Higgs Boson Reconstruction and Detection Algorithm Analysis of the $h^0 \to Z^0 + Z^0 \to 2\mu^+ + 2\mu^-$ Decay Chain Using CMSSW

Matthew Sanders*
Washington and Lee University
(Dated: August 20, 2008)

Abstract

The Compact Muon Solenoid is expected to become operational towards the end of 2008. When it does, our understanding of the world of elementary particles could change dramatically with the discovery of new particles such as the Higgs boson. The CMS collaboration had already developed a programming package called CMS Software (CMSSW) to aid in the discovery of the Higgs boson and other new physics. Using CMSSW researchers can develop and test algorithms designed to successfully detect the Higgs boson. In this project, I developed and tested one such algorithm to reconstruct the Higgs boson from four muons. I then compared the effectiveness of my algorithm to the effectiveness of a similar algorithm that also reconstructs the Higgs boson from four muons.

Contents

I. INTRODCUTION

When the Large Hadron Collider at CERN becomes fully operational, CMS will be one of two general purpose experiments at CERN whose goal is to search for new physics. The CMS collaborations hopes to use CMS to satisfy three ends. They hope to probe physics at the TeV level, discover new physics beyond the standard model (i.e. supersymmetry and/or extra dimensions) and discover the Higgs boson [1]. The latter of these goals has been the focus of my research.

A. What is CMS?

The CMS collaboration is a collection of researchers from various educational and private research institutions. Together they created the CMS detector and the CMSSW high energy particle physics analysis package. While the researchers are spread across the world, the CMS itself detector is located underground in Cessey, France. As was already aforementioned, CMS stands for compact muon solenoid. However, if one was to look at a picture of the CMS detector, he/she would find that "compact" is a relative term (See Figure 1). Contrary to its name, the CMS detector is actually quite large as it weighs over 12,500 metric tones, has a length of over 21 meters and a diameter of over 15 meters.

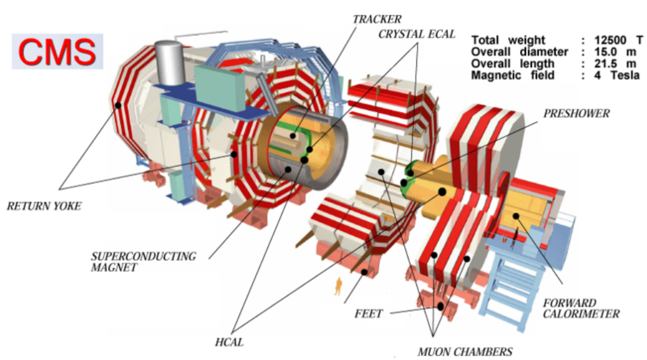

FIG. 1: A picture of the CMS detector with dimensions. Notice how small the person is in comparison. [1, 2]

The purpose of the CMS is detector is to look for new physics by through $p\bar{p}$ collisions at energy levels greater

*Dr. John Conway and Milan Nikolic, thank you for the direction and advice throughout my research.; Electronic address: `sandersm@wlu.edu`

than the capabilities of any current particle accelerator. The expected center of mass energy produced by the LHC is about 14TeV using a magnetic field of 3.8 Tesla [1]. At these energies, physicists hope that the of the constituents of the hadrons in the $p\bar{p}$ collisions will lead to the observation of new particles such as supersymmetric particles like squarks and sleptons and/or the Higgs boson. Of particular importance to my research is the possible discovery of the Higgs boson.

As its name implies, CMS will look for new particles, like the Higgs boson, primarily through the reconstruction of detected muons. That is not to say, however, that other particles cannot also be reconstructed. In fact, using CMSSW, one can reconstruct particles using just about any particle that CMS is capable of detecting. For example one can reconstruct the Higgs boson from the decay chain $h^0 \rightarrow Z^0 + Z^0 \rightarrow 2l^+ + 2l^-$ using any combination of $\mu^+\mu^-$, $\tau^+\tau^-$ or e^+e^-. Figure 2 shows how various particles are expected to behave in the CMS detector.

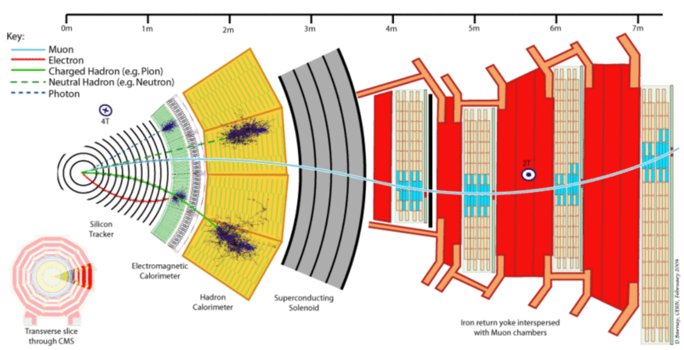

FIG. 2: Notice expected flight of the muons and how they penetrate many more layers of the CMS detector than any other particle. [1, 2]

B. The Higgs Boson A Brief History and Itroduction

The primary purpose of the CMS detector will be to find new physics. A particle of particular importance to the CMS collaboration is the Higgs boson. The Higgs boson is the only particle predicted by the Standard Model that has not yet been discovered. Physicists theorize that the Higgs boson gives mass to all elementary particles that have mass through the action of the Higgs mechanism. The existence of the Higgs boson was first theorized by a group led by physicist Peter Higgs in 1964 [3].

Physicists theorize that there may be several Standard Model Higgs bosons of different masses as well as Higgs bosons that extend beyond the Standard Model [3]. Physicists have estimated that the lowest possible mass a Higgs boson can have is $115\text{GeV}/c^2$–just outside the range of detection given our current particle accelerators (i.e. the Tevatron) [3]. However, given the size and capabilities of the LHC, CMS should easily be able to detect a Higgs boson with a mass of 115 GeV$/c^2$ if it exists.

The Higgs boson has been referred to as the "God Particle" in both popular culture and the scientific community [1] and for good reason. The discovery of the Higgs boson could prove to be a victory for the Standard Model of particle physics However, if the researchers of the CMS collaboration fail to discover the Higgs boson, then an overhaul of our entire understanding of the world of elementary particles must ensue.

C. An Introduction to CMSSW

CMS Software, more commonly referred to as CMSSW, is an open source software package designed by the members of the CMS collaboration for use with the CMS detector. CMSSW 1.6.8 was the primary tool I used for my research. Within CMSSW 1.6.8 one will find three powerful analysis tools.

The fist of these tools is the Monte Carlo high energy particle physics event generator. My research used a Monte Carlo event generator known as Pythia, but the CMSSW package contains other sources of Monte Carlo event generators such as ParticleGuns. Pythia is specially designed to work with Tevatron and LHC applications (like CMS)–in essence, Pythia produces a digital CMS. The build of Pythia (Pythia 6.4) available for use with CMSSW 1.6.8 is written in F77 (Fortran), but a newer version (Pythia 8.1) written in C++ has recently finished its development stages and is now ready for full scale use with LHC applications such as CMSSW [4].

The second analysis tool contained in CMSSW is ROOT. ROOT is a C/C++ interpreter with a graphics engine designed to analyze and plot large amounts of data. In the case of use with CMSSW, ROOT is used in conjunction with a user designed algorithm to analyze the Monte Carlo high energy particle physics events output by Pythia and plot the meaningful data contained within these events in histograms for further analysis.

The third analysis tool available in CMSSW is actually a software package in and of itself. The scramv1 build tool contains the compiler or interpreter for the user designed algorithm code which can be written in either C++ (a compiled language) or Python (an interpreted language). The scramv1 build tool also contains the "mastermind" behind the CMSSW package. Scramv1 build tool is designed to ensure that the user designed algorithm code, the Pythia output files and the ROOT histogram files all work nicely together without the user having to supervise. The scramv1 build tool is also able to build a workspace within which one can build, execute and save all of his/her codes and work.

Because CMSSW is open source, it is constantly changing as researchers develop new analysis packages for use. As an open source tool, the developers of CMSSW are constantly updating the CMSSW package and publishing analysis algorithms and results. To view these analysis packages as well as to learn more about CMSSW, visit the CMS Twiki website [5].

II. HIGGS BOSON RECONSTRUCTION THROUGH ALGORITHM DEVELOPMENT AND ANALYSIS

I came to the University of California, Davis high energy physics department with very basic understanding of elementary particle physics and no idea about what CMS and CMSSW were. Therefore, the first weeks of research with the UC Davis high energy physics department focused becoming accustomed to the various tools I would need to aid me in my research. I spent the first three weeks of my research program reading the framework tutorials on the CMS Twiki page [5], running through the CMSSW 1.6.8 particle reconstruction tutorials supplied by UC Davis graduate student Milan Nikolic on the UC Davis high energy particle physics website [6] and taking crash courses in C++ programming and ROOT. All of these tasks were supplemented with readings from an elementary particle physics text book [7].

A. Summer Project $h^0 \rightarrow Z^0 + Z^0 \rightarrow 2\mu^+ + 2\mu^-$

Following a three week crash course in to CMS, CMSSW and elementary particle physics, I was given a project to research. Figure 3 displays a Feynman diagram illustrating a possible decay chain for a Higgs boson created from a hard interaction between two gluons and the "miracle of the top quark," (Figure 4) as my advisor would say–consequently, the same picture also displays the results of the first ROOT macro I have ever written. The focus of my research would be to design an algorithm that successfully reconstructs a Higgs boson of mass 300 GeV/c^2.[1] In order to successfully reconstruct a Higgs boson candidate my algorithm had to:

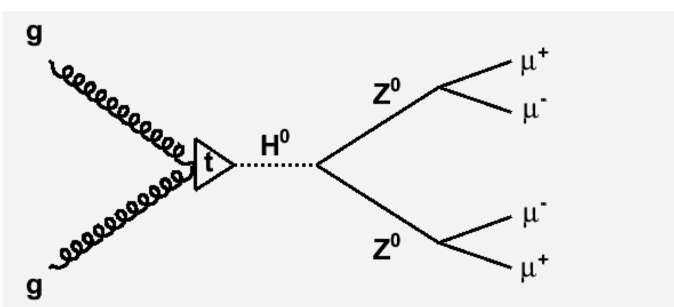

FIG. 3: A Feynman diagram of a Higgs decay to four muons.

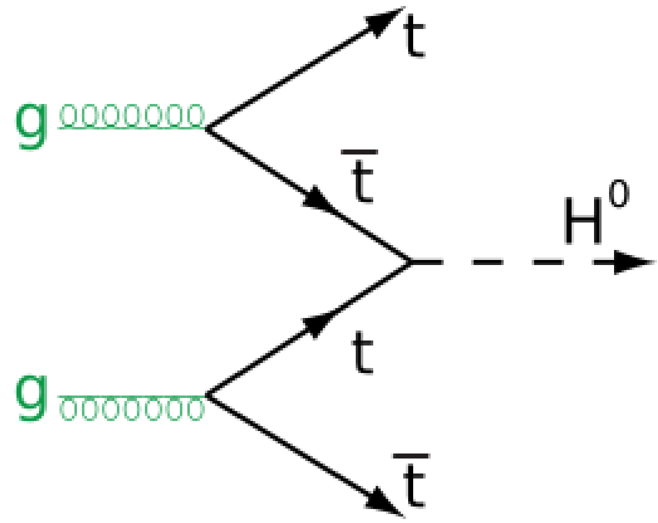

FIG. 4: "Miracle of the top quark" explained. [3]

1. Find all of the muons and anti-muons in an event.
2. Reconstruct the Z boson masses (91.1876 GeV/c^2) using the best $\mu^+\mu^-$ pairs.
3. Reconstruct the Higgs boson (300 GeV/c^2 in this case) mass from the Z bosons masses.
4. Plot all relevant data masses in ROOT histograms.

I quickly began working on my new research assignment. The first task that I undertook was data generation. I created two simulations in Pythia that exactly mimics the illustration shown in Figure 4. To create these simulations, I needed the help of the gaudy 489 page Pythia 6.4 manual [8] which contains a list of all of the commands and processes needed to create a Monte Carlo event generator that correctly mimics my Higgs decay chain. The first of the simulations was a "normal" simulation containing only 100 Higgs events while the second simulation was a "fast" simulation containing 10,000 Higgs events. The difference between the two simulations is the time it takes each simulation to finish execution–the fast simulation can finish 10,000 events in the same amount of time as it takes for a normal simulation to finish 100 events.

[1] The reason my research, at least at first, dealt with such a massive Higgs boson was because my advisor wanted to keep it as simple as possible at the start. A Higgs boson of 300 GeV/c^2 has a mass far greater than the combined mass of two Z bosons (mass 91.1876 GeV/c^2 each). However, once the mass of the Higgs boson in question falls below 183 GeV/c^2 one of the Z bosons becomes off-shell ,or virtual. So, in order to keep things simple at the beginning of my research, I had to deal with Z bosons that would remain real, or on shell–dealing with off-shell bosons is no more difficult than dealing with on-shell bosons when it comes to reconstruction, but the concept is more advanced.

After a day of running the Pythia simulations, it was time to work on developing a successful algorithm. However, before I could begin serious work on my algorithm, I had to make sure the simulation data I had just generated was mimicking the Higgs decay displayed in Figure 3. In order to ascertain whether or not my simulation was behaving properly I used the analysis code from Milan's CMSSW 1.6.8 tutorial that is posted on the UC Davis HEP website [6]. This rudimentary code simply finds all the muons produced in a single event and reconstructs them to their "parent particle"–in the case of Milan's tutorial, a graviton; in my case, a Higgs boson–using the muons' Lorentz, or four, Vector ($< p_x, p_y, p_z, E >$). The code then reconstructs the mass of the parent particle using the information contained within the Lorentz Vector and the equation $E^2 = m^2c^4 + p^2c^2$. Using Milan's

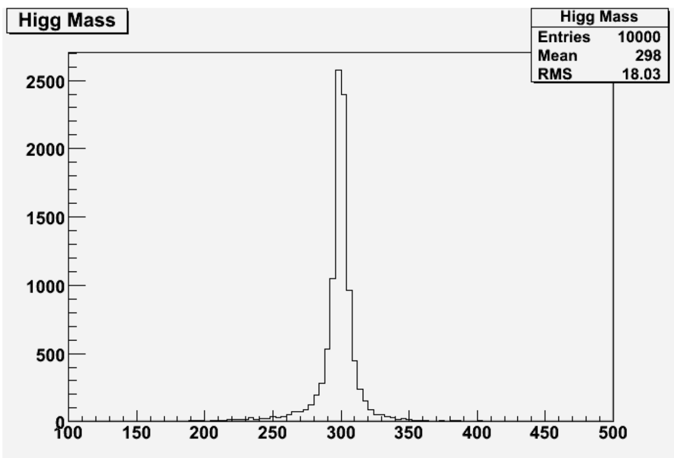

FIG. 5: First ROOT histogram produced using Milan's code. Notice the peak is at the mass of the Higgs boson generated by the simulation. This indicates that the simulation is probably working properly.

code produced the histogram shown in Figure 5. The peak of the histogram is located at 300 GeV/c^2 which indicates that the simulation worked properly to mimic the Higgs decay pictured in Figure 4. Milan's code was a start, but it did not employ the appropriate method for reconstructing the Higgs boson from muons.

Knowing that the Pythia simulations was producing Higgs events, it was time to begin developing my own analysis code. I began with a template provided on the CMS Twiki page by the Higgs Analysis group [5]. The code I found reconstructed the Higgs candidates using an algorithm close to the appropriate algorithm, so I decided that this code would be a good place to start. I rewrote the code to become familiar with the function calls contained within CMSSW and also to eliminate pieces of code not pertinent to my analysis. I then compiled and ran my rewritten code. The results were not ideal.

My re-written algorithm did reconstruct the $\mu^+\mu^-$ pairs to Z bosons with which in turn were constructed to a Higgs boson, but as is demonstrated in Figures 6

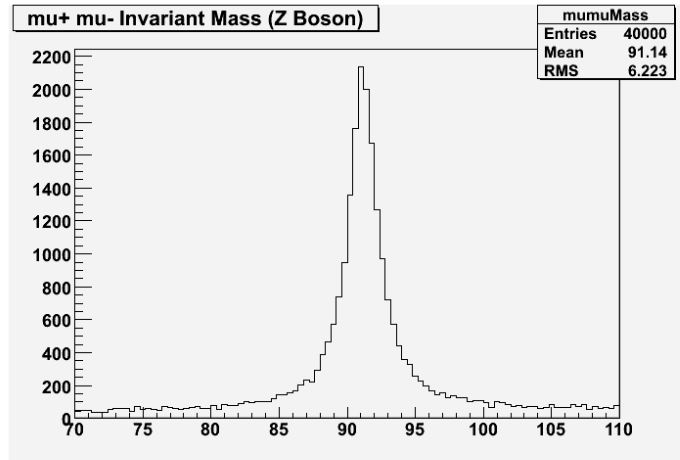

FIG. 6: Reconstructed Z boson mass using first attempt (re-written) at algorithm

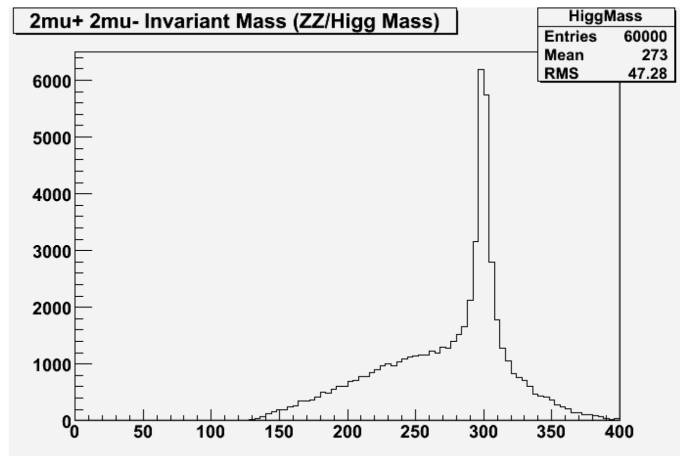

FIG. 7: Reconstructed Higgs mass using first attempt (re-written) at algorithm

and 7, although the peaks of the histograms are in the proper locations, the number of data entries is seriously off–40,000 Z boson mass entries appear even though there are 10,000 events with each event containing only two Z bosons; a similar problem occurs with the Higgs mass reconstruction. After some investigation, I found that the problem with the algorithm occurred because I programmed my code to reconstruct all possible $\mu^+\mu^-$ pairs into Z bosons instead of reconstructing the best possible pairs (a method for determining the "best possible" pairs has yet to be developed). Figure 8 depicts the problem in my code.

Knowing that the logic error depicted in Figure 8 was responsible for the problems in my code, I decided to use Z boson mass cuts to determine which $\mu^+\mu^-$ pairs were the best pairs. Implementing mass cuts requires that the code first reconstructs all possible $\mu^+\mu^-$ pairs and checks to see which ones, when reconstructed, yield a mass closest to the Z boson mass of 91.1876 GeV/c^2. Then the

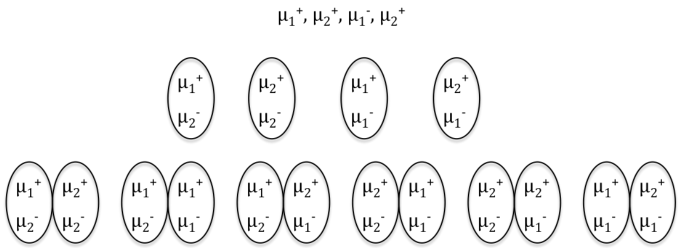

FIG. 8: The logic error in my code accounts for the incorrect number of Z boson and Higgs boson reconstructions

code has to take the two $\mu^+\mu^-$ pairs with masses closest to the Z boson mass and put them into their own C++ array to be reconstructed to the Higgs boson. Using these mass cuts ensures that only two Z boson candidates in any single event will be reconstructed to a Higgs boson, and since the Higgs decays into two Z bosons only one Higgs candidate will appear per event. Figures 9 and 10 show the results of implementing these new mass cuts.

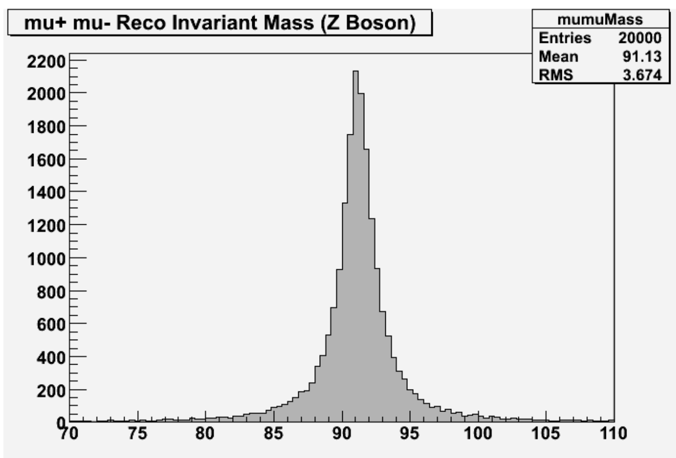

FIG. 9: Reconstructed Z boson mass using mass cuts in algorithm

While the Z boson mass histogram in Figure 9 looks correct, the Higgs mass histogram in Figure 10 looks incorrect because of its left skew (it should be nearly Gaussian in distribtution). I could not determine what the problem was. I decided to move on, for the time being, and as per my advisor's instructions, I began to generate Z boson mass scatter plots–$(x, y) \rightarrow (Z_{m_1}, Z_{m_2})$–of both reconstruction level Z bosons and generator level Z bosons (generator level Z bosons are the actual Z bosons produced by the Pythia simulation in an event while reconstruction level Z bosons are the "best guess" of the analysis code based on the information provided by the detected muons). After adding a few lines to my code, I was able to generate the Z boson mass scatter plots shown in Figures 11 and 12. However, what good is it to plot the reconstruction level Z boson mass scatter and the generator level Z boson mass scatter if one does

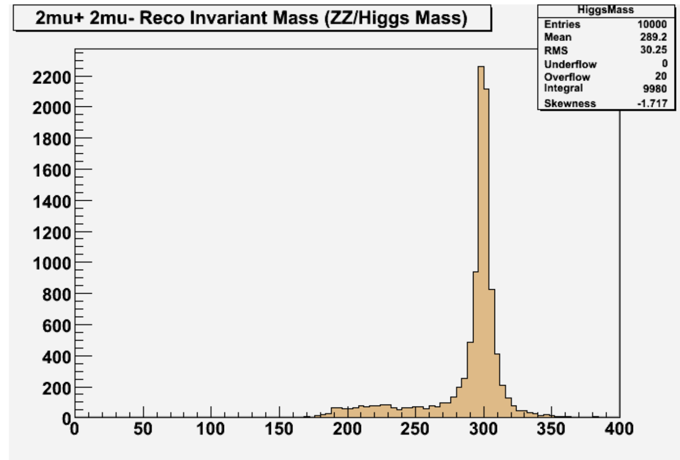

FIG. 10: Reconstructed Higgs mass using mass cuts in algorithm

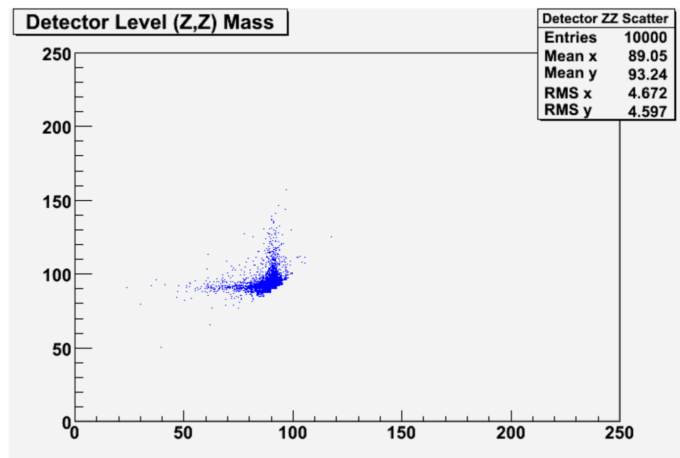

FIG. 11: Reconstructed level Z boson mass scatter

not know what the difference is between them? In order to determine just how often my code was reconstructing the correct $\mu^+\mu^-$ pairs, I created a subtraction plot that is equal to the difference between the generator level Z boson mass scatter plot and the reconstruction level Z boson mass scatter plot. If either the x coordinate Z boson mass or the y coordinate Z boson mass differ between the two plots for a given event, then the reconstruction level Z boson masses are added to the subtraction plot (as shown in Figure 13).

Using the information provided on Figure 12, one can see that my algorithm correctly reconstructed the correct $\mu^+\mu^-$ pairs to Z bosons 87.6% of the time. While this is a good number for the percentage of correct reconstructions, it is not good enough–my advisor informed me that the percentage should be 95% or higher. Therefore, as I suspected when I saw Figure 10, there must still be a problem with my algorithm.

It took a while, but eventually I pinned down what the problem in my code was. Using Figure 8 one can deduce

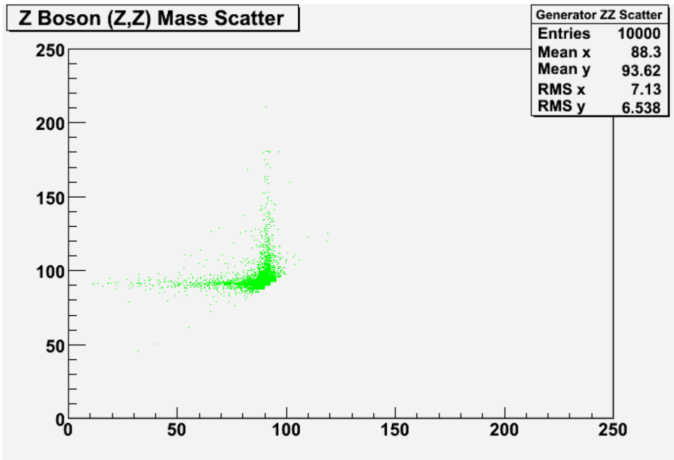

FIG. 12: Generator level Z boson mass scatter

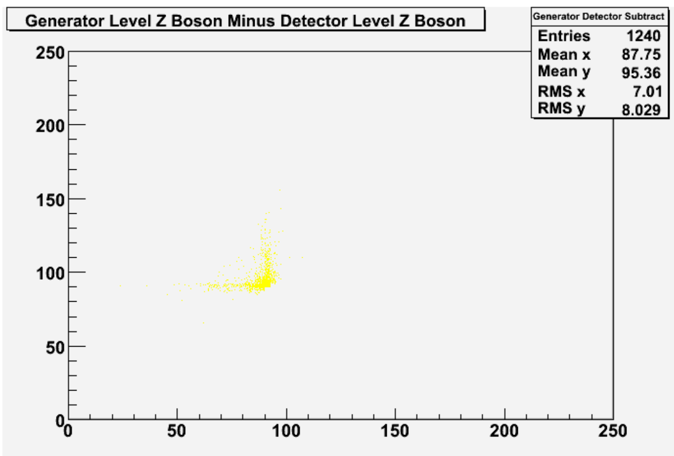

FIG. 13: Generator level minus reconstruction level Z boson mass scatter

what the error in my code is. My algorithm uses Z boson mass cuts to eliminate the non-ideal $\mu^+\mu^-$ pairs, but it does not ensure that the same muon is not used twice in the reconstruction of the Higgs boson. For example, there is no safeguard in my code to ensure that first Z boson in Figure 8 is not paired with the second Z boson in Figure 8 even though each of these reconstructed Z bosons is reconstructed from the same μ_2^- muon. This error in my algorithm must be why the Higgs boson mass histogram in Figure 10 was skewed to the left instead of Guassian in distribution.

To fix the error in my code I needed to add another Z boson mass check. However, this time, after reconstructing all of the possible $\mu^+\mu^-$ pairs into Z bosons and selecting the pair to have a mass closest to 91.1876 GeV/c^2, the algorithm needs to again reconstruct all of the $\mu^+\mu^-$ pairs, and as it does so, check to see which pair equals the already selected Z boson mass. When it finds this pair, the algorithm must then delete the $\mu^+\mu^-$ pair from the C++ array containing the all of the muons. Doing this second mass check would ensure that the same muon is not used twice. Once the second mass check is complete, the algorithm proceeds as it did before–it finds the second $\mu^+\mu^-$ pair that has a mass closest to 91.1876 GeV/c^2 and appends it to the Z boson C++ array. Using my new algorithm with this new method of Z boson mass cutting produced the nearly Gaussian histograms shown in Figures 14 and 15.

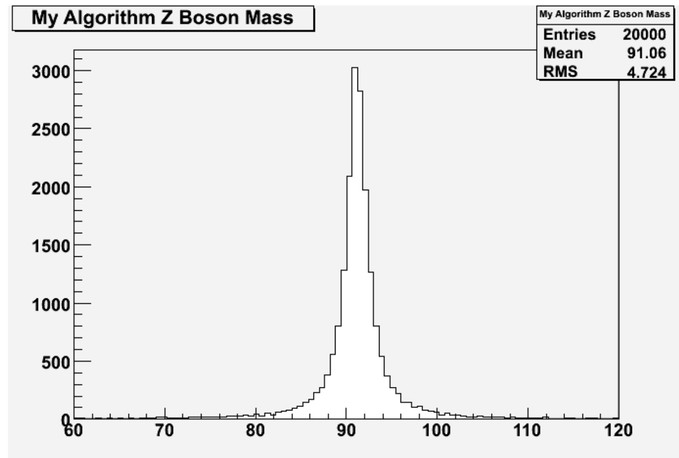

FIG. 14: Reconstructed level Z boson mass using corrected algorithm

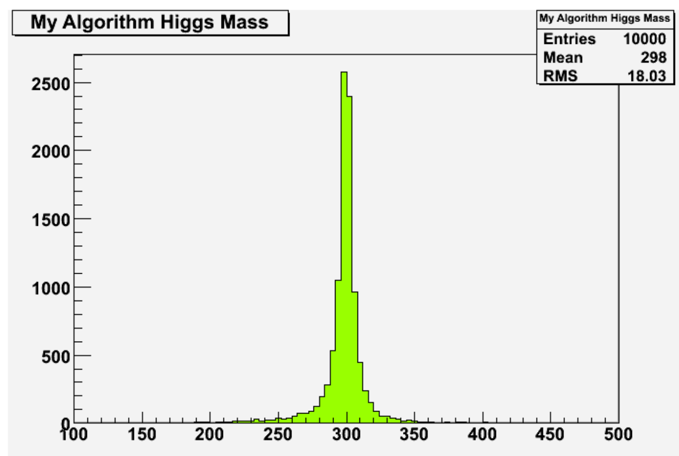

FIG. 15: Reconstruction level Higgs mass using corrected algorithm

The new mass cut to my algorithm fixed the skewness of the Higgs mass histogram present in Figure 10, but did this new addition make my algorithm any more accurate in its Z boson reconstructions? To check, I again implemented the same subtraction algorithm that produced Figure 13 as before. As can be seen from Figure 16 the new mass cut in my algorithm did indeed affect the accuracy with which my code reconstructed Z bosons. Using the new mass cut, my algorithm reconstructed the Z bosons from the $\mu^+\mu^-$ pairs 97.29% of the

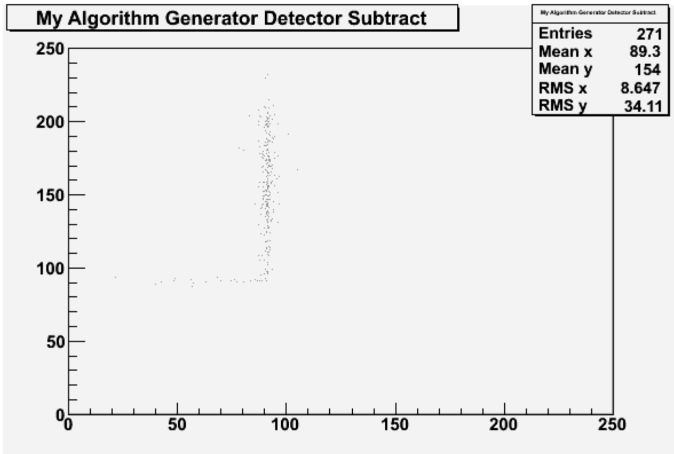

FIG. 16: Generator level minus reconstruction level Z boson mass scatter using corrected algorithm

time (9,729 correct reconstructions out of 10,000 possible events). This percentage is much closer to what should be expected from a successful reconstruction algorithm–remember, my advisor said the correct Z bosons should be reconstructed upwards of 95% of the time.

B. Comparing Algorithms Without Detector Background

Having successfully developed and tested my own algorithm for reconstructing Higgs bosons from two muons and two anti-muons, it was time to compare my algorithm to algorithms currently being developed by scientists working within the CMS collaboration. My first goal was to get my hands on one such algorithm. I received this algorithm via a printed e-mail handed to me by Dr. John Smith of UC Davis High Energy Experimental Physics group. The algorithm in question had been developed by a group of physicists from the University of Florida led by Andrey Korytov. Their algorithm was different from mine in two main respects. First, their algorithm was designed to hand events with four or more muons per event (i.e. their algorithm can handle detector background)–presumably, my algorithm could also handle this background, but I had not tested my algorithm to this particular end. Second, their algorithm contained muon cuts–a set of requirements that the muons had to pass in order to even be considered in the analysis. The muon cuts from their algorithm are as follows:

Accept muons for analysis from a given event if:

1. $(P_t)^2 > 5$ for $|E| < 1.1$
2. $(\|P\| > 9$ and $P_t > 3)$ for $1.1 < |E| < 2.4$
3. If there are four or more muons, the first correct $\mu^+\mu^-$ pair is the pair with a mass closest to the mass of the Z boson
4. The second $\mu^+\mu^-$ pair will be constructed from one of the remaining μ^+ with the highest P_t and one of the remaining μ^- with the highest P_t
5. Drop the entire event if the reconstructed Higgs mass is less than 100 GeV/c^2

[2] Transverse Momentum

In order to compare my algorithm to the advanced algorithm given to me by Dr. Smith, I first had to program this new algorithm into my code. The coding itself was straightforward and within two days I began generating results using both my algorithm and the advanced algorithm. In order to determine if the advanced algorithm I had programed into my code was working correctly, I needed to create a muon multiplicity plot for the advanced algorithm as well as muon energy and transverse momentum plots. Having the muon multiplicity plot (Figure 17), energy plot (Figure 18) and transverse momentum plot (Figure 19) for the advanced algorithm would allow me to determine if the advanced algorithm was indeed making the muon cuts as described in the advanced algorithm outline given above.

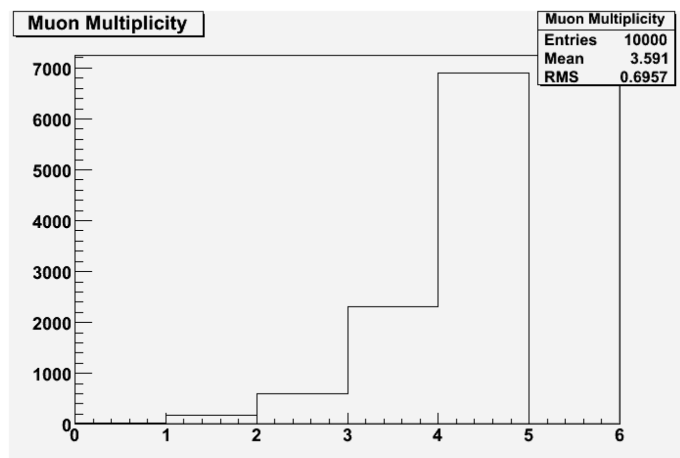

FIG. 17: Muon multiplicity for advanced algorithm

The muon multiplicity plot, the muon energy plot and the muon transverse momentum plot for the advanced algorithm did indeed show that the advanced algorithm was working properly. With both algorithms working properly, it was time to compare the Z boson and Higgs boson mass histograms. The histograms for my algorithm did not change, of course, but I generated new histograms for the advanced algorithm. The Z boson mass histogram and the Higgs boson histograms produced by the advanced algorithm are shown in Figures 20 and 21.

Comparing Figures 20 and 21 to Figures 14 and 15 one can see that the difference between the respective Z boson mass plots and Higgs mass plots for both my algorithm and the advanced algorithm is negligible when

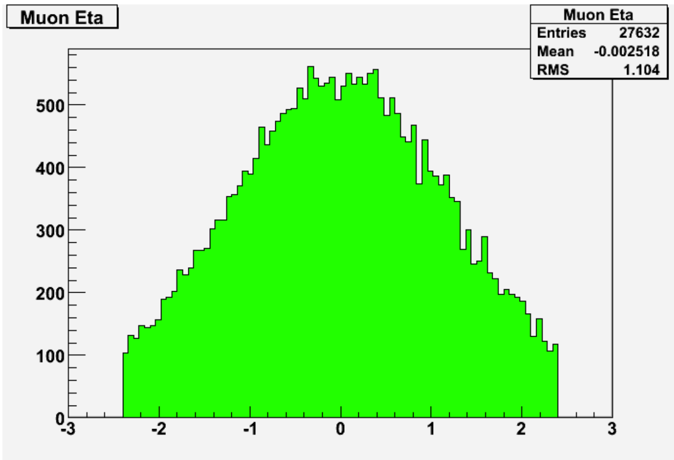

FIG. 18: Muon energy for advanced algorithm

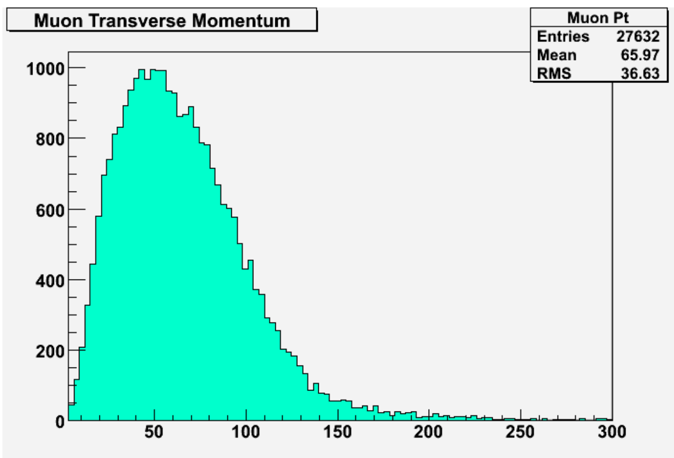

FIG. 19: Muon transverse momentum for advanced algorithm

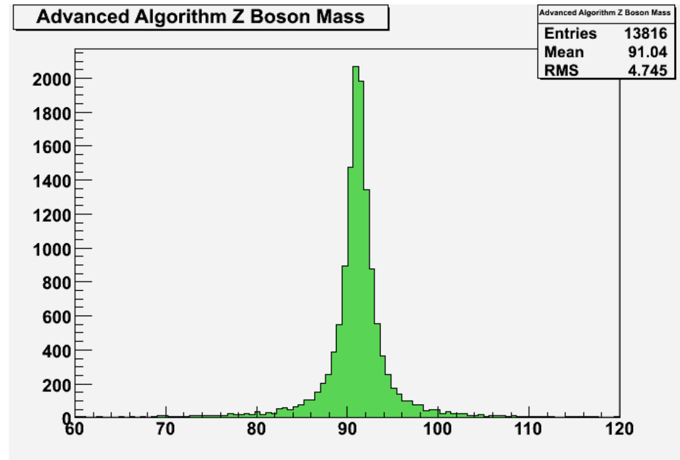

FIG. 20: Reconstructed level Z boson mass using advanced algorithm

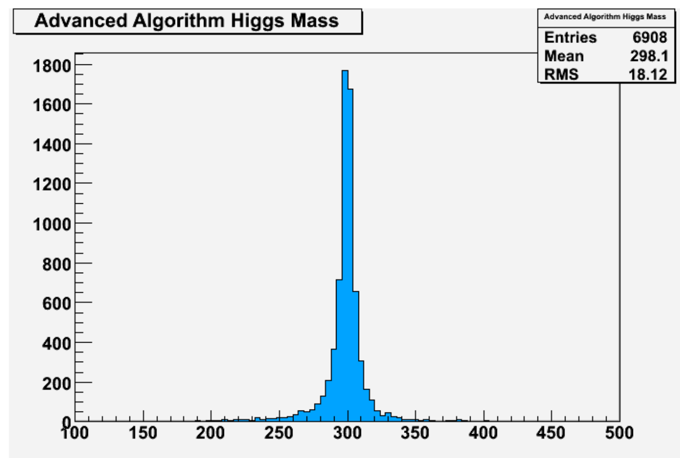

FIG. 21: Reconstruction level Higgs mass using advanced algorithm

there is no detector background. The main difference is a difference between the number of entries and slight differences between the averages of the masses in each histogram. Of course, these small differences are to be expected when analyzing events with only four muons present because essentially, the two algorithms are the same when the number of muons present is equal to four– both my algorithm and the advanced algorithm find the $\mu^+\mu^-$ pair closest to the Z boson mass, and although the advanced algorithm sorts the remaining muons according to their transverse momenta, there are only two muons left, so both algorithms reconstruct the remaining muon and anti-muon into the remaining Z boson. The only difference between the two algorithms when only four muons are present is the muon cuts in the advanced algorithm which lead to a smaller number of event entries. The similarities between the two algorithms are further pointed out when one looks at the generator level minus reconstruction level Z boson mass scatter for the advanced algorithm (see Figure 22) and compares it to the same plot for my algorithm.

Figure 17 shows that the advanced algorithm correctly reconstructs the Z bosons 97.21% of the time (6,715 correct reconstructions of out 6,908 viable events). Compared to the 97.29% correct reconstructions produced by my algorithm, the percent difference of 0.08% is not significant. A true test of the difference between the two algorithms must include detector background.

C. Comparing Algorithms With Detector (Pseudo)Background

In order to truly evaluate the differences between these algorithms, I had to either generate or locate simulation files that contained CMS detector background (i.e. extra muons in the simulations not created as a decedent of the Higgs decay chain). I first tried to look for the files myself on the UC Davis CMS network, but had no luck in finding them. Next, I tried to generate the simulations myself

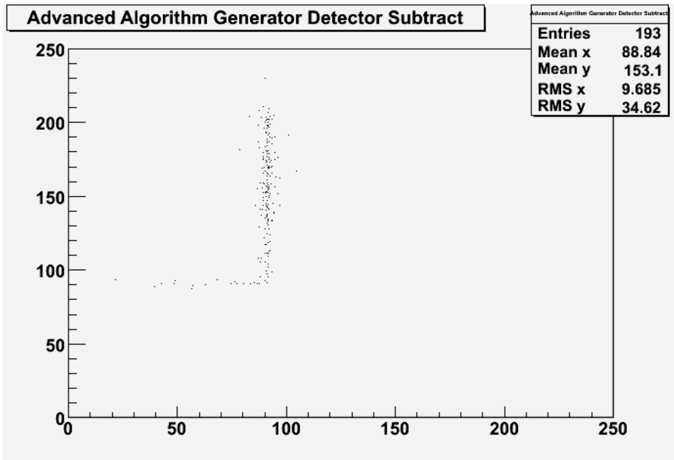

FIG. 22: Generator level minus reconstruction level Z boson mass scatter using advanced algorithm

using Pythia, but I could not find a way to include detector noise in the Monte Carlo event generation. Luckily, Milan was able to track down the background files and provide them to me.

The signal files and background files produced by the simulations Milan had run were far larger than any data I had run through to that point. The files on average contained 200,000 events (the largest had over 500,000 events and the smallest had 80,000 events). The events contained in the signal files also contained Higgs decays involving not only Higgs masses smaller than 183 GeV/c^2 (the magic number for one on-shell and one off-shell Z boson), but they also contained leptons other than muons and anti-muons (see Figure 23). These extra leptons

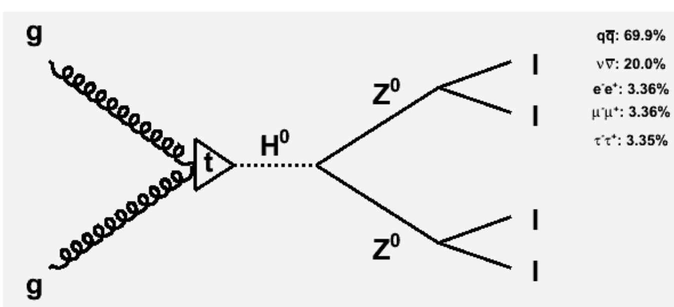

FIG. 23: The Higgs decay chain present in the lighter mass Higgs signal files

would not affect my code[3]–as my code only recognizes muons–but they would reduce the number of Higgs reconstructions present because only 11.3% of the events in the signal files would contain four muons (according to the percentages given on Figure 23).

I now had the signal and background files that I needed to continue my research, but there was still a problem. The signal and background files must be analyzed at different times. Therefore, the background is not true background because it is not contained within the same event as the actually Higgs decay, but rather it is contained in a wholly separate event that is designed to mimic a Higgs decay. Because of this nature between the signal and background files, the comparison of my algorithm and the advanced algorithm would come down to which processed the least (or the least significant) information from the background files because both algorithms would produce nearly the same results when reconstructing actual Higgs events (as described towards the end of the previous section).

With this signal and background file problem in mind, I began analyzing the several terabytes of data using both my algorithm and the advanced algorithm. I analyzed three different series of Higgs events. Each series of Higgs events contained Higgs bosons of different masses. One series contained a 150 GeV/c^2 Higgs decay chain, another contained a 180 GeV/c^2 Higgs decay chain, and the last was the signal file from my previous analysis containing a 300 GeV/c^2 Higgs decay chain. I also analyzed three background files designed to mimic Higgs decays. The first background file contained $Z^0 + Z^0 \rightarrow 4l$ decay chain, the second contained $t + \bar{t} \rightarrow 4l$ decay chain and the last contained $Z^0 + (b + \bar{b}) \rightarrow 4l$ decay chain. I then plotted all of the significant quantities (Z boson mass, Higgs boson mass, muon energy and muon transverse momentum) for each signal and background file. The results can be seen if Figure 24–the $t + \bar{t} \rightarrow 4l$ event and the $Z^0 + (b + \bar{b}) \rightarrow 4l$ events were not plotted because their analyses did not produce any results (no four muon events).

As is evidenced in Figure 24, the advanced algorithm did handle the noise (purple) significantly better when it came to reconstructing the Higgs boson and on the lower end of the Z boson mass reconstruction, and the muon transverse energy plot, but the muon energy plot shows no significant difference. Furthermore, the background peak on my algorithms' Higgs mass plot is below 100 GeV/c^2. Since discovering particles with this mass is within our current technological abilities, the peak is insignificant because no one will be looking for a Higgs boson in this mass range.

D. Further Exploration

My stay with the University of California, Davis physics department finished before I could really get into the meat of CMS physics. There are still many functionalities and new algorithms that I would like to have included in my program that I cannot because of my limited time here. The following new functionalities and algorithms are possible ares for further exploration into

[3] In fact, as mentioned before, one could re-write these codes to handle all leptons (a potential project that will be discussed later)

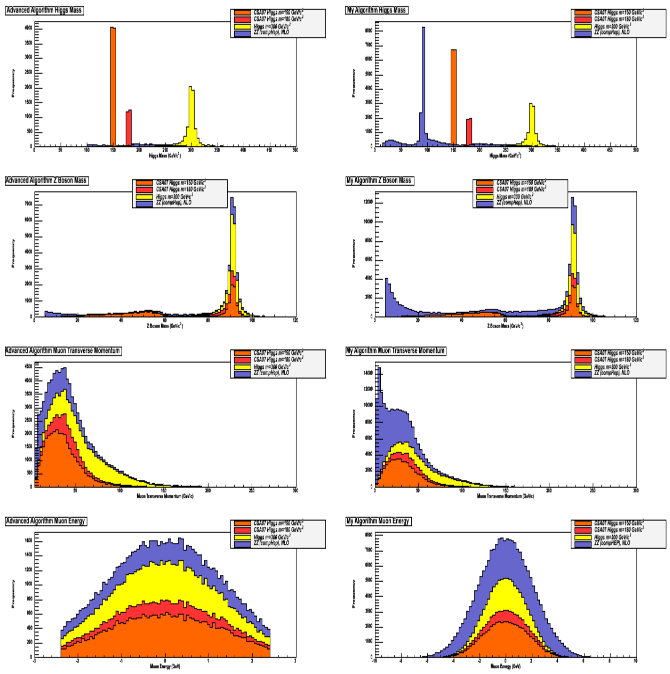

FIG. 24: A series of histograms containing mass plots, energy plots and transverse momentum plots for several Higgs decay chain analyses and background analyses

CMS.

1. Look For the Presence of $Z^0 \rightarrow \mu^+ + \mu^- + \gamma$ Decays

Every once in a while, a Z boson will decay into a muon, an anti-muon and an energetic gamma ray. The gamma ray carries off some of the energy from the Z boson decay which could cause potential inaccuracies when reconstructing the muon masses back into the Z boson mass–keep in mind that Einstein's famous equation $E = mc^2$ says that mass and energy are interchangeable. In order to account for this possible loss of mass, one would have to develop an algorithm that could detect documentation level gamma rays from the detector simulation and recombine the gamma ray with the appropriate detected muons. Doing this type of reconstruction could lead to even more accurate Z boson mass scatter plots (accuracies above 98%).

2. $h^0 \rightarrow Z^0 + Z^0 \rightarrow 4l$ Decay Chains

As I mentioned previously in this paper, the $h^0 \rightarrow Z^0 + Z^0 \rightarrow 4l$ decay chain is not limited to a four muon decay. Figure 23 shows that this decay chain can decay to any combination of e^+e^-, $\mu^+\mu^-$ or $\tau^+\tau^-$. It would be an interesting project to develop and test an algorithm that can successfully and efficiently handle all of the combinations of these lepton decays and reconstruct them to Z bosons and in turn into, Higgs bosons. It would be especially challenging if one could develop such an algorithm that functioned successfully and efficiently even with true detector background–with detector background the algorithm would have to successfully negotiate up to eight or more leptons in a given event!

3. Geometric Cuts

The advanced algorithm in used my research used certain standards to determine which of the detected muons would be analyzed. The advanced algorithm mass cuts accepted or rejected muons based on their energies and momenta. However, why not determine if a muon is to accepted or rejected based on its geometric track location inside the CMS detector? In essence, if it can be reasonably determined that four muons came from the same point in space, then those muons should be kept. I will not go into further detail here because a group of researchers in the University of California, Davis High Energy Particle Physics Department is currently investigating this very possibility and are in the process of publishing a paper with their results!

III. CONCLUSION

Over the past ten weeks, I believe I accomplished a lot here at the University of California, Davis. Not only did I get a practical education and introduction into the world of elementary particle physics, but I advanced my knowledge of the C++ programming language through the trial and error of developing analysis algorithms, learned how to use ROOT, CMSSW 1.6.8, the Linux Shell and Pythia. I even learned to LaTeX in order to type this paper. I also got to meet and get to know a great group of fellow students within both the REU program and the University of California, Davis Physics Department.

IV. SPECIAL THANKS

First, I would like to thank the University of California, Davis Physics Department for giving me the opportunity to participate in the REU Program with their faculty and student body. I would also like to thank the National Science Foundation for providing the funding to make the REU Program possible. I would like to thank my advisor John Conway. Thanks for advising and directing me on this project. I learned a lot from the process, and I have developed a strong interest in particle physics. I would also like to thank John's wife, Robin. I know you just had a baby boy, Ian, so thanks for allowing John to come into the lab and direct my studies when I needed his help. I also want to thank Milan Nikolic for providing me with help and direction when John was busy with Ian. Thanks for helping to get me started with CMSSW and ROOT.

Without your guidance, tutorials and ROOT macros, I never would have accomplished as much as I did. Thanks also to the National Science Foundation for funding the UC Davis REU program and making this research possible. Lastly, I want to thank my fellow REU program member and roommate, Jake Ellowitz. Thanks for all of your help and thanks for being a good roommate.

[1] t. f. e. Wikipedia, *Compact muon solenoid*, internet (2008), URL http://en.wikipedia.org/wiki/Compact_Muon_Solenoid.

[2] C. Collaboration, *Cms outreach*, internet (2008), URL http://cmsinfo.cern.ch/outreach/CMSmedia/CMSphotos.html.

[3] t. f. e. Wikipedia, *Higgs boson*, internet (2008), URL http://en.wikipedia.org/wiki/Higgs_boson.

[4] P. S. Torbjörn Sjöstrand, Stephen Mrenna, *Pythia* (2008), URL http://home.thep.lu.se/~torbjorn/Pythia.html.

[5] *Cms wiki homepage*, internet (2008), URL https://twiki.cern.ch/twiki/bin/view/CMS/WebHome.

[6] *Uc davis hep homepage*, internet (2008), URL https://hep.physics.ucdavis.edu.

[7] D. Griffiths, *Introduction to Elementary Particles* ("John Wiley & Sons, Inc.", 1987).

[8] P. S. Torbjörn Sjöstrand, Stephen Mrenna, *Pythia 6.4 Physics and Manual* (FERMILAB-PUB-06-052-CD-T, 2006).