Reconstruction of the Higgs mass from 4 leptons at the Atlas detector using the Athena framework

Summer student report
26th July-19th September 2006

Abstract:

In this paper I do a detection of the Higgs boson mass at one of the most background free channel of the Higgs decay: H->ZZ->4leptons. From an analysis object data file (.aod) I take signal data from this channel, for mH=130GeV and I do combinations of 4 leptons. Because the highest branching ratios for the Z->2l are Z->ee and Z->mumu, the vectors I add are belong to 4electrons, 4 muons or 2 electrons+2muons. The mass of the resulting composite particle should have a peak to the 130GeV. The data is generated with Pythia and the analysis is done on Athena, a framework specialized for the Atlas detector.
A secondary objective of my analysis is a comparison between the Muon Muid Container and the Muon Staco Container, to check which one goes better for a Muon Container.

Student: Madalina STANESCU-BELLU
Supervisor: Dr. Klaus MOENIG

Contents:

CHAPTER 1. INTRODUCTION

1.1. Searching for the Higgs –theoretical motivation

The Standard Model answers many of the questions of the structure and stability of matter with its 6 quarks, 6 leptons and 4 forces. But it still leaves unresolved problems. It proposes that there is another field not yet observed called the Higgs field. We think that all of space is filled with this field, and that by interacting with this field, particles acquire their masses. The Higgs field has at least one new particle associated with it, the Higgs particle (or Higgs boson). This is what the LHC (Large Hadron Collider) tries to detect, if this particle exists.

To create a high mass particle we need high energies because some of the incoming energy can be used to create new particles. Higgs boson should be very massive => can only be created in the collisions of very high energy beams.

Then to detect the particle, we can do it by direct measurement, by decay or by radiative corrections. ***The most background free decay channel*** to detect the Higgs, even though they have BR just 1% : H->γγ (for light H) and H->ZZ->4l(for heavy Higgs).

LEP past direct Search: mH > 114GeV. LHC is dealing with it now.

1.2. The Atlas experiment

ATLAS (A Toroidal LHC ApparatuS) is one of two General Purpose Detectors at the CERN Large Hadron Collider (LHC). The LHC will collide 7 TeV protons together with a centre of mass energy of 14 TeV and a design luminosity of 10^34 cm^-2 s^-1. The bunch crossing time will be 25ns and at full luminosity there will be approximately 22 proton-proton collisions per bunch crossing. The ATLAS Detector is situated at Point 1, directly opposite the CERN main entrance. The ATLAS detector consists of four major components, the Inner Tracker which measures the momentum of each charged particle, the Calorimeter which measures the energies carried by the particles, the Muon spectrometer which identifies and measures muons and the Magnet system that bends charged particles for momentum measurement. The detector is a cylinder with a total length of 42 m and a radius of 11 m and weighs approximately 7000 tonnes.

The ATLAS Coordinate System:
The ATLAS Coordinate System is a right-handed system with the x-axis pointing to the centre of the LHC ring, the z-axis following the beam direction and the y-axis going upwards. In Point 1, positive z points towards Point 8 with a slope of -1.23%. The azimuthal angle \phi=0 corresponds to the positive x-axis and \phi increases clock-wise looking into the positive z direction. The polar angle \theta is measured from the positive z axis.

1.3 The Atlas Software: Athena Framework

Athena is a control framework with a Gaudi kernel and classes from Atlas specifications. Particles are identified in Atlas detector and then saved in ***Athena Containers.***

CHAPTER 2. INPUT/OUTPUT SYNOPSIS:

2.1 The input package -

The source code structure, the purpose of the python file, events & tracks in the AOD

a) The UserAnalysis package

The ATLAS Software is organized into a hierarchical structure of Projects and Packages:

The ATLAS offline (which analyzes data) is split into the following set of projects: AtlasAnalysis(which depends on the Atlas Trigger), AtlasCore(contains core components and services, e.g. Athena and StoreGate), AtlasEvent, AtlasReconstruction, AtlasSimulation, AtlasTrigger etc.

In each project, the software is organized in a number of packages. The main domains are: Generators, Simulation, Trigger, Reconstruction, PhysicsAnalysis, InnerDetectorSoftware, MuonSoftware, LArCalorimeter, TileCalorimeter etc.

 I WORK IN ATHENA -> USERANALYSIS PACKAGE ->PHYISICS ANALYSIS.

You have first to prepare you account for the Atlas Software (Athena) and then copy the UserAnalysis package.

The USERANALYSIS package contains, among the others:
- /src/AnalysisSkeleton.cxx -- the C++ code source file
- /UserAnalysis – the header file
- /run/AnalysisSkeleton_jobOptions.py – the job Options file

b) The purpose of the AnalysisSkeleton_jobOptions.py script

- a conventional text file used to control the configuration of an Athena application at run time.
- It loads some other python scripts that contain specifications;
- It specifies the data input file:

EventSelector.InputCollections =["csc11.005300.PythiaH130zz4l.recon.AOD.v11004204._00001.pool.root"]

- It appends libraries and specifies what algorithms to run:

theApp.Dlls   += ["TruthParticleAlgs"]
theApp.Dlls   += ["AnalysisTools"]
theApp.Dlls   += ["UserAnalysis"]
AnalysisSkeleton = Algorithm("AnalysisSkeleton")

- It specify the root output file, the log output file, the total number of events
- **The most important:** it specifies some properties that also find themselves within the cxx file. In the cxx file the properties have default values, but the values from the job options file have priority. This way the cuts on the particle parameters can be easily modified, without being necessary to recompile the source code:

AnalysisSkeleton.ElectronContainer = "ElectronCollection
AnalysisSkeleton.ElectronPtCut = 10.0*GeV

   c) **The input file. The structure of the AnalysisSkeleton.cxx file. The event--track terminology.**

 AOD = Analysis Object Data;
CBNT = Combined N-tuples (Software);
ESD   = Event Summary Data;
TDS   = Transient Data Store ;
TES   = Transient Event Store;

The input file is an Analysis Object Data file, generated by Pythia. The data is a signal data, so background free. The channel analyzed is H→ZZ->4 leptons for mH=130GeV and has data for many generated events.

Regarding the source code, the order in which the predefined methods are called at running is:
CBNT_initialize()→ CBNT_clear()→ CBNT_execute()→ CBNT_clear()→ CBNT_finalize(). The execute method does the analysis itself, by calling the user defined methods: electronIterator(), muonIterator(), Z_reconstruction(), higgs_reconstruction().

The execute method is called N times, where N is the number of events specified in the python script, so the algorithm reads N events signal data from the AOD input file. Each iterator method performs a loop on the corresponding particle container, to catch every track from that corresponding event. So the loop is performed N times, for each event. Each reconstruction method performs a loop on the container and creates combination of 2,4 particles until all the tracks are examined. The loop is also repeated N times.

Event = a process, one Higgs decays into 4 leptons
Track = a signal that a particle within the process (H, Z, e ,mu) creates at the corresponding container

2.2 The containers & the classes used

   a) **The Classes**

The AOD particle classes are the basic objects to be used in physics analysis, and are the bulk of the AOD. Each particle class has some basic properties defined by the *IParticle* interface and adds additional functionality.
The *IParticle* interface extends the *I4Momentum*(an abstract base class providing 4-momentum behavior) and the *INavigable*(allows one to 'navigate' to the constituents of the particle) interfaces.

Class reference over the most used members in this analysis:

- virtual double I4Momentum::   **eta()** const = **pseudo rapidity**
- virtual double I4Momentum::   **phi()** const = **phi** in [-pi,pi]
- virtual double I4Momentum::   **et()** const = **transverse energy** defined to be e*sin(theta)
- virtual double I4Momentum::   **pt()** const = **transverse momentum**
- virtual HepLorentzVector I4Momentum**::   hlv()** const = CLHEP **HepLorentzVector**

- virtual const bool IParticle::   **hasCharge ()** const = to **check if charge** information is available
- virtual const ChargeType IParticle::   **charge()** const = **returns charge** as a typedef ChargeType
- virtual const PDG::pidType IParticle::   **pdgId()** const = return enum indicating **particle id**

PDG Namespace:
PDG:: e_minus == 11 ,PDG:: e_plus == -11; PDG:: mu_minus == 13 ; PDG:: mu_plus == -13; PDG:: Z0 == 23; PDG::Higgs0 == 25;

HepLorentzVector class is for Lorentz vectors consisting of *Hep3Vector* and *HepDouble* components. Lorentz transformations (rotations and boosts) of these vectors are perfomed by multiplying with objects of *HepLorentzRotation* class.

- inline HepDouble   **m()** const = **invariant mass** (the rest mass of the system in the central of mass system).Can be used to add up the vectors of the particles and then to use m() as the mass of the particle generating the system

Electron class("ElectronCollection"):
- unsigned int   **author ()** const = electron author-specific functions
- double parameter (**ElectronParameters:**:ParamDef x) const = predefined parameters (ex x=**Eoverp**)
- int   **isEM ()** const = the first 4 bits set to 0 means the particle is electromagnetic (electron)
- bool   **hasTrack ()** const

Muon class("MuidMuonCollection" or "StacoMuonCollection"):
- const double Analysis::Muon::   **matchChi2OverDoF** ()const = get the chi2 over number of DoF of the track match. Chi2 is a sign of a statistical method to **check that the muon is good.**

CompositeParticle class:
- void   **add** (const ParticleBase *p) = **to add stuff** to the composite particle
- void   **set_charge** (ChargeType x)
- void   **set_pdgId** (PDG::pidType x)

Combination class:
- Combination (COLL *coll, const unsigned int nElement) = **constructor**
- template<class CALLER, class OUT, class CRITERIA> bool   **goodOnes** (CALLER *caller, OUT &comb, CRITERIA criteria) = **get a combination** which passes a selection criteria

IAnalysisTools class:
- template<class COLL> bool   **matchR** (const INavigable4Momentum *t, COLL *coll, int &index, double &deltaR, const int pdg) const = find the closest (in R) element in a collection to an INavigable4Momentum

b) The Containers

A container is a collection (an array) of objects. In this case, the objects are instantiations of the particle classes, and within the containers they are called tracks.

I used in my analysis:
> the Electron Container
> the Muon Container

> ➢ the Monte Carlo Container (Truth Container)

The first two have the reconstructed values of particle parameters (this is done by the Athena algorithms) and the MC has the real values.

2.3 The structure of the root output file

AAN = Athena Aware NTuple (for example root leaves)

- *Collection Tree*: leaves in a tree (AAN's) with the number of particles
- *Electron*: histograms over the electrons parameters
- *Muon*: histograms over the muons parameters
- *MassReconstruction*: histograms with the invariant mass of the composite particles
- *MonteCarlo*: the same histograms from above, but for the Truth Continer (MonteCarlo)
- *Efficiency:* histograms over the pt (mass) ratio or deltaR between particles and their match in MC

2.4 How to read the histograms and the AANs

The histograms/AANs for particle parameters (class member methods like the pt() of an electron or the mass of the reconstructed Hep Lorentz vector), which take different values for each track in each event, are filled at each track so the number of entries is therefore equal to the total number of electrons analyzed. Each column height means how many total tracks in the N events have x value pt() , mass, etc.

The histograms/AANs for the number of particles per event are filled at the end of the loops, at each event, putting a block at the correspondent number of tracks for that event. The number of total blocks = the number of events N (usually 1000). Each column height means how many events have x value tracks. And since the mean=$\Sigma \# blocks\_per\_collumn * xvalue(\# tracks\_per\_ev) / entries$ and usually #entries=1000, then Total #particles = mean*1000.

CHAPTER 3. DATA ANALYSIS: HISTOGRAM INTERPRETATION

In this analysis I retrieve the Electron, Muon and MonteCarlo Containers *(sc=m_storeGate->retrieve(elecTES, m_electronContainerName);)* then the iterator methods I define are repeated the #events times and I browse the Electron and Muon Containers by doing a loop over each event:

> *ElectronContainer::const_iterator elecItr = elecTES->begin();*
> *ElectronContainer::const_iterator elecItrE = elecTES->end();*

> *for (; elecItr != elecItrE; ++elecItr) {...reconstructed particles histograms...}*

For each track I find a match within the MonteCarlo(Truth) Container:

bool findAMatch = m_analysisTools->matchR((*elecItr), truthContainer,index, deltaRMatch, (*elecItr)->pdgId());
*   if (findAMatch) {*
*   ...deltaRMatch histograms (efficiency study)...*
*   if (deltaRMatch < m_deltaRMatchCut){*
*   ...montecarlo histograms...*
*   }*
}

If particles match and the parameter difference between the reconstructed and the truth particles are small, we prove this way that the Athena reconstructing algorithms are good so the Atlas detector works well.

Terminology:
- fast particles = reconstructed particles that have pT() > pT cut;
- good particles = the particles from the MonteCarlo container found after matching

Kinematics

We consider a frame in which the z axis is the boost direction (the beam direction, the direction of the Lorentz transformation). Then:

Phi   = the azimuthal angle in polar coordinates of the particle vector;
Theta = the polar angle angle (tha angle with respect to the z axis);
pT   = the transversal momentum= p* sin (theta);
pL   = the longitudinal momentum= p* cos (theta);
η =   pseudorapidity $= -\ln\tan(\theta/2)$;

$$pT = \sqrt{px^2 + py^2} \; ; p'T = pT;$$

$pL=pz;\ p'L = \gamma(pL - \beta E)\,;\ E' = \gamma(E - \beta pL)\,;$

The invariant mass of N particles:
- $m^2 = E^2 - p^2,$ where $E = \Sigma\ Ei,\ p = \Sigma pi.$
- it has the same value in any reference frame, since $E'^2 - p'^2 = E^2 - p^2.$
- for a single particle is equal to it's rest mass.
- to find the inv mass of the decaying higgs, evaluate the inv mass of the 4 leptons , most easily within the centre-of-mass frame, where: $\sum p' = 0 \ \Rightarrow\ \sum E' = inv\_mass$

Why pT is preferred for histograms rather the pL:
The pL of the Higgs comes from the asymmetry of the Parton components and after the decay it transfers to the mass and pL of the 4 leptons. The pT however of the Higgs is 0, so the pT of the 4 leptons is given just by the mass of the Higgs (by the QCD).
Also, because the energy and momentum are conserved, the presence of the invisible particles (the neutrinos) can be noticed only by the difference in pT between the 4l and the H. These are called missing ET's.
In addition, the transversal components of the position and of the speed (momentum) do not change under a Lorentz transformation.
Therefore it is more convenient to do a histogram over the pT and not the pL.

The pT cut:
It is mandatory (10GeV) to eliminate background, because purely hadronic processes are possible only at high energies and at small energies the particles decay more often in non meaningful channels which deviate from the main H -- >4leptons channel.

The pT measuring:
The pT is measured by applying a magnetic field inside the detector which bends the tracks, then the radius is measured and the speed is obtained through the Lorentz force formula. High pT particles have almost no bending.

Particle tracks of ZZ-->4 leptons decay
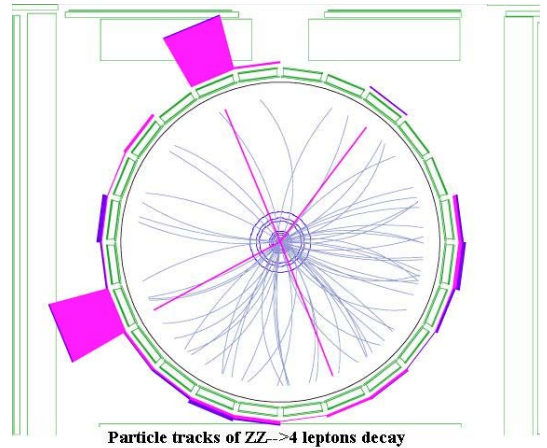
3.1 Kinematical variables histograms

The procedure was to retrieve the Electron and Muon
Containers, loop through each track in the events and plot the meaningful kinematical variables: pT, eta, EoverP (for electrons), chi2 (for muons), the number of tracks per event, the number of good matches.

In the same time I retrieve the Monte Carlo Container (the Truth Container), which has the true variables. The Monte Carlo Container does not exist in the analysis of real processes at the Atlas detector, just in software simulations.

There are 2 ways to use the MC Container: at each reconstructed particle find a match in the MC Container with the matchR() method or just browse (loop) through the container. At matching you retrieve the

deltaRmatch, which is a sign for the degree of the matching. If deltaRmatch is small => the Athena reconstruction works well.

Cuts used:
*For electrons & muons* → pT>10GeV, |eta|<2.5 (the TileCalorimeter can handle only small eta tracks)
For electrons → isEm divisible by 16, hasTrack() = true (signd that the reconstruction is good)
*For good matches* → matchR() =true, deltaRMatch <0.2

Regarding the muons, the Atlas physicists are not yet convinced which container to use: Muid or Staco. I do histograms for both of them. Theoretically Muid should reconstruct better than Staco.

a) Electrons

In the following histograms I overlapped the electrons' variables from the Electron Container and from the Monte Carlo Container matching particles.
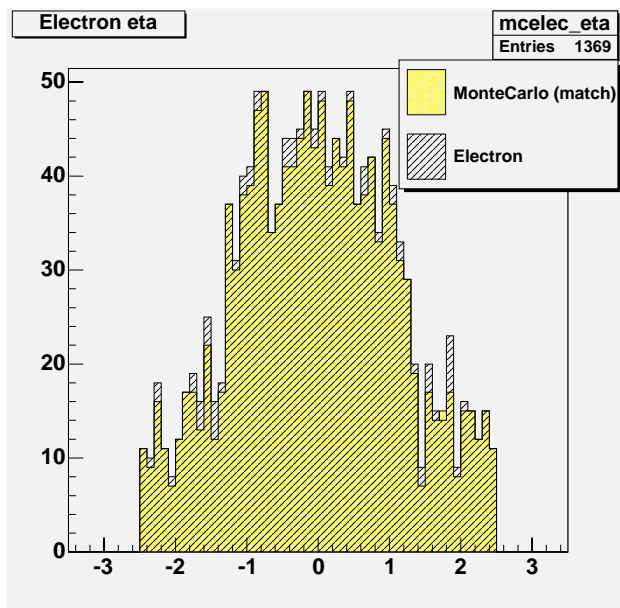
eta (Electron Container vs MonteCarlo Container)
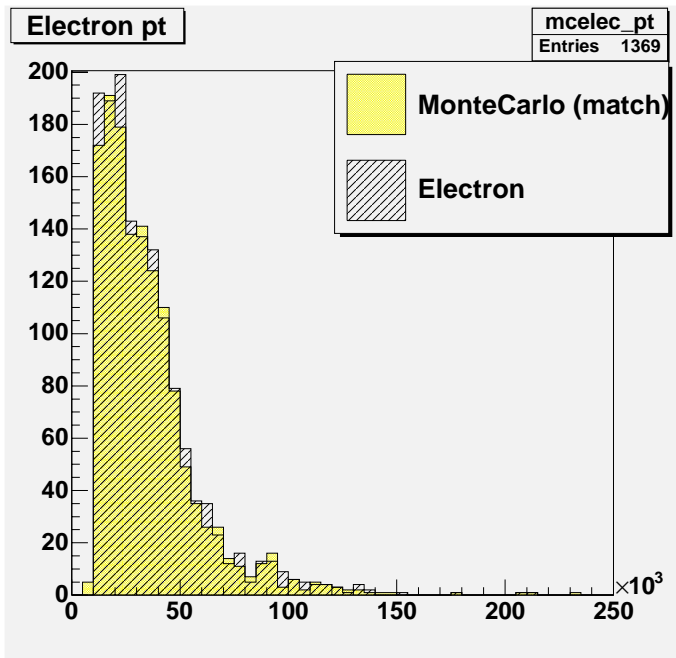
pt (Electron Container vs MonteCarlo Container)
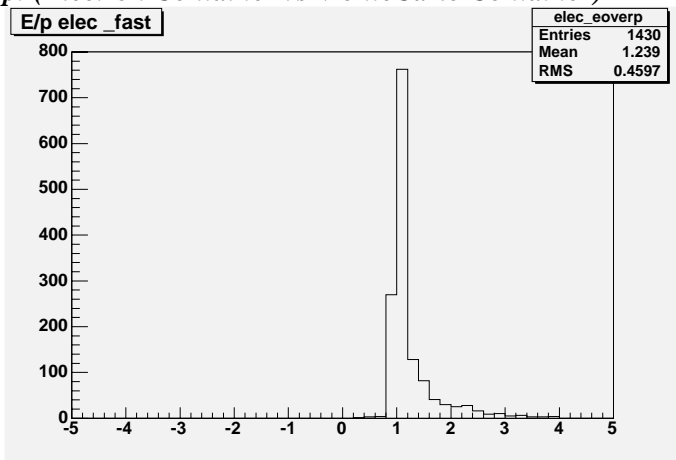
eoverp (Electron Container) – Eoverp=1 in the MonteCarlo Container

b) Muons
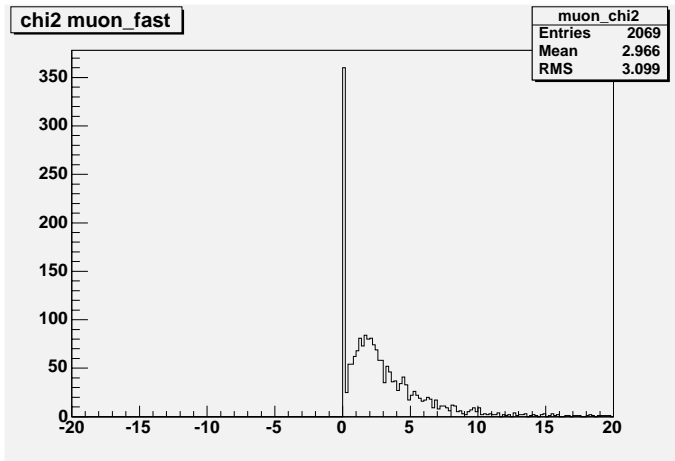
chi2(Muid Container) – there is no chi2 member in the TruthContainer class
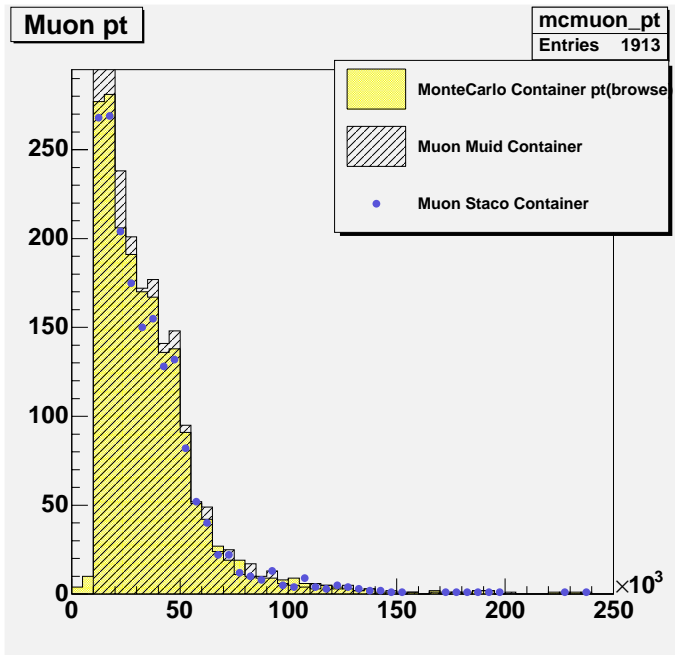
pt (Muid vs Staco vs MonteCarlo Container-by browsing)
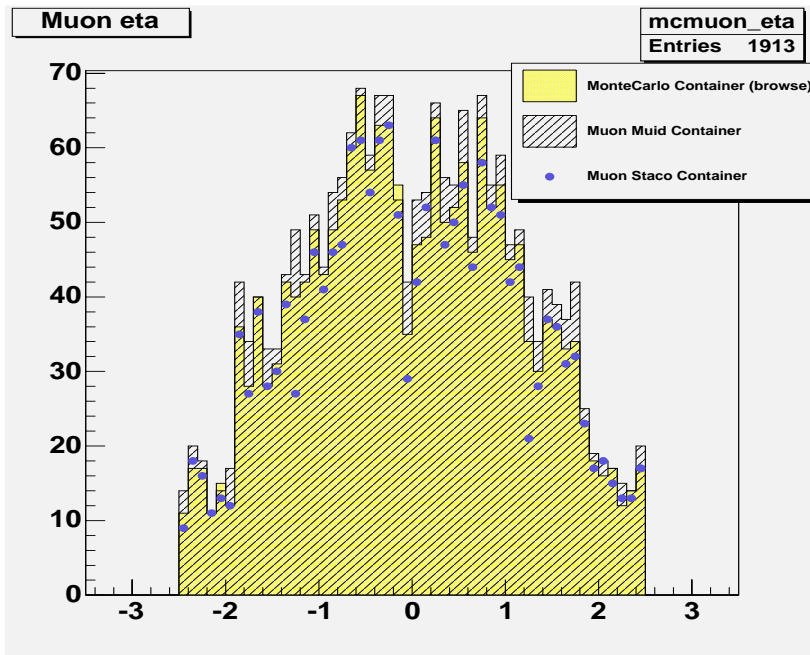
eta (Muid vsStaco vs MonteCarlo Container)

From these histograms it cannot be deduced precisely which container is better, muid or staco. Better results will be at the mass reconstructions.

3.2 Mass reconstruction histograms

Among the Z →2 leptons decays, the highest Branching Ratios is at Z →e+e- and Z →mu+mu- => the combinations analyzed from the AOD were H→4e, H→4mu, H→2e2mu.

The method is to loop through the containers and to do combinations with selections of 2e, 2mu (for Z reconstruction) 4e, 4mu, 2e2mu (for H reconstruction):

AnalysisUtils::Combination<const MuonContainer> combmuons(muonTES,2); //def of combination function;

   muonVect muonPair;//muonVect is a 2 elements array
 .....
   while (combmuons.goodOnes(this, muonPair, selectMuons)) { //the loop; goodOnes assures that the selectMuons function is satisfied

    /// create the Z candidate as a composite of 2 muons
.....
}

The selection functions check the cuts already mentioned and that Nb positive charges = Nb negative.

Add the HEP Lorentz vectors, and plot the invariant mass => should have peaks at mZ (91 GeV) and mHiggs (130GeV). The other Z mass is <91GeV, offshell, so that mZ1+mZ2 <= 130GeV. Otherwise the Z would be virtual.

13
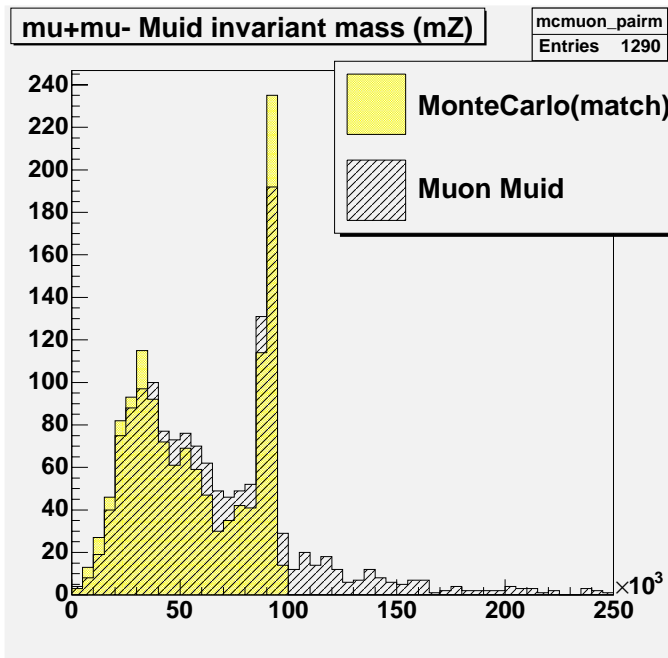
Statistically we expect Z decays equally to ee & mumu => Nb 4e =Nb 4mu = 0.5 Nb 2e2mu.

By analyzing a mass reconstruction histogram and searching for peaks we can deduce new physics: new heavy objects like the Higgs boson.
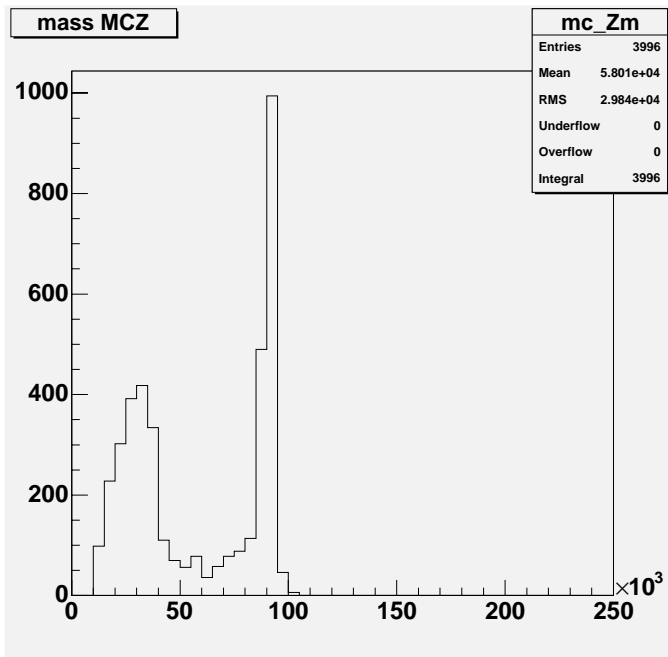
a) Z reconstruction
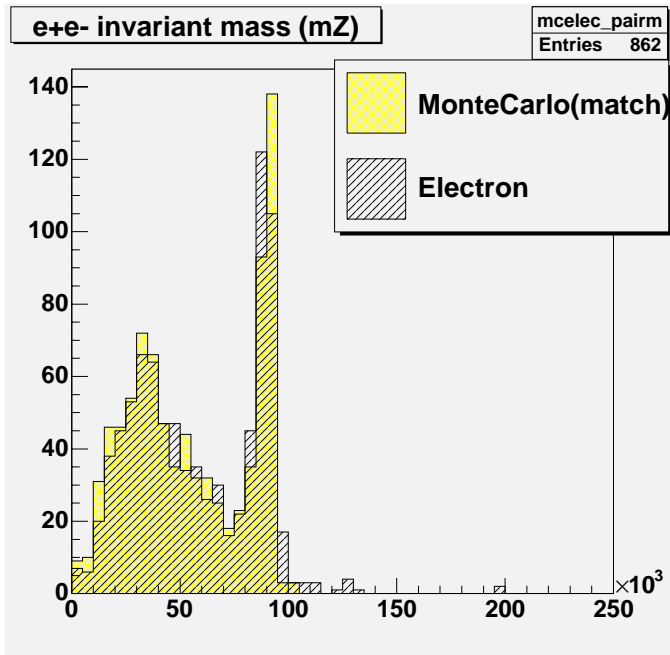
invariant mass of the e+e- composite particle

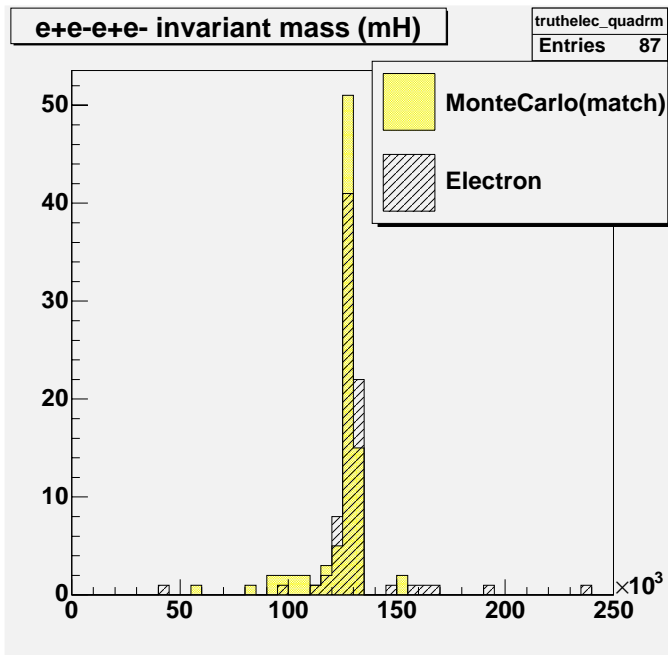*invariant mass of the mu+mu- composite particle*
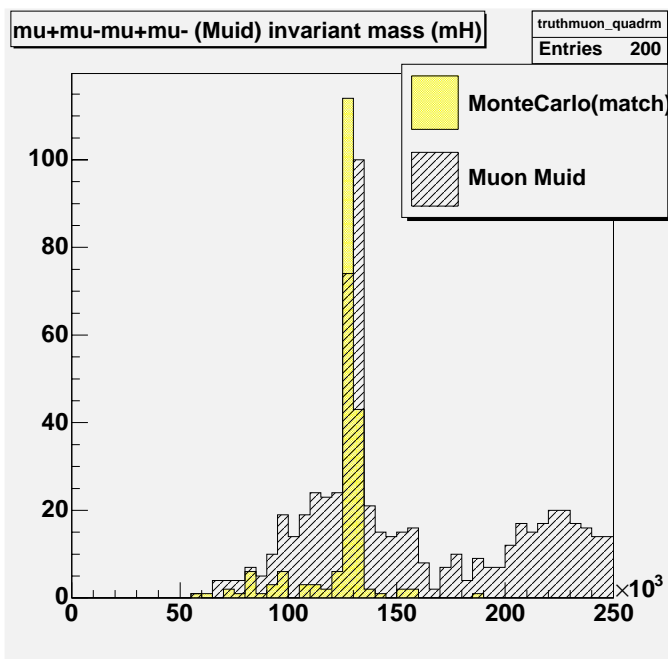

mass of the Z's in the Monte Carlo Container

It can be seen one peak at approx 90GeV and one peak at approx 30GeV.
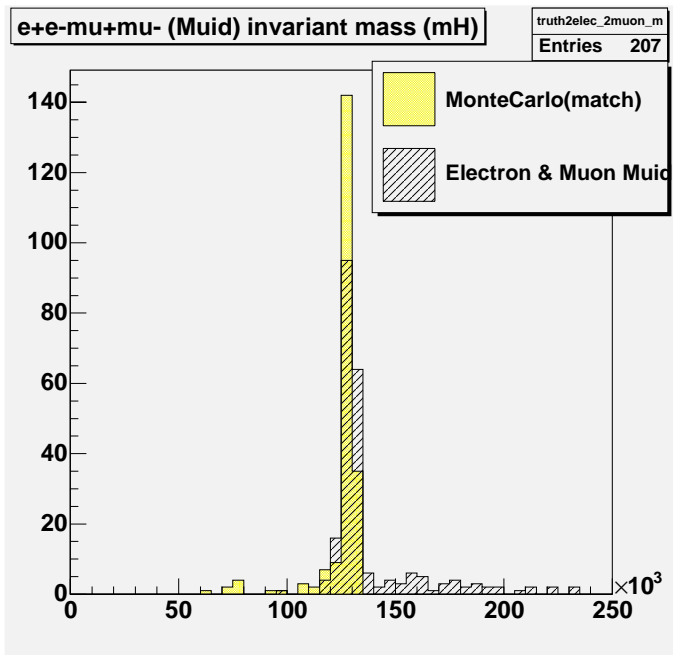
b) Higgs reconstruction

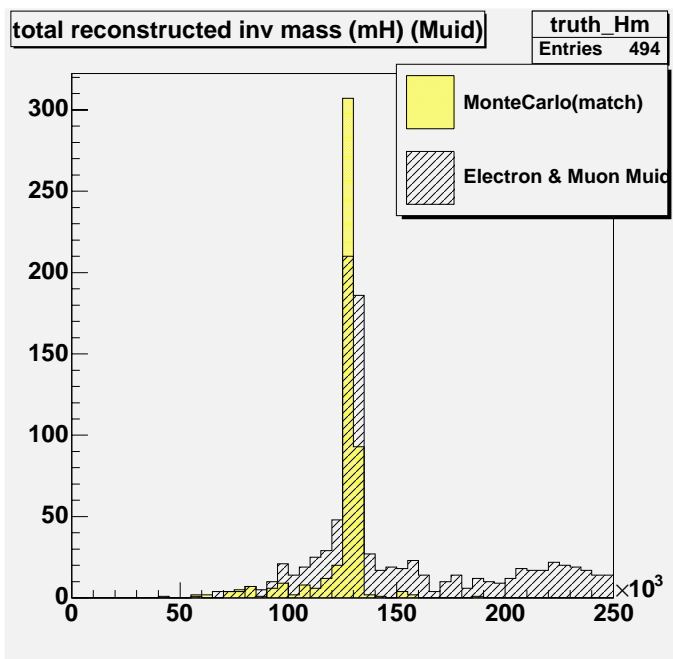**Using the Muon Muid Container**
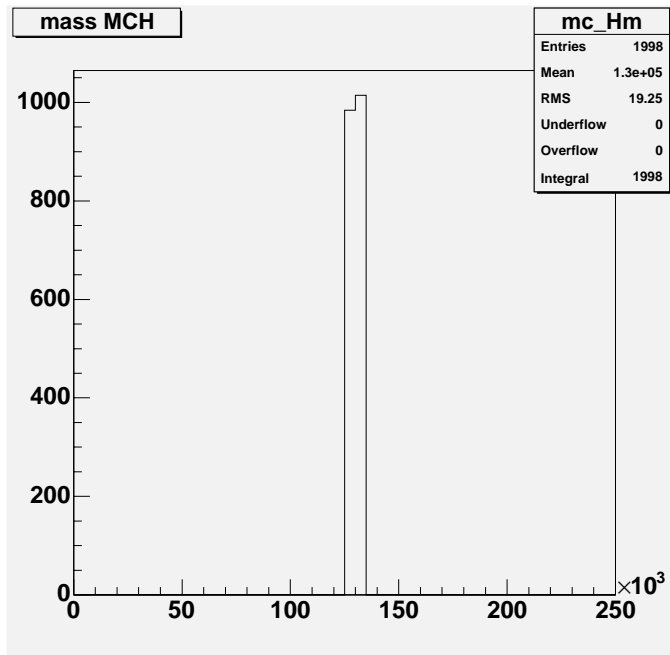
invariant mass of the 4e composite particle

invariant mass of the 4mu composite particle

invariant mass of the 2e2mu composite particle

invariant mass of the composite particle from all 4 leptons combinations

mass of the H's in the Monte Carlo Container

In the MC Container by browsing with the pdgId()=23 condition we find ~4000 Z's but the reconstructed Z's from the matched electrons and muons is ~865+1500 ~=1365. Possible causes:
- Z also decays into neutrinos (invisible for the detector => missing et particles)
- Not all the true electrons and muons from the MC Container are reconstructed, because of the pt cut at the electron and muon iterators or because the reconstruction doesn't work well.

STILL TO DO: Browse the Monte Carlo Container and count the numbers of electrons (identifying by |pdgID()| = 11) and muons (identifying by |pdgID()| = 13), without imposing the pt cut.

There are 1998 Higgs in the MC Container(by browsing). They all decay into 3996 Z's with ratio 1:2.

The Higgs mass reconstruction from 4e and 2e2mu looks very much alike the matched one. But the 4mu reconstruction has a lot of noise.

Possible causes:
- there are Kaons in the detector which decay into muons (so sources of muons outside the channel)
- hadronic showers (Kaons, Pions) punch through the Muon Calorimeter and decay into many particles that are badly identified as muons

Possible solution:
- I plotted the number of 4muon combination and I remarked 1 event with 450 combinations and 1 with 140 combinations => large number of tracks for that event.
- The explanation that the noise is not so large at the number of muons tracks per event is that those 2 events with overflow multiply its nb of tracks when performing combinations, so the noise is propagated.
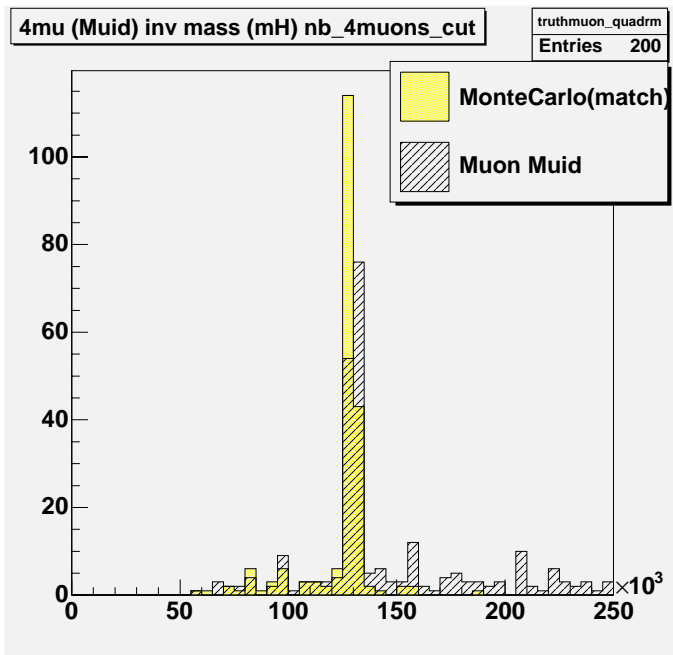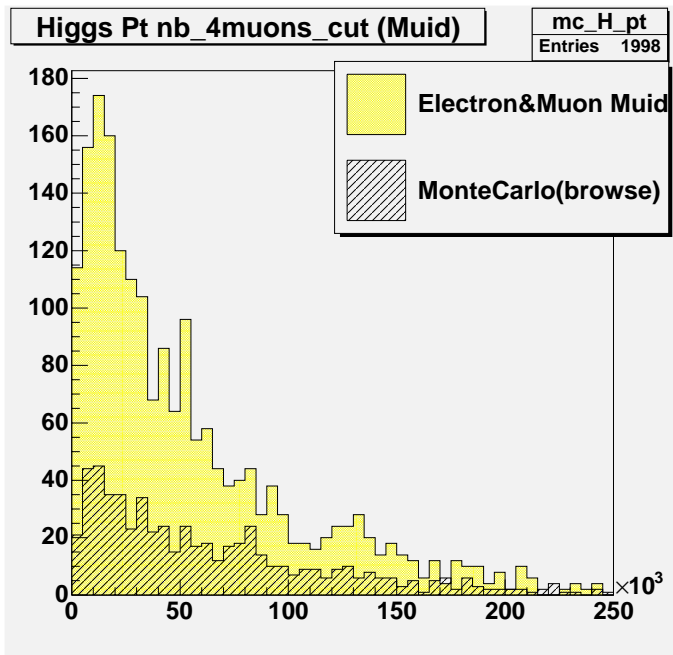
invariant mass of the 4mu composite particle, eliminating the 2 events with a high 4mu combinations.
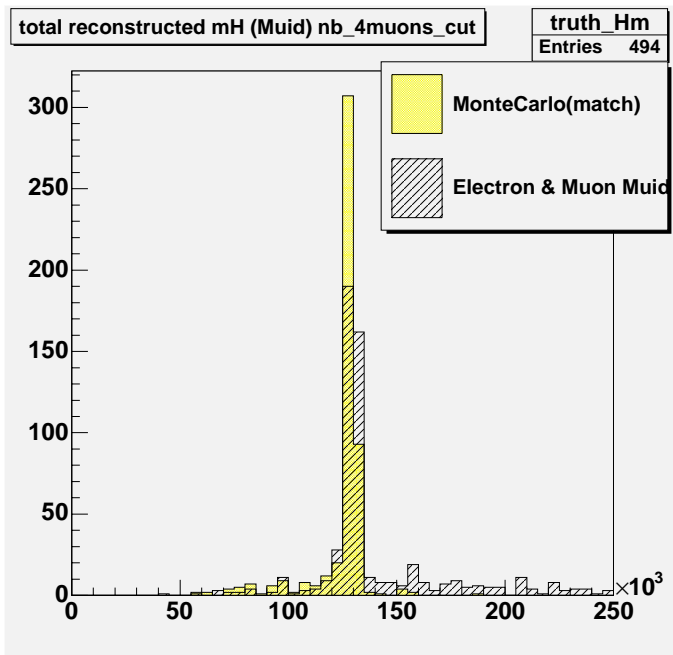
invariant mass of the composite particle from all 4 leptons combinations, eliminating the 2 events with a high 4mu combinations.

Missing pt particles:

Higgs pt from the reconstructed composite particle vs Higgs pt by browsing through the MC Container

The figure stresses out the missing et particles mentioned above, by showing this high pt difference between the MC Higgs and the reconstructed Higgs. The reason why I plotted the pt to shoe this -> *see Kinematics.*

Using the Muon Staco Container
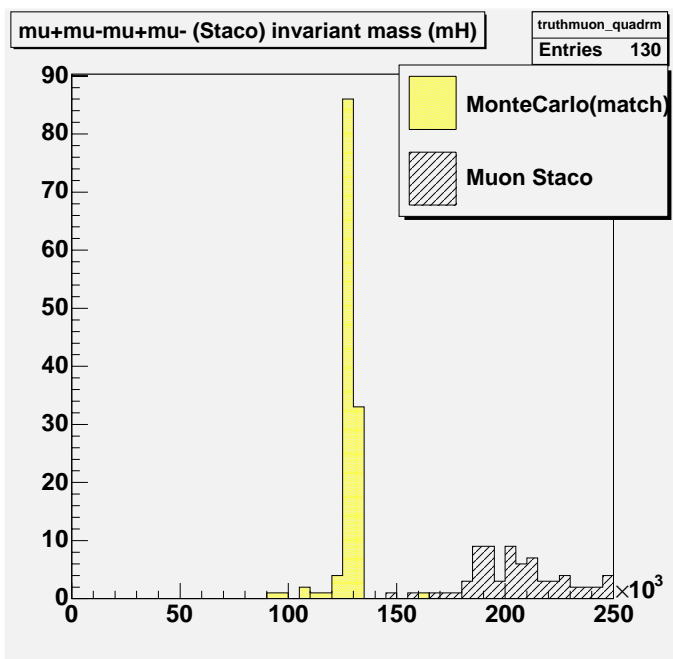
*invariant mass of the 4mu composite particle*

It can be seen that the Higgs is very badly reconstructed, with no 130GeV values. And a cut to keep a 130 GeV peak would just eliminate all the muons ➔ **Muid Container has a much better efficiency in reconstruction than the Staco Container.**

c) Numbers overall analysis. Efficiency study

NElectrons -> 1430

NElectronsGoodMatch ->1369

N2Electrons -> 866

N2ElectronsGoodMatch ->865

N4Electrons -> 87

N4ElectronsGoodMatch -> 87

TruthZ's -> 3996

TruthH's ->1998

NMuons -> 2048

NMuonsGoodMatch ->1913

N2Muons ->1525

N2Muons_ n4muonscut -> 1446

N2MuonsGoodMatch_ n4muonscut ->1284

N4Muons -> 879

N4Muons_ n4muonscut -> 288 (after cutting the 2ev)

N4MuonsGoodMatch_cut -> 200

Conclusions:

- 2*N2Leptons > NLeptons because in the H->ZZ->l+l-l+l- decay we have an overflow of 2leptons combinations: 1st lepton with 2nd lepton, 1st lepton with 4th lepton => more than in reality.
- N4Muons_ n4muonscut ≠ N4Electrons, disobeying the statistics.
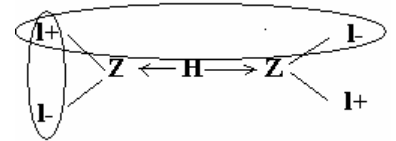
*Efficiency:*

- The efficiency tool is deltaRMatch = $\sqrt{\delta\eta^2 + \delta\varphi^2}$. It has nothing to do with the pt but it is obtained after searching for a match in the MC Container and I used it for selecting the matches, because the findAMatch=matchR(…) to be true is not enough: so I used the cut deltaRMatch<0.2.
- For the efficiency I plotted the deltaRMatch, ratios and differences between reconstructed and truth values ➔
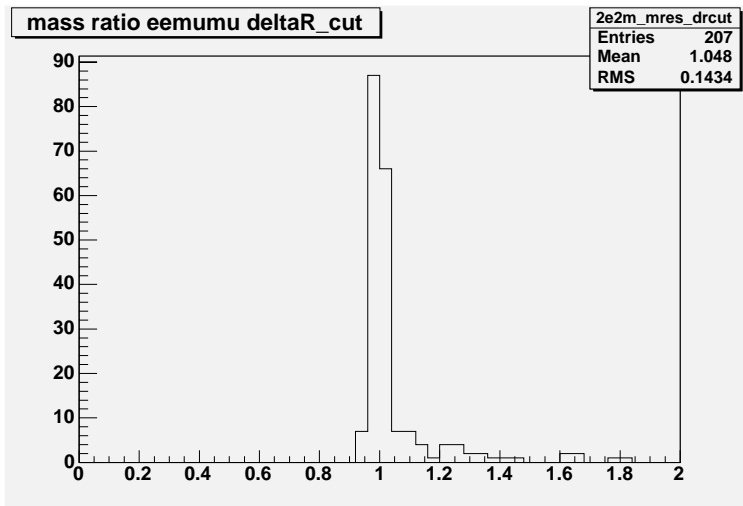
Ratio masses between reconstructed Higgs from 2e2mu and reconstructed Higgs from the matched 2e2mu
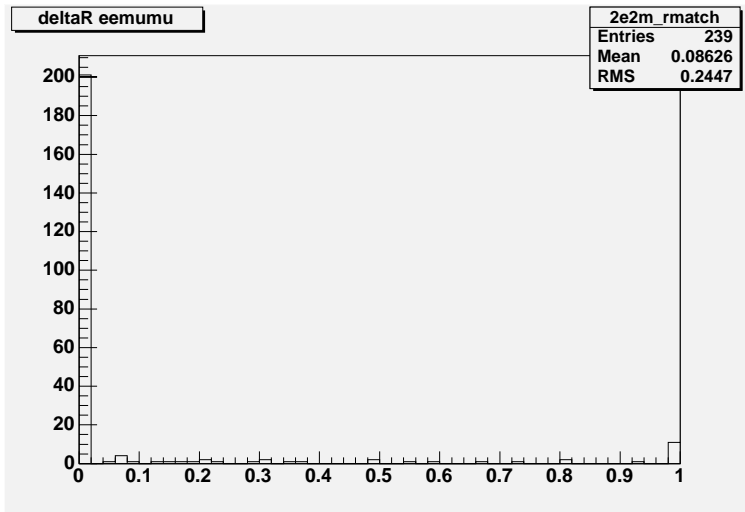
deltaRmatch between reconstructed Higgs from 2e2mu and reconstructed Higgs from the matched 2e2mu

Bibliography

1. R. Bock, H. Grote, *Data analysis techniques for high-energy physics experiments,* Cambridge University Press, 1990
2. Fabiola Gianotti, Phys.Rep 403-404 (2004), *Physics at the LHC*
3. B.R.Martin, G.Shaw, *Particle Physics,* The Manchester Physics Series, 1992
4. Particle Data Group, *Particle Physics Booklet*
5. D. Perkins, *Introduction to High Energy Physics,* 4th ed, Cambridge University Press, 2000
6. http://atlas.ch/ - *Atlas Page*
7. http://atlas.web.cern.ch/Atlas/GROUPS/SOFTWARE/OO/dist/10.0.4/InstallArea/doc/AnalysisUtils/html/classAnalysisUtils_1_1Combination.html
8. http://atlas.web.cern.ch/Atlas/GROUPS/SOFTWARE/OO/dist/latest/InstallArea/doc/CompositeParticleEvent/html//classCompositeParticle.html
9. http://atlas-computing.web.cern.ch/atlas-computing/links/buildDirectory/AtlasOffline/12.0.1/InstallArea/doc/EventKernel/html/namespacePDG.html#a458
10. http://atlas-computing.web.cern.ch/atlas-computing/links/buildDirectory/AtlasOffline/latest/InstallArea/doc/MuonIDEvent/html/classAnalysis_1_1Muon.html
11. http://wlav.web.cern.ch/wlav/athena/athask/tutorials.html - *Athena tutorials*
12. http://www.physics.usyd.edu.au/hienergy/PP2005_course.html#material – *Kinematics*
13. http://www.usatlas.bnl.gov/PAT/analysis_on_aod.html
14. http://wwwasd.web.cern.ch/wwwasd/lhc++/clhep/manual/RefGuide/Vector/HepLorentzVector.html
15. http://www-zeuthen.desy.de/~moenig/ - *Supervisor webpage*
16. http://www-zeuthen.desy.de/ILC/lectures/academic_training/ac_tr_2_bw.ps.gz - *Higgs material*
17. https://twiki.cern.ch/twiki/bin/view/Atlas/AtlasComputing
18. https://twiki.cern.ch/twiki/bin/view/Atlas/ComparatorDetails
19. https://twiki.cern.ch/twiki/bin/view/Atlas/MuonPerformance
20. https://twiki.cern.ch/twiki/bin/view/Atlas/PhysicsAnalysisTools
21. https://twiki.cern.ch/twiki/bin/view/Atlas/StoregateKeysForAOD
22. https://twiki.cern.ch/twiki/bin/view/Atlas/WorkBook
23. https://uimon.cern.ch/twiki/bin/view/Atlas/AnalysisExamples
24. http://www-zeuthen.desy.de/students/2006/lectures/summer_lecture.pdf - *Higgs material*
25. M.Veltman, CERN 97-05 (1997), *Reflections on the Higgs System*, Lectures given at the Academic Program of CERN 1996-1997