Gauss(
        xPosition, &parameter[18] );
}

void Pi0BaBarSoftHistFitMeasureTuple()
{

```



```

TFile *MC = new
  TFile("Datafiles/FullRuns/MC/Pi0Anal_Run5-MCTuple.root");
TFile *data = new
  TFile("Datafiles/FullRuns/data/Pi0Anal_Run5-DataTuple.root");

TTree *MCTree1 = (TTree*)MC->Get("ntp101");
TTree *MCTree2 = (TTree*)MC->Get("ntp102");

TTree *dataTree = (TTree*)data->Get("ntp102");

Double_t MCMass[100];
Double_t trueMCp3[100];
Double_t trueMCcosth[100];
Double_t trueMCp3CM[100];
Double_t MCp3[100];
Double_t MCcosth[100];
Double_t MCP3CM[100];
Double_t MCcoshel[100];
Double_t dataMass[100];
Double_t datap3[100];
Double_t datacosth[100];
Double_t datap3CM[100];
Double_t datacoshel[100];

MCTree1->SetBranchAddresses("p3", trueMCp3);
MCTree1->SetBranchAddresses("costh", trueMCcosth);
MCTree1->SetBranchAddresses("p3CM", trueMCp3CM);
MCTree2->SetBranchAddresses("mass", MCMass);
MCTree2->SetBranchAddresses("p3", MCp3);
MCTree2->SetBranchAddresses("costh", MCcosth);
MCTree2->SetBranchAddresses("p3CM", MCP3CM);
MCTree2->SetBranchAddresses("coshel", MCcoshel);
dataTree->SetBranchAddresses("mass", dataMass);
dataTree->SetBranchAddresses("p3", datap3);
dataTree->SetBranchAddresses("costh", datacosth);
dataTree->SetBranchAddresses("p3CM", datap3CM);
dataTree->SetBranchAddresses("coshel", datacoshel);

// cout << "Opened the input file." << endl;

TF1 *function = new TF1("fitFunction", fitFunction, 0.085,
  0.185, 21);

```

```

Int_t iHist = 0;

TCanvas *canv = new TCanvas("canv","The_fit_
  canvas",1800,1000);

Int_t iter = 0;
Int_t iMom = 0;

TString runName = "_FullRun_063020_1";

Double_t binWidth = 0.50; //set the bin width, please
  set binWidthStr on next line
TString binWidthStr = "";
binWidthStr = "0.5GeV";

Double_t pmin = 0.0;
Double_t pmax = 5.0;

TString dataset = "";
TString selection = "";
TString trueSel = "";
TString myname = "";
TString projection = "";

TString BGFit = "";

Double_t yint = 1.0;
Double_t yint2 = 1.0;

Double_t funcInt = 0;
Double_t bffInt = 0;
Double_t numPi0s = 0;

Double_t totalPi0s = 0;
Double_t correctedPi0s = 0;
TString Pi0sString = "";

TString saveName = "";
TString fitName = "";

TString bffFile = "";
TString ngffFile = "";
TString nffFile = "";
TString gffFile = "";

```

```

Int_t trueNum = 0;

Double_t bgfs [21];
Double_t parameter [21];
Double_t p18 [20];
Double_t p19 [20];
Double_t p20 [20];

Double_t MCEff = 0.0;
Double_t eff [20];

Double_t pSpectrumTruthX [10];
Double_t pSpectrumTruthY [10];
Double_t pSpectrumX [10];
Double_t pSpectrumY [10];

TFile *output = new TFile("SavedFiles/outputHists.root",
"RECREATE");
TFile::Open("outputHists.root", "RECREATE");

for ( iter = 0; iter < 2; iter++)
{
    if(iter == 0)
        {dataset = "MC";}
    else
        {dataset = "data";}

for ( iMom = 0; iMom < 10; iMom++)
{

    //set the binning
    if(iMom == 0)
    {
        pmin = 0;
        pmax = binWidth;

        //initialize the signal parameters to set up
        cascade "guesses"
        parameter [18] = 0.1335 ;
        parameter [19] = 0.005;
        parameter [20] = 2000.0;

    }
}

```

```

else
{
    pmin = ((double)iMom)*binWidth;
    pmax = pmin + binWidth;
}
//if you change the momentum binning, make sure to
change the file names!

//set the momentum bin for the histogram
selection = "p3CM>";
selection += pmin;
selection += "&&";
selection += "p3CM<";
selection += pmax;

trueSel = "";
trueSel += selection;

selection += "&&";
selection += "coshel <0.8";
selection += "&&";
selection += "coshel >-0.8";

myname = "my" + dataset + "Hist";
myname += iMom;

projection = "mass>>";
projection += myname;

//create the temporary histogram for this loop,
//set the title and the "data" name
//set number of bins and x-axis width
TH1D *myHist = new TH1D(myname, selection, 100,
    0.085, 0.185 );

//initialize the fit function onto the data
if (iter==0)
    {MCTree2->Draw( projection, selection );}
else
    {dataTree->Draw( projection, selection );}

//myHist is now the mass histogram for this
selection

```

```

myHist->Draw();

//set various display options for histogram
myHist->SetLineWidth(3);
myHist->SetStats(true);
gStyle->SetOptFit(0002);
gStyle->SetStatW(0.25);
gStyle->SetStatH(0.5);
gStyle->SetTextSize(0.75);
myHist->SetTitle(selection);
myHist->GetXaxis()->SetTitle("#pi^{0} Mass
    (GeV/c^{2})");
myHist->GetYaxis()->SetTitle("Number of entries
    per 10 MeV/c^{2} bin");
myHist->GetYaxis()->SetTitleOffset(1.1);

TF1 *bg1 = new TF1("left", Background, 0.085,
    0.105, 9);
TF1 *bg2 = new TF1("right", Background, 0.16,
    0.185, 9);

bg1->SetParameter(0, 0.085);    // Offset parameter
    - put in middle.
bg1->FixParameter(0, 0.085);    // Why change it?
bg1->SetParameter(1, 1.0);    // Overall norm.
    Making zero would kill
bg1->SetParameter(2, 0.0);    // 4th order term
bg1->FixParameter(2, 0.0);    // No 4th order
    contribution
bg1->SetParameter(3, 0.0);    // 3rd order term
bg1->FixParameter(3, 0.0);
bg1->SetParameter(4, 0.0);    // 2nd order term
bg1->SetParameter(5, -1.0);    // Linear term

yint = myHist->GetBinContent(1);

bg1->SetParameter(6, yint);
bg1->FixParameter(6, yint);
bg1->SetParameter(7, 0.0);    // These are the
    highest order terms and
bg1->FixParameter(7, 0.0);    // are fixed to zero -
    limiting us to
bg1->SetParameter(8, 0.0);    // a 4th or 3rd order
    poly for bg fit
bg1->FixParameter(8, 0.0);

```

```

bg2->SetParameter(0, 0.16);
bg2->FixParameter(0, 0.16);
bg2->SetParameter(1, 1.0);
bg2->SetParameter(2, 0.0);
bg2->FixParameter(2, 0.0);
bg2->SetParameter(3, 0.0);
bg2->FixParameter(3, 0.0);
bg2->SetParameter(4, 0.0);
bg2->SetParameter(5, -1.0);

yint2 = myHist->GetBinContent(65);

bg2->SetParameter(6, yint2);
bg2->FixParameter(6, yint2);
bg2->SetParameter(7, 0.0);
bg2->FixParameter(7, 0.0);
bg2->SetParameter(8, 0.0);
bg2->FixParameter(8, 0.0);

myHist->Fit(bg1, "R");
bg1->GetParameters(&bgfs[0]);

myHist->Fit(bg2, "R+");
bg2->GetParameters(&bgfs[9]);

bg1->Draw("same");
bg2->Draw("same");

function->SetParameters(bgfs);
function->SetLineColor(3);

function->Draw("same");

BGFit = "SavedFiles/" + myname + runName + ".pdf";
canv->SaveAs(BGFit);

//      function->FixParameter(0, bgfs[0]);
//      function->FixParameter(1, bgfs[1]);
//      function->FixParameter(2, bgfs[2]);
//      function->FixParameter(3, bgfs[3]);
//      function->FixParameter(4, bgfs[4]);
//      function->FixParameter(5, bgfs[5]);

```

```

//      function->FixParameter(6, bgfs [6]);
//      function->FixParameter(7, bgfs [7]);
//      function->FixParameter(8, bgfs [8]);
//      function->FixParameter(9, bgfs [9]);
//      function->FixParameter(10, bgfs [10]);
//      function->FixParameter(11, bgfs [11]);
//      function->FixParameter(12, bgfs [12]);
//      function->FixParameter(13, bgfs [13]);
//      function->FixParameter(14, bgfs [14]);
//      function->FixParameter(15, bgfs [15]);
//      function->FixParameter(16, bgfs [16]);
//      function->FixParameter(17, bgfs [17]);

// Optional 2nd Background suppressing
function->FixParameter(9, 0);
function->FixParameter(10, 0);
function->FixParameter(11, 0);
function->FixParameter(12, 0);
function->FixParameter(13, 0);
function->FixParameter(14, 0);
function->FixParameter(15, 0);
function->FixParameter(16, 0);
function->FixParameter(17, 0);

//set initial conditions
//Background function
//      function->SetParameter( 0, 0.085 );
//      function->SetParameter( 1, 1.0 );
//coefficient
//      function->SetParameter( 2, 1.0 ); //x^4
//      function->SetParameter( 3, 1.0 ); //x^3
//      function->SetParameter( 4, 1.0 ); //x^2
//      function->SetParameter( 5, 1.0 ); //x
//      function->SetParameter( 6, 1000.0 ); //C
//function->SetParLimits( 6, 0.0, 10000000.0 );
//      function->FixParameter( 7, 0.0 );
//      function->FixParameter( 8, 0.0 );

//re-initialize p23 and p24 if they are out of
//      bounds from last pbin
if(parameter[18] < 0.13 || parameter[18] > 0.138)
    {parameter[18] = 0.135;}
if(parameter[19] < 0.005 || parameter[19] > 0.008)

```

```

    {parameter[19] = 0.006;}

//Gaussian function
function->FixParameter( 18, parameter[18] );
    //peak_g
//function->SetParLimits(23, 0.525, 0.54);
function->SetParLimits( 18, 0.132, 0.1355 );
function->FixParameter( 19, parameter[19] );
    //sigma_g
//function->SetParLimits(24, 0.023, 0.029);
function->SetParLimits( 19, 0.005, 0.01 );
function->FixParameter( 20, parameter[20] );
    //coefficient

cout << "Made_fit_function_and_parameter." << endl;

//create background function for display purposes
TF1 *bff = new TF1( "bff", MultiBackground, 0.085,
    0.185, 18);

TF1 *gff = new TF1( "gff", Gauss, 0.085, 0.185, 3);

//systematically attempt a function fit and
    release parameters
//    function->FixParameter(16, 1.0);

//    function->ReleaseParameter(25);

myHist->Fit(function, "", "", 0.085, 0.185);

function->ReleaseParameter(20);
function->SetParLimits( 20, 0.0, 500000.0);

myHist->Fit(function, "", "", 0.085, 0.185);

function->ReleaseParameter(18);
function->SetParLimits(18, 0.1325, 0.1355);

myHist->Fit(function, "", "", 0.085, 0.185);

function->ReleaseParameter(19);
//function->SetParLimits(24, 0.024, 0.026);

```



```

function -> SetParLimits(19, 0.005, 0.009);

myHist -> Fit(function, "", "", 0.085, 0.185);

function -> ReleaseParameter(0);
function -> ReleaseParameter(1);
function -> ReleaseParameter(2);
function -> ReleaseParameter(3);
function -> ReleaseParameter(4);
function -> ReleaseParameter(5);
function -> ReleaseParameter(6);
function -> ReleaseParameter(7);
function -> ReleaseParameter(8);
//function -> ReleaseParameter(9);
//function -> ReleaseParameter(10);
//function -> ReleaseParameter(11);
//function -> ReleaseParameter(12);
//function -> ReleaseParameter(13);
//function -> ReleaseParameter(14);
//function -> ReleaseParameter(15);
//function -> ReleaseParameter(16);
//function -> ReleaseParameter(17);

myHist -> Fit(function, "", "", 0.085, 0.185);

function -> ReleaseParameter(18);
function -> ReleaseParameter(19);
function -> ReleaseParameter(20);

myHist -> Fit(function, "", "", 0.085, 0.185);

//get final function parameters and set background
//fit parameters

function -> GetParameters( parameter );

if( iter == 0)
{
    p18[iMom] = parameter[18];
    p19[iMom] = parameter[19];
    p20[iMom] = parameter[20];
}
else

```

```

{
    p18[iMom + 10] = parameter[18];
    p19[iMom + 10] = parameter[19];
    p20[iMom + 10] = parameter[20];

    cout << "Parameter_18,MC:_ " << p18[iMom] << " _
    |_Data:_ " << parameter[18] << endl;
    cout << "Delta:_ " << p18[iMom] -
    parameter[18] << " |_%Change:_ " <<
    (parameter[18] - p18[iMom]) / p18[iMom] <<
    endl;
    cout << "Parameter_19,MC:_ " << p19[iMom] << " _
    |_Data:_ " << parameter[19] << endl;
    cout << "Delta:_ " << p19[iMom] -
    parameter[19] << " |_%Change:_ " <<
    (parameter[19] - p19[iMom]) / p19[iMom] <<
    endl;
    cout << "Parameter_20,MC:_ " << p20[iMom] << " _
    |_Data:_ " << parameter[20] << endl;
    cout << "Delta:_ " << p20[iMom] -
    parameter[20] << " |_%Change:_ " <<
    (parameter[20] - p20[iMom]) / p20[iMom] <<
    endl;

}

bff->SetParameters( parameter );
bff->SetLineColor(1);
bff->Draw("same");

canv->Update();

gff->SetParameter(0, parameter[18] );
gff->SetParameter(1, parameter[19] );
gff->SetParameter(2, parameter[20] );

//calculate number of found Pi0s in bin and add to
running total
funcInt = function->Integral(0.085, 0.185);

bffInt = bff->Integral(0.085, 0.185);

numPi0s = (funcInt - bffInt) * 1000;

Pi0sString = "Number_of_Pi0s:_ ";

```

```

Pi0sString += numPi0s;

saveName = "SavedFiles/JakePi0Analysis" +
    binWidthStr;

//Number in GeVs represents width of binned
//momentum
//Shift is amount in GeV the first bin is offset
//from 0

saveName += dataset;

TCanvas *fitCanv = new TCanvas("fitCanv", "Fit_
    Function_Canvas", 1800, 1000);

fitName = "";
fitName += saveName;
fitName += "_FitFunctions_pbin";
fitName += iMom;
fitName += runName;

bff->Draw();
bffFile = "";
bffFile += fitName;
bffFile += "_background.pdf";

fitCanv->SaveAs(bffFile);

gff->Draw();
gffFile = "";
gffFile += fitName;
gffFile += "_gaussian.pdf";

fitCanv->SaveAs(gffFile);

fitCanv->Close();

//get "the answer" of actual Pi0s in MC
if(iter == 0)
{
    TCanvas *trueCanv = new
        TCanvas("trueCanv", "True_Momentum_

```

```

Canvas",1800,1000);

TH1F *trueHist = new TH1F("trueHist","Number_
of_Pi0's_in_pBin", 10, 0, 10);

MCTree1->Draw("p3CM>>trueHist", trueSel);

trueHist->Draw();

trueNum = trueHist->GetEntries();

pSpectrumTruthX[iMom] = (pmax + pmin) / 2;
pSpectrumTruthY[iMom] = trueNum;

delete trueHist;
trueHist = 0;

trueCanv->Close();

canv->cd();

//calculate the rate at which Pi0s are
  identified in the MC
if(trueNum == 0)
  {MCeff = 0;}
else if(numPi0s == 0)
  {MCeff = 0.000001 / trueNum;}
else
  {MCeff = numPi0s / trueNum;}

//set the efficiencies for the bins in the Data
eff[iMom] = MCeff;

Pi0sString += "  Efficiency: ";

Pi0sString += MCeff;

TPaveText *effText = new
  TPaveText(0,0,.4,.05,"NDC");

effText->AddText(Pi0sString);
effText->Draw();

```

```

saveName += "_pBin";

saveName += iMom;
saveName += runName;
saveName += ".pdf";

}
else
{

    //use calculated MC efficiency to predict
    //actual number of Pi0s in Data
    correctedPi0s = numPi0s/eff[iMom];

    pSpectrumX[iMom] = (pmax + pmin) / 2;
    pSpectrumY[iMom] = correctedPi0s;

    Pi0sString += "_Corrected:_";
    Pi0sString += correctedPi0s;

    TPaveText *numText = new
        TPaveText(0,0,.4,.05,"NDC");

    numText->AddText(Pi0sString);
    numText->Draw();

    saveName += "_pBin";

    saveName += iMom;
    saveName += runName;
    saveName += ".pdf";
    //count up total number of corrected Pi0s
    if(correctedPi0s < 0)
    {
        totalPi0s = totalPi0s + 0;
    }
    else
    {
        totalPi0s = totalPi0s + correctedPi0s;
    }
}

```

```

        myHist->Write(myname);
        canv->SaveAs(saveName);

        delete myHist;
        myHist = 0;

        canv->Clear();
    }
}

```

Example of *BABAR* run time cutting code

```

// *****
// JB Cuts for Pi0
// *****

cout << "Running JB_Cuts...\n";

bool failJBCuts = false;
bool failMomMass = false;
bool failMomMassErr = false;
bool faild1Costh = false;
bool faild2Costh = false;
bool faild1ECal = false;
bool faild2ECal = false;
bool faild1ErawCal = false;
bool faild2ErawCal = false;
bool faild1nCrys = false;
bool faild2nCrys = false;
bool faild1Lat = false;
bool faild2Lat = false;
bool faild1A42 = false;
bool faild2A42 = false;

//      if ( reco->mass() < 0.11104 || reco->mass() >
//          0.14672 ) failMomMass = true;
if ( reco->massErr() < 0.0036525 || reco->massErr() >
    0.0060225 ) failMomMassErr = true;

if ( cos(daug1->p4().theta()) > 0.9307 ) faild1Costh =
    true;
if ( cos(daug2->p4().theta()) > 0.9307 ) faild2Costh =
    true;

```

```

const BtaCalQual* d1CalQual =
    daug1->getMicroAdapter()->getCalQual();
const BtaCalQual* d2CalQual =
    daug2->getMicroAdapter()->getCalQual();

if ( d1CalQual->ecalEnergy() < 0.138 ) faild1ECal = true;
if ( d2CalQual->ecalEnergy() < 0.138 ) faild2ECal = true;
if ( d1CalQual->rawEnergy() < 0.1305 ) faild1ErawCal =
    true;
if ( d2CalQual->rawEnergy() < 0.1305 ) faild2ErawCal =
    true;
if ( d1CalQual->nCrystals() < 4.0425 ) faild1nCrys =
    true;
if ( d2CalQual->nCrystals() < 4.0425 ) faild2nCrys =
    true;
if ( d1CalQual->lateralMoment() < 0.0125 ) faild1Lat =
    true;
if ( d2CalQual->lateralMoment() < 0.0125 ) faild2Lat =
    true;
if ( d1CalQual->absZernike42() < 0.0002 ) faild1A42 =
    true;
if ( d2CalQual->absZernike42() < 0.0002 ) faild2A42 =
    true;

cout << "L627\n";

failJBCuts = failMomMass || failMomMassErr ||
    faild1Costh || faild2Costh || faild1ECal || faild2ECal
    || faild1ErawCal || faild2ErawCal || faild1nCrys ||
    faild2nCrys || faild1Lat || faild2Lat || faild1A42 ||
    faild2A42;

if ( failJBCuts = false )
    { cout << "Passed_LB_Cuts!\n"; }
else
    { cout << "Failed_LB_Cuts.\n"; }

```

Example of π^0 veto code

```

BtaCandidate* reco(0);

while (reco = recoIter()){

```

```

BtaCandidate *boostedCand = new
    BtaCandidate(myBooster->boostTo(*reco));

double pCMMom = boostedCand->p();

delete boostedCand;
boostedCand = 0;

bool proceed(true);

BtaCandidate* daug;
HepAListIterator<BtaCandidate> iterDau =
    reco->daughterIterator();
while( (daug = iterDau()) != 0 && proceed){

    const BtaCalQual* CalQual =
        daug->getMicroAdapter()->getCalQual();

    double photonp3 = daug->p();
    double photonE = daug->energy();
    double photonphi = daug->p4().phi();
    double photonth = daug->p4().theta();
    float photonLat = CalQual->lateralMoment();
    float photonnCrys = CalQual->nCrystals();

    double photoncosth = cos (photonth);
    double photonsinth = sqrt(1 - photoncosth *
        photoncosth);

    HepAList<BtaCandidate>* phoList = Ifd<HepAList<
        BtaCandidate > >::get(anEvent, "CalorNeutral");

    HepAListIterator<BtaCandidate> iterPho(*phoList);
    BtaCandidate* photon(0);

    bool pizero(false);

    while ( (photon = iterPho()) != 0 && !pizero)
    {

        if( ! (daug->overlaps(*photon)) ){

            double compmass = photon->mass();

            if(compmass < 0.1){

```



```

    double compp3 = photon->p();
    double compE = photon->energy();
    double compth = photon->p4().theta();
    double compphi = photon->p4().phi();

    double compcosth = cos (compth);
    double compsinth = sqrt(1 - compcosth *
        compcosth);

    if(compE > 0.1){

        double d1d2cospsi = (photonsinth * compsinth
            * cos (photonphi - compphi)) +
            (photonncosth * compcosth);

        double invmass = sqrt( 2 * photonp3 * compp3
            * (1 - d1d2cospsi));

        if( invmass > 0.110 && invmass < 0.155){
            pizero = true;
        }
    }
}

if (!pizero && photonE > 0.15 && photonLat < 0.5 &&
    photonnCrys >= 4)
{
    cout << "Candidate_passed_cuts" << endl;
}
else
{
    proceed = false;
    cout << "Candidate_did_not_pass_cuts" << endl;
}
}

```

CURRICULUM VITA

Jacob W. Berg

Education

Ph.D., Physics, University of Louisville, Louisville, KY, Expected May 2021

M.S., Physics , University of Louisville, Louisville, KY, May 2013

B.S., Physics , University of Wisconsin Stevens Point, Stevens Point, WI, May 2011

Professional Positions

Graduate Teaching Assistant, University of Louisville Department of Physics, 2011-2013

Graduate Research Assistant, UofL High Energy Physics (HEP) Group, 2013-2015

Honors and Awards

Manuel Schwartz Award, University of Louisville Department of Physics, 2013