

DÖRDÜNCÜ AİLE KUARKLARININ
ANORMAL REZONANS ÜRETİMLERİ

Mehmet YILMAZ

Yüksek Lisans Tezi

Fizik Anabilim Dalı

Eylül - 2010

DÖRDÜNCÜ AİLE KUARKLARININ ANORMAL REZONANS ÜRETİMLERİ

Mehmet YILMAZ

Dumlupınar Üniversitesi
Lisansüstü Eğitim Öğretim ve sınav Yönetmeliđi Uyarınca
Fen Bilimleri Enstitüsü Fizik Anabilim Dalında
YÜKSEK LİSANS
Olarak Hazırlanmıştır.

Danışman : Doç. Dr. Hatice DURAN YILDIZ

Eylül – 2010

KABUL VE ONAY SAYFASI

Mehmet YILMAZ'ın Yüksek Lisans Tezi olarak hazırladığı “DÖRDÜNCÜ AİLE KUARKLARININ ANORMAL REZONANS ÜRETİMLERİ” başlıklı bu çalışma, jürimizce Dumlupınar Üniversitesi Lisansüstü Eğitim Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

/ / 2010

Üye: Doç. Dr. Hatice DURAN YILDIZ (Danışman)

Üye: Doç. Dr. Suat ÖZKORUCUKLU

Üye: Doç. Dr. Şükrü ŞENTÜRK

Fen Bilimleri Enstitüsün Yönetim Kurulu'nun/...../2010 gün ve sayılı kararıyla onaylanmıştır.

Prof. Dr. Atalay KÜÇÜKBURSA
Fen Bilimleri Enstitüsü Müdürü

DÖRDÜNCÜ AİLE KUARKLARININ ANORMAL REZONANS ÜRETİMLERİ

Mehmet YILMAZ

Fizik Bölümü, Yüksek Lisans Tezi, 2010

Tez Danışmanı: Doç. Dr. Hatice DURAN YILDIZ

ÖZET

Bu tezde Dördüncü Standart Model fermiyonlarından u_4 kuarkın üretim ve bozunum modlarını elde etmek için yapılan modelleme çalışmaları sunulmaktadır. 14 TeV 'lik kütle merkezi enerjisinde ve düşük luminositi $3 \times 10^{-31} \text{ cm}^{-2} \text{ s}^{-1}$ değerinde pp çarpışmasından u_4 'ün tekli üretimleri CompHEP'te sağlanmıştır. Bu tezdeki işlemler CERN-Linux4.7 işletim sisteminde fortran tabanlı program yazılarak elde edilmiştir.

CompHEP ile u_4 'ün tekli üretimleri için elde edilen sonuçlar $pp \rightarrow u_4 x \rightarrow WbWb$ kanalıyla modellenirken kütle değerleri olarak 200-800 GeV aralığı çalışılmıştır. LHC'de de u_4 tekli üretimleri çok miktarda bulunacaktır.

Ayrıca PGS, ATHENA gibi dedektör modelleme programları yardımıyla da simulasyon çalışmaları yapılarak dedektör modellemesi ele alınmıştır. Elde edilen sonuçların analizlerinin yapılabilmesi içinse grafiksel bir analiz programı olan ROOT ve yine görsel bir çizim programı olan GnuPlot programlarından yararlanılmıştır.

Anahtar Kelimeler: ATHENA, PGS, ROOT, Dördüncü SM Ailesi, ATLAS Deneyi, CompHEP, LHC

ANOMALOUS REZONANS PRODUCTION OF THE FOURT FAMILY QUARKS

Mehmet YILMAZ

Physics Department, M. S. Thesis, 2010

Thesis Supervisor: Assoc. Prof. Dr. Hatice DURAN YILDIZ

SUMMARY

In this thesis, simulation studies is presented for decay and production modes of the Standard Model fourth family quark. Single production of the u_4 from the pp collision, CompHEP is utilized with 14 TeV center of mass energy and low luminosity $3 \times 10^{-31} \text{ cm}^{-2} \text{ s}^{-1}$. Calculations of this thesis is obtained by writing a fortran based program in CERN-Linux4.7 operational system.

Single production results for u_4 with CompHEP, the mass interval is chosen from 200-800 GeV while the simulation studies are done with the $pp \rightarrow u_4 x \rightarrow WbWb$ channel. Single production of the u_4 will be copiously produced at LHC.

On the other hand, dedector simulation studies are done with the help of the dedector modelling programs such as PGS, ATHENA. In order to analyze the results, both visual analysis program-ROOT and visual drawing program-GnuPlot are utilized.

Keywords: ATHENA, PGS, ROOT, Fourth SM Family, ATLAS Experiment, CompHEP, LHC,

TEŐEKKÜR

Çalıőmalarım süresince bilgi ve tecrübelerini benimle paylaşan ve desteęini hiçbir zaman esirgemeyen deęerli danıőman hocam Sayın Doç. Dr. Hatice DURAN YILDIZ'a teőekkür ediyorum.

Ayrıca maddi ve manevi olarak her zaman yanımda olan, hayatım boyunca emeklerini ödeyemeyeceęim çok kıymetli aileme tüm yüreęimle teőekkür ediyorum.

Son olarak sevgili arkadaşlarım Hatice Yılmaz, Metin Aydar'a ve emeęi geçen herkese teőekkür ediyorum.

Beyaz bir meleęe...

İÇİNDEKİLER DİZİNİ

	<u>Sayfa</u>
ÖZET	iii
SUMMARY	iv
ŞEKİLLER DİZİNİ.....	viii
ÇİZELGELER DİZİNİ	ix
SİMGELER VE KISALTMALAR DİZİNİ.....	x
1. GİRİŞ	1
2. PARÇACIK HIZLANDIRICILARI	2
2.1 Cern	4
2.2 Lhc	5
2.3 Atlas Dedektörü	7
2.3.1. İz Takip Edici	9
2.3.2. Elektromanyetik Kalorimetre	10
2.3.3. Hadronik Kalorimetre	11
2.3.4. Muon Kalorimetre	11
2.4 Cms	12
2.5 Lhcb	13
2.6 Alice	13
3. DEDEKTÖR MODELLEME	14
3.1.Pgs	14
3.2.Athena	16
3.3.Comphep	19
3.4.Root	20
4. DEDEKTÖR MODELLEME UYGULAMALARI.....	28
4.1 PGS ile Modellemeler	28
4.2 ATHENA ile Uygulamalar	43
4.3 COMPHEP ile $pp \rightarrow t\bar{t}$ bozunumu için yapılan Modelleme Çalışmaları	49
5. SONUÇLAR VE TARTIŞMA	56
6. KAYNAKLAR DİZİNİ	57

ŞEKİLLER DİZİNİ

<u>Sekil</u>	<u>Sayfa</u>
2.1 Hızlanan parçacığın yönlendirilmesi.....	3
2.2 LHC'deki 4 ana deney platformu.....	6
2.3 Dedektör içindeki eta sınırları.....	8
2.4 ATLAS dedektörü katmanları.....	8
2.5 Yüklü bir Parçacığın yörüngesi.....	10
2.6 ATLAS Dedektöründeki Kalorimetreler.....	12
3.1 Koni temelli jet görüntüsü.....	16
3.2 Histogram örnek görüntüsü.....	21
3.3 $\tan(x)/(x^2)$ fonksiyonu	22
3.4 Root işlevselliği.....	23
3.5 TCanvas uygulaması.....	24
3.6 Boyutsal grafik gösterim örnekleri1.....	26
3.7 Boyutsal grafik gösterim örnekleri2.....	27
4.1 pythia_events_bB.hep.root kütüğü için grafikler.....	32
4.2 pythia_events_cC.hep.root kütüğü için grafikler.....	33
4.3 pythia_events_dD.hep.root kütüğü için grafikler.....	34
4.4 pythia_events_gG.hep.root kütüğü için grafikler.....	35
4.5 pythia_events_sS.hep.root kütüğü için grafikler.....	36
4.6 pythia_events_uU.hep.root kütüğü için grafikler.....	37
4.7 bB_BTAG işaretleme oranları grafiği.....	39
4.8 cC_BTAG işaretleme oranları grafiği.....	39
4.9 dD_BTAG işaretleme oranları grafiği.....	40
4.10 gG_BTAG işaretleme oranları grafiği.....	40
4.11 sS_BTAG işaretleme oranları grafiği	41
4.12 uU_BTAG işaretleme oranları grafiği.....	41
4.13 J/psi->mumu sürecinin ROOT histogramları.....	49
4.14 Tesir kesitinin kütleyle ilişkin grafiği.....	52
4.15 Rapidity histogramı.....	53
4.16 Cosine histogramı.....	54
4.17 Transverse Momentum histogramı.....	54

ÇİZELGELER DİZİNİ

<u>Çizelge</u>	<u>Sayfa</u>
2.1. Kütlelerin denk enerjisi Tablosu	3
4.1. PGS girdi kartı	29
4.2. PGS olay üretim zinciri.....	30
4.3. LHCO kütüğünün içeriği.....	30
4.4. Tesir kesiti hesaplamaları tablosu	51
4.5. Rezonans Tesir kesiti (pb) ve t' kuarkın bozunma genişlikleri $\kappa/\Lambda=1 \text{ TeV}^{-1}$	52
4.6. Rezonans Tesir kesiti (pb) ve t' kuarkın bozunma genişlikleri $\kappa/\Lambda=0.1 \text{ TeV}^{-1}$	52

SİMGELER ve KISALTMALAR DİZİNİ

<u>Simger</u>	<u>Açıklama</u>
L	Lagrangian, ışınlık
η	skaler Higgs alanının boşluk beklenen değeri
L_{int}	İntegrali alınmış ışınlık
ss	son durum parçacıkları için istatistiksel (statistical significance) değer
σ_s	pb olarak sinyalin tesir kesiti
σ_B	pb olarak fonun tesir kesiti
E_T	Dik enerji
E_T^{init}	İlk dik enerji
P_T	Dik momentum
<u>Kısaltma</u>	<u>Açıklama</u>
SM	Standart Model
SUSY	Süper Simetri
LHC	Büyük Hadron Çarpıştırıcısı
CERN	Avrupa Nükleer Araştırma Organizasyonu
ATLAS	Toroidal Büyük Hadron Çarpıştırıcısı Aleti
FERMILAB	Uluslar arası Tevatron Hızlandırıcı Fermi Laboravtuarı
QED	Kuantum Elektrodinamik
QCD	Kuantum Kromodinamik
SLAC	Stanford Lineer Hızlandırıcısı
FCNC	Çeşni Değiştiren Nötral Kanal
ALICE	Büyük İyon Çarpıştırıcısı
LHCb	LHC-beauty
CMS	Compact Müon Selenoid
LEP	Büyük Elektron Pozitron Çarpıştırıcısı

1.GİRİŞ

Bilim insanları, atomaltı parçacıklar için günümüze kadar ulaşılmış olan deneysel ve teorik verileri bir çerçeve altında toplamışlardır. Birtakım eksikliklerine rağmen genel anlamda doğru olduğu kabul edilen bu çerçeveye Standart Model (SM) adı verilmiştir [1]. Bütün kuramlar gibi, SM’de başarı ve başarısızlığı kuramsal hesaplarla deneysel ölçümler karşılaştırılarak bulunur [2]. Ancak bu kuram hali hazırda ne kadar başarılı olsa da hala açıklık getiremediği bazı sorular vardır. Bu sorulara cevap arayan bilim insanları sağ-sol simetri modelleri ve büyük birleşim teorileri gibi yeni kuramlar ortaya atmışlardır. SM’deki eksikleri gidermek için önerilen en önemli öngörülerden biri Dördüncü SM ailesinin varlığıdır [3].

Dördüncü ailenin varlığıyla SM’deki fermiyon aile sayısı ve Higgs’in gözenebilirliği modelleme yoluyla ispat edilmiştir. Ancak dördüncü aile çalışmalarının yapılabilmesi için yüksek enerjilerde çalışma olanağı sunan bir hızlandırıcıya ihtiyaç vardır. Bu anlamda düşünülebilecek en iyi hızlandırıcı CERN’deki Büyük Hadron Çarpıştırıcısı (Large Hadron Collider-LHC) dır.

Bu tezde ATLAS deneyi hızlı dedektör modelleme programı-PGS, tam dedektör modelleme programının-ATHENA çalıştırılması ve dördüncü SM fermiyonlarından u_4 ün (t') tekli üretimi dedektör modellemesi katılmadan CompHEP modelleme programı ile çalışılmıştır.

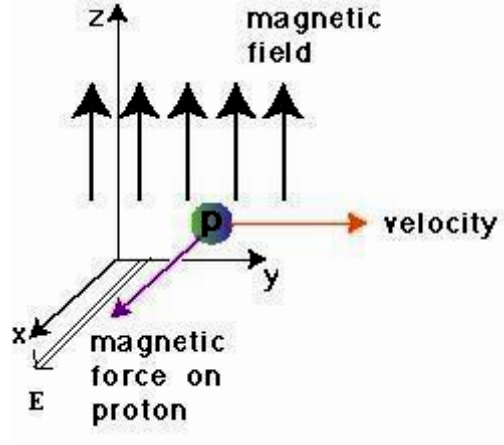
Tezin birinci bölümünde genel yüksek enerji fiziği hakkında bilgi verilmekte, ikinci bölümde ise Parçacık Hızlandırıcıları, CERN-LHC ve Atlas dedektörünün çalışma yapısından bahsedilmektedir. Üçüncü bölümde, hızlı ve tam dedektör modelleme programlarının çalışma ve yapılarından söz edilirken dördüncü bölümde bu modelleme programlarına ilişkin uygulamalar ve çalışılan konular anlatılmakta, PGS (Pretty Good Simulation), ATHENA ve CompHEP modelleme programları ile yapılan modelleme ve analizler yorumlanmaktadır. Son bölümde ise yapılan çalışmaların sonuçları tartışılmaktadır.

2. PARÇACIK HIZLANDIRICILARI

Geçmişten günümüze kadar insanoğlu etrafında gördüğü her nesneyi inceleyip algılamaya çalışmıştır. Önce etrafında gördüğü nesnelere bakmış sonra onlara dokunmuş, koklamış, bütün bunları yaparken incelediği nesneye olan merakı daha da artmıştır. Ancak yakın tarihte artık dış yapısını gördüğü nesneyi değil, o nesneyi oluşturan içyapıyı ve bu yapıyı bir arada tutan şeyin ne olduğunu merak etmeye başlamıştır. Bizler bugün bile göremediğimiz maddeleri merak ederiz ve bir maddenin içini görmek istediğimizde ki bu maddenin elle tutulabilir ve gözle görülebilir olduğunu kabul edersek onu parçalarına ayırmamız gerektiğini biliriz. Bu amaçla o cismi ya bir yere çarparak parçalarına ayırırız ya da başka bir cisimle çarpıştırarak parçalarız.

Ancak elle tutamadığımız gözle göremediğimiz maddeler için ne yaparız?

Böyle küçük yapıdaki parçacıkları incelemek ve içlerini görebilmek için hızlandırıcılara ihtiyaç duyarız. Parçacık hızlandırıcıları, parçacıkları yüksek enerjilere ulaştırmak için elektrik alanları, yönlendirmek için manyetik alanları kullanırlar. Bir hızlandırıcının amacı maddeyle veya başka bir parçacıkla etkileşmesi için yüksek enerjili parçacıklar üretmektir. Lineer ve dairesel olmak üzere iki temel tip hızlandırıcı vardır. Parçacıkların düz bir hat boyunca hızlandırıldığı sisteme yüksek enerjili lineer hızlandırıcılar (Linac) denir. Hızlandırılan parçacıklara yeterli enerjiye ulaşana kadar dairesel hareket yaptırılarak yüksek enerjilere ulaşılan sisteme ise dairesel hızlandırıcılar denilmektedir. Hızlandırılan parçacıklar dipol mıknatısların yardımıyla dairesel yörüngede tutulurlar. Dairesel hızlandırıcılar gelişim sırasına göre siklotron, senkrotron, siklosenkrotron olarak sıralanabilir. Dünya'daki hızlandırıcı laboratuvarları arasında en önemlileri; CERN (Cenevre), Fermilab-FNAL (Chicago), DESY (Hamburg), ve SLAC (California) şeklinde verilebilir. Lineer hızlandırıcılara göre dairesel hızlandırıcıların en büyük avantajı parçacık demetlerini tekrar tekrar kullanabilmesidir. Ancak sinkrotron radyasyonu sebebiyle enerji kaybetmeleri de bir dezavantajdır.



Şekil 2.1 Hızlanan parçacığın yönlendirilmesi

Parçacıkları yüksek enerjilere çıkarabilmek için hızlandırıcılar kullanılmak zorundadır. Fakat neden yüksek enerjilere ihtiyaç duyulduğu şu şekilde açıklanabilir.

İlk sebep olarak, yüksek enerjilere ulaşıldığında, düşük enerjilerde gözlemlenen karmaşanın çoğundan kurtulmuş oluruz.

Bir başka neden ise aranan yeni parçacıkların kütleleri arttıkça, onları oluşturmak için gerekli olan enerjinin de artması gerekliliği gösterilebilir. Aşağıda bazı kütle değerlerine karşılık gelen enerji değerleri verilmektedir.

Çizelge 2.1 Bazı özel kütle değerlerine karşı gelen atomaltı parçacıklar

0.000000511 TeV	Elektron
0.000938 TeV	Proton
0.08 TeV	W parçacığı
0.1 - 1.0 TeV	Higgs ve diğer yeni parçacıklar

Günümüzde bilinen en büyük dairesel hızlandırıcı CERN’de yapılan deneyde kullanılan LHC (Large Hadron Collider) büyük hadron çarpıştırıcısıdır.

2.1 Cern

CERN (Avrupa Nükleer Araştırma Merkezi), 1954 yılında Fransa-İsviçre sınırında kurulmuş, dünyanın en büyük parçacık fiziği araştırma merkezlerinden biridir.

2006 yılında CERN'in 20 tane üye ülkesi ve Türkiye'nin de dahil olduğu 6 tane gözlemci üye ülkesi vardı. Ancak Türkiye bildiğimiz üzere yeni bir atılımda bulunarak diğer bazı ülkelerle CERN'e tam üyelik konusunda büyük bir adım atmıştır. Türkiye ile CERN arasındaki ilişkinin çerçevesini belirleyen TAEK-CERN işbirliği anlaşması 14 Nisan 2008 tarihinde Cenevre'de imzalanmıştır. Türkiyenin CERN'e tam üyelik süreci ise 16 Mart 2009 tarihinde resmen başlamıştır [4]. Merkezde değişik ülkelerden 6500'ün üzerinde araştırmacı bilimsel çalışmalar yapmaktadır [5]. Zayıf etkileşimin ara taşıyıcı parçacıkları olarak bildiğimiz Z^0 ve $W^{+ -}$ bozonları CERN'de bulunmuştur. Bunun yanı sıra günümüzün en iyi haber alma araçlarından biri olan World-Wide Web (www), büyük ve dünyaya dağılmış kollaborasyonlar halinde çalışan parçacık fizikçilerinin hızlı ve kolay bir biçimde iletişimlerini sağlayıp hızlandırmak amacıyla CERN'de bir bilgisayar programcısı olan Tim Berners Lee'nin 1990 yılında "HTML" adlı bilgisayar dilini bulup geliştirmesiyle oluşmuştur [6]. Ancak www'den daha geniş kapsamlı yeni veri ve bilgi aktarma ve iletişim sistemi olan GRİD sistemi üzerinde çalışılmaktadır.

LHC deneylerinden gelen verilerin çözümlenmesine yardımcı olacak bilgi işleme ortamını hazırlamak LCG (LHC Computing Grid) LHC Hesaplama Grid'i geliştirilmiştir. GRID, bilgisayarların hesaplama ve veri depolama kaynaklarını internet üzerinden paylaşmak amacı ile oluşturulan bir servis olarak tanımlanabilir. GRID ile bilgisayarlar arasındaki basit haberleşmenin ötesine geçilerek küresel bilgisayar ağının büyük bir hesaplama kaynağına çevrilmesi hedeflenmektedir. Bu sistem yine CERN'de doğmuştur. Şu sıralarda Türkiye'de kendi GRID alt yapısını Türk Atom Enerjisi Kurumu (TAEK) bünyesinde kurmuş ve bu sistemi CERN'de ve ülkemizde yüksek enerji fiziği alanında çalışmalarda bulunan Türk bilim insanları için geliştirme çalışmaları içerisine girmiştir. GRID sistemi sayesinde çok büyük veri aktarımı yapılabilen ve bu veriler saklanabilmektedir. Ayrıca bilim insanlarının çok uzun süren veri hesaplama ve bu veriyi analiz etme çalışmalarını çok daha kısa sürelerde sonuçlandırmasını sağlamaktadır.

CERN'de yürütülen araştırmaların esas amacı maddenin yapısını ve maddeyi bir arada tutan kuvvetleri anlamaktır. İnsanlığın asırlardır yürüttüğü maddenin yapısını anlamak amaçlı büyük çabanın arenası bugün parçacık hızlandırıcılarıdır. Parçacık hızlandırıcılarında en yüksek enerjilere ve çarpışma sayılarına erişmek, çarpışmalardan çıkan çok sayıdaki parçacığı

algılayabilmek, mevcut teknolojinin sınırlarını zorlamaktadır. CERN, temel bilim arařtırmalarının yanında, yarının teknolojilerini geliřtirmekte de ok nemli bir rol oynamaktadır.

Süperiletken teknolojisinin CERN hızlandırıcıları sayesinde ilerlemesi, yeni temiz enerji kaynaklarının arařtırılması, temel bilimlerin ilerlemesi, yeni reaktör sistemlerinin geliřtirilmesi, bilgisayar teknolojisi, tıpta tedavi ve teřhis uygulamaları, yeni elementlerin bulunuşu en önde gelen arařtırmalardır. Paracık Fizięi arařtırmaları lazer fizięi, plazma fizięi, elektronik, telekomünikasyon, nanobilim, malzeme bilimi, nükleer tıp ve ışın tedavisi, biliřim teknolojisi (yazılım geliřtirme, bilgisayar mimarisi, bilgisayar aę bilimi v.s), savunma sanayii ve mühendislięin eřitli dallarındaki yeni geliřmelerin lokomotifidir. Ayrıca dięer nemli bir rnek ise hızlandırıcılar sayesinde ilk defa bir anti-madde rneęi olan antihidrojen yapımı gerekleřtirilmiřtir [7].

Bu kapsamda CERN arařtırma merkezi, temel bilimin en ileri saflarında yeni bilgi üretmeye alıřan bilim adamlarına teknolojinin izin verdięi en ileri deneysel olanakları sunmaktadır [7].

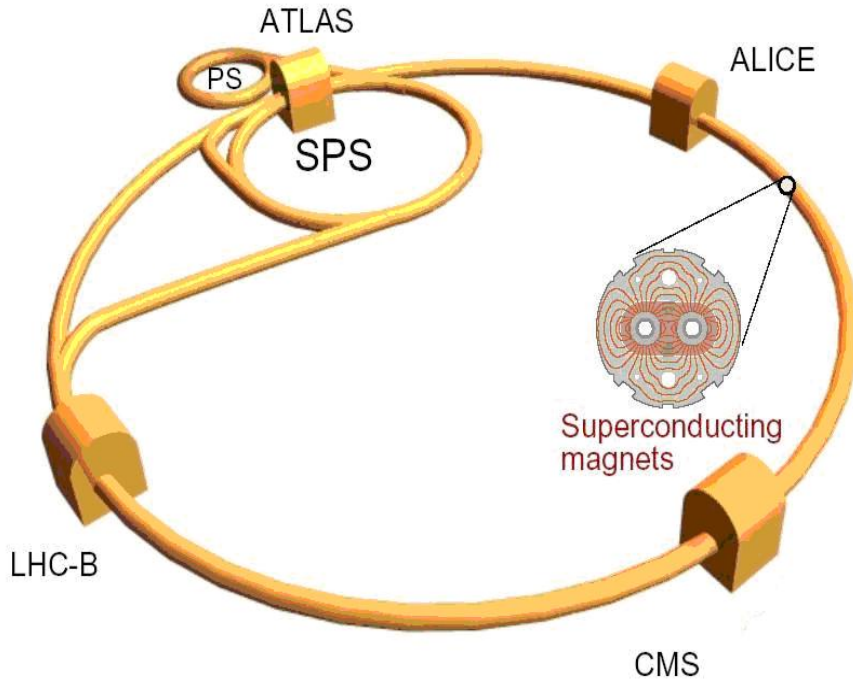
2.2 Lhc

LHC (Large Hadron Collider - Büyük Hadron arpıřtırıcısı) bugün bildięimiz en büyük dairesel hızlandırıcıdır. LHC, 2000 yılında faaliyeti sona eren LEP'in (Large Elektron-Positron - Büyük elektron-pozitron arpıřtırıcısı) yerine inřaa edilmiřtir [8]. evresi 27 kilometre ve yer yüzeyinden 100 m derinliktedir [9]. Halka biçimli bir tünel ierisine kuruludur. Tünelin apı yaklaşık olarak 3,8 m uzunluęundadır. Tünel İsvire-Fransa sınırında, her iki ülkenin toprak sınırı ierisinde yer almaktadır. LHC'de evrenin varoluşu ile ilgili bir takım bilgilere ulařmanın yanı sıra birok yeni fizięi keřfetmek ve Higgs bozonu hakkında yeni veriler elde etmek amalanmaktadır. Birbirine ters yönde iki proton demeti, SPS (Süper Proton Senkrotronu) adı verilen özel bir hızlandırıcı aracılıęı ile LHC halkası ierisine enjekte edilecektir. Proton demetleri 450 GeV (milyar elketronvolt) enerji düzeyinde halkaya alındıktan sonra 7 TeV (trilyon elektronvolt) düzeyine kadar hızlandırılacaktır. Iřık hızının yanıbařına kadar hızlandırılan ve ters yönlerde yol alan proton demetleri, birbirlerinin iinden saniyede 40 milyon kez geecekler. Demetlerin karřılıklı her geiřinde 20 proton-proton arpıřması olacaęı hesaplanmıřtır. Bu da saniyede 800 milyon arpıřma demektir. Ancak bunların arasında "ilgin" olayların sayısı ok az olacaktır. ünkü protonların oęu birbirini yalnızca sıyırtıp geecek, kafa kafaya arpıřmalar son derece seyrek görülecektir. Kafa kafaya arpıřmalarda ortaya yeni paracık ıkaranlar daha da az olacaktır. Sonuçta Higgs paracıęının ancak 10 trilyon

çarpışmadan birinde ortaya çıkabileceği hesaplanmaktadır. Yani, her saniye 800 milyon çarpışma gerçekleşse bile, bir günde yalnızca tek bir Higgs parçacığı ortaya çıkabilir [5]. Heyecanla beklenen dünyanın bu en büyük çarpıştırıcısının çalıştırılmasına kısa bir zaman önce başlandı. LHC'den ilk veriler alınmaya başlandığı sırada bir arıza meydana geldi ve bu büyük çarpıştırıcı arızanın giderilmesi için bir süreliğine tekrar kapatıldı.

Eylül 2009 tarihinde tekrar faaliyete geçen LHC'de parçacıkları gözlemlemek için 4 ana deney yer almaktadır. Bunlardan başka birçok küçük çaplı deneyde yer almaktadır. Dört büyük deney şekil 2.2 de gösterildiği gibi:

- ATLAS (A Large Toroidal LHC ApparatuS, Büyük Toroidsel LHC Cihazı)
- CMS (Compact Muon Solenoid, Kompakt Müon Sarmalı)
- ALICE (A Large Ion Collider Experiment, Büyük İyon Çarpıştırıcısı Deneyi)
- LHCb (A Large Hadron Collider beauty Experiment, b Fiziği İçin Büyük Hadron Çarpıştırıcısı)



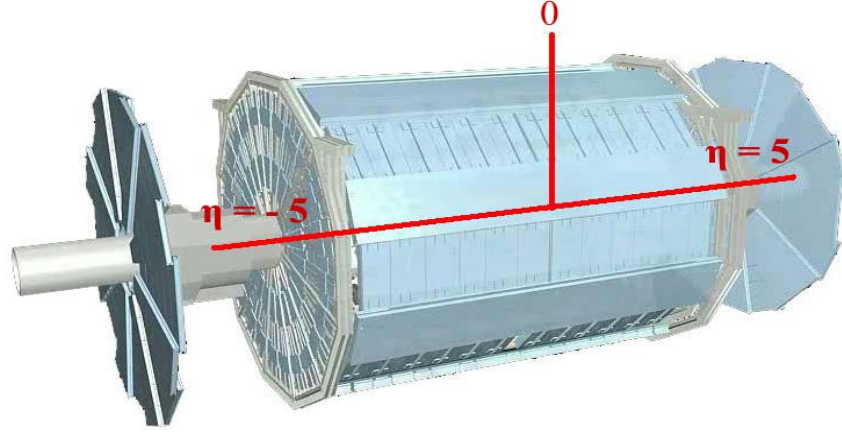
Şekil 2.2 LHC'deki 4 ana deney platformu

2.3. Atlas Dedektörü

Atlas, LHC deki genel amaçlı iki dedektörden biridir [10]. Atlas dedektörü 46 m uzunluğa, 12,5 m yarıçapa, 25 m genişliğe ve yaklaşık olarak 7000 tonluk bir ağırlığa sahiptir [11]. Geniş bir araştırma ve çalışma alanına sahip olan Atlas'ta Higgs Bozonu, Ekstra Boyutlar ve Parçacıklar, Karanlık Madde gibi cevap bekleyen konular araştırılmaktadır [10]. Bunların yanı sıra evrenimizi oluşturan temel kuvvetler ve maddenin temel yapısı araştırılacaktır [9]. Dedektörün iç yapısı, parçacık izleyiciler, ışık yayan kristaller ve yüz milyon veri kanalına bilgi akışı sağlayacak pek çok aygıt ve fiber kablolarla doludur [5]. Hızlandırılarak çarpıştırılan parçacıkların enerji ve momentumları denetlenebilecek ve ölçülebilecektir. Atlas dedektörü iç dedektör, kalorimetre, muon spektrometresi ve magnet sistemlerine sahiptir. Pseudorapidity parçacık ve deneysel yüksek enerji fiziğinde, rölativistik parçacıkların açısının, çarpışma eksenine bağlı olarak ifade edilmesinde kullanılan bir niceliktir. Atlas dedektörü için parçacık demet yönü z alınmak üzere, z-ekseni ile olan kutup açısı θ ve y-ekseni ile olan azimuthal açı ϕ olmak üzere iki açı tanımlanır. Genelde kutup açısı yerine pseudorapidity (η) kullanılır ve değeri, $-5 \leq \eta \leq 5$ aralığındadır ($\eta = -\ln(\tan \theta/2)$).

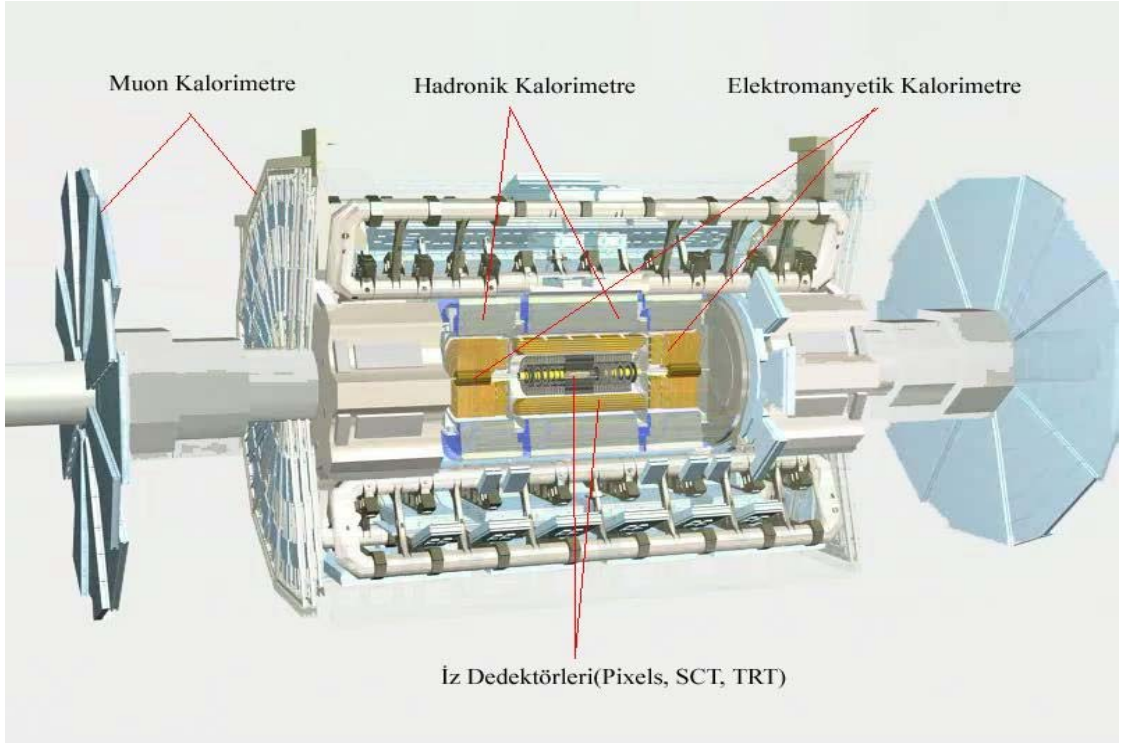
Pseudorapidity, y rapiditesinin bir yaklaşımıdır ve şu şekilde verilir: $y = (1/2) \ln((E+P_z)/(E-P_z))$. Burada E ve P sırasıyla parçacığın enerji ve momentumudur. Pseudorapidity, rapiditenin $P \gg m$ ve $\theta \gg 1/\gamma$ için çok iyi bir yaklaşımıdır.

Dedektörün açısal bağımlılığı önem taşır. Silindirik şeklindeki Atlas dedektörünün bir ekseninden diğer eksenine gidiş eta taramasına göre aşağıdaki şekil 2.3 de de gösterildiği gibi $-5 \leftrightarrow 5$ aralığındadır. Fakat 5 ve 4'lere yakın değerlerde ancak çok yüksek enerjili parçacıklar bulunur. Orta bölgelerde ise enerjisi çok yüksek olmayan parçacıkların geniş bir birikimi yer almaktadır. $|\eta| \leq 2.5$ Özelliğine sahip olan parçacıklara merkezi parçacıklar ve bunların dağılım gösterdiği bölgeye de merkezi bölge denir.



Şekil 2.3 Dedektör içindeki eta sınırları ($-5 \leq \eta \leq 5$)

ATLAS deneyi kollarasyonunda, 35 ülkeden 150 üniversite ve laboratuvarından katılan toplam 1800 fizikçi bulunmaktadır [9]. Atlas dedektörünün farklı bölümlerinde birçok Türk bilim insanı da çalışmaktadır. ATLAS dedektörü dört farklı katmanın üst üste monte edilmesiyle elde edilmiştir. Bu katmanlar iz takip edici, elektromanyetik kalorimetre, hadronik kalorimetre ve Müon kalorimetre olarak verilebilir.



Şekil 2.4 ATLAS dedektörü katmanları

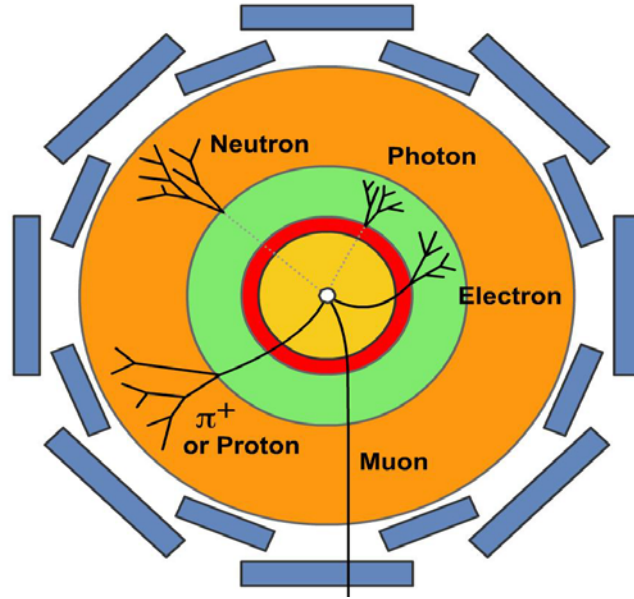
2.3.1. İz Takip Edici

Proton demet borusundan radyal yönde 1.5 cm'den başlayarak 1.2 m'ye kadar uzaklaşan üç ayrı kısımdan oluşur. En içte Pixel, ortada SCT ve dışta TRT vardır. İz dedektörünün ana amacı etkileşme noktasından gelen parçacıkların momentumlarını ölçmek ve bu parçacıkların izlerini belirleyerek etkileşme köşelerini bulmaktır [12].

Başka bir deyişle çarpışmada oluşan yüklü parçacıkların enerjilerinin bir kısmını iyonizasyonla kaybettirerek, parçacıkların momentumunu, yükünü ve yörüngesinin belirlenmesini sağlamaktır. Çünkü dedektörde sadece yüklü parçacıklar izlerini bırakırlar.

Ancak parçacıklar yüksüz ise, foton ve nötron gibi, o zaman içerisinden geçtikleri bölgeyle hiçbir etkileşmede bulunmadan diğer bir tabakaya geçiş yaparlar. İlk tabaka da enerji kaybı yok denecek kadar azdır. Çünkü buradaki parçacıklar etkileşimlerden neredeyse hiç etkilenmezler. Solenoid manyetik alan parçacıklarının elektrik yükleri nedeniyle bıraktıkları izleri takip ederler ve parçacığın hızını ve etkileşme noktalarını belirlememizi sağlarlar.

Müonlar dedektörün dış çeperinde bulunan bu tür cihazlar ile yakalanırlar. Bu detektörler güçlü bir manyetik alan içerisinde bulunur. Manyetik alan parçacıkların yörüngelerinin dairesel olarak bükülmesine sebep olur. Her bir yörüngenin yarıçapı parçacığın momentumunu, bükülme yönü ise parçacığın yükünün işaretini belirler. Parçacıkların elektrik yükü taşıyıp taşımadıkları, iz saptama katmanında iz bırakıp bırakmamalarından yola çıkarak anlaşılır. Yüklü bir parçacığın yörüngesi; detektörün içindeki elektromıknatısın oluşturduğu alanın etki ettirdiği manyetik kuvvetin hareket doğrultusuna dik yönde olması nedeniyle kıvrılır. Eğer artı yüklü bir parçacığın yörüngesi, saat yönünde kıvrılmışsa, eksi yüklü parçacığınki ters yönde kıvrılacaktır.



Şekil 2.5 Yüklü bir parçacığın yörüngesi

Bu durum detektörün, ekseni doğrultusunda bakıldığında görülen dairesel kesitinde, zıt yüklü parçacıkların zıt yönlerde hareket etmeleri şeklinde görünür. Dolayısıyla, parçacıkların sadece yüklerinin var olup olmadığı değil, varsa bu yükün işareti de kolaylıkla belirlenebilir.

Ayrıca, yükü bilinen bir parçacığın, üzerindeki merkezkaç (mv^2/r) ve manyetik kuvvetlerin (qvB) eşit olması gerektiğinden; yörüngesinin eğrilik yarıçapından hareketle, momentumunu ($p=mv$) hesaplamak ($p=qBr$)'da oldukça kolaydır. Yapılan deneyler sağlam bir iz takip edici detektörün güçlü manyetik alanda müon, elektron, foton ve jetlerin rekonstrüksiyonu için çok güçlü bir araç olduğunu göstermektedir [13].

2.3.2. Elektromanyetik Kalorimetre

Elektromanyetik kalorimetre elektron, pozitron ve foton gibi elektromanyetik etkileşme yapan parçacıkların enerjilerini soğuran detektörlerdir. Bu dedektörün amacı soğurduğu bu parçacıkların enerjilerini ölçmektir. Gelen yüksek enerjili elektron ve fotonlar kalorimetrenin atomları ile etkileşerek enerjilerini daha düşük enerjili elektron, pozitron ve fotonlardan oluşan elektromanyetik duşlarla kaybeder. Elektromanyetik etkileşen parçacıklar enerjilerini iyonlaşma, bremsstrahlung (frenleme ışınımı), çift oluşum, çift yok olma, Compton, fotoelektrik etki gibi fiziksel süreçlerle kaybeder. Kaybolan enerjiler elektrik sinyalleri olarak kaydedilir ve bu sinyaller parçacıkların başlangıçtaki enerjileriyle orantılıdır. Elektromanyetik Kalorimetre yüklü ve nötral parçacıkların enerjilerini ölçmek için tasarlanmıştır. Aralarında $-185\text{ }^{\circ}\text{C}$

sıcaklığa sahip sıvı argon bulunan emici çelik katmanlar ve akordiyon şekilli kurşun elektrotları içerir. Sıvı argon, parçacık yağmurunun etkisi ile elektronları serbest bırakır. Bunlar ise elektrotlar tarafından alınarak çiplere gönderilir. Böylece veri akışı başlar. Örneğin; çelikten geçen bir elektron; daha düşük enerjili elektron, pozitron ve foton yağmuru oluşturur.

Elektromanyetik Kalorimetre yaklaşık 15 mm incelikte kurşun levhalar ve bunların aralarına yerleştirilmiş sintilatör kristallerinden oluşmaktadır. Elektromanyetik kalorimetre iz takip edicinin etrafına yerleştirilmiştir. Enerjisi yüksek olan elektronlar Elektromanyetik Kalorimetreye girince, ortamdaki atom çekirdekleri ile elektromanyetik etkileşme yaparak yüksek enerjili fotonlar üretirler. Bu fotonlarda çekirdeğin Coulomb alanından etkilenip elektron-pozitron çiftleri oluştururlar. Oluşan elektronlar tekrar yeni fotonlar, fotonlarsa tekrar yeni elektron pozitron çiftleri oluştururlar. Bu durum oluşan ikincil parçacıkların enerjilerinin yeni parçacık üretmeye yetmeyeceği ana kadar sürüp gider [14].

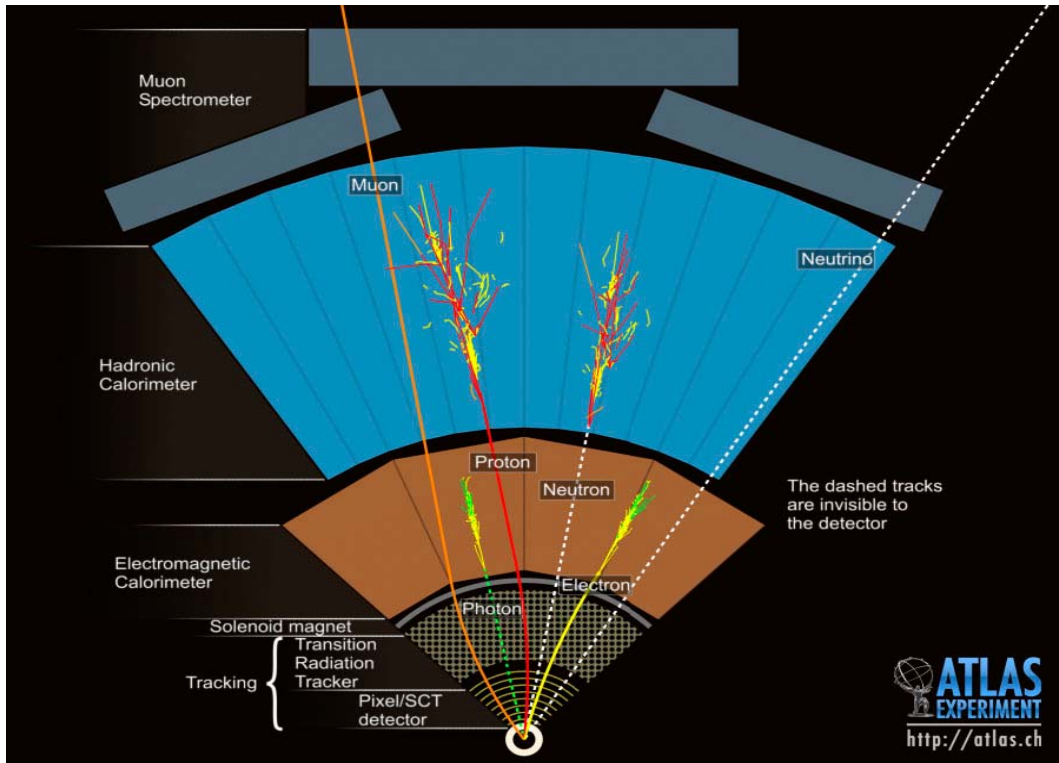
2.3.3. Hadronik Kalorimetre

Proton, nötron gibi baryon ve mezonları içeren hadron olarak adlandırılan parçacıkların enerjilerini ölçmek için tasarlanmıştır. Emici materyal olarak çelik kullanılır. WLS (Dalgaboyu kaydırma) fiberler tarafından okuma yapılan sintilatör levhalara sahiptir. İç yarıçapı 2280 mm ve dış yarıçapı 4230 mm olan bir silindirik yapıya sahiptir. Fiberlerdeki foton akışının yoğunluğu elektronik sinyaller olarak ölçülür. Hadronik kalorimetreler çarpışmadan çıkan hadronların (proton, nötron, pion ve diğer mezonların) enerjilerini ölçen aletlerdir. Hadronik kalorimetre elektromanyetik kalorimetrenin dışına yerleştirilmiş durumdadır. Bu nedenle elektronlar ve fotonlar Elektromanyetik Kalorimetrede soğurulduğu için burada oluşan sinyale katkıda bulunmazlar [15].

2.3.4. Muon Kalorimetre

Genel olarak Müon kalorimetreler dedektörün en dış kısmında bulunur. Amacı müonları algılamaktır. Müonlar keşif çalışmalarında önemli bir rol oynayacaktır. Çarpışmadan sonra oluşan yüklü parçacıklardan muonların yaşam süreleri göreceli olarak diğer atomaltı parçacıklardan birçoğuna göre daha uzun olduğundan ve etkileşmeden müon kalorimetreye kadar gidebildiklerinden dolayı, kalorimetrelerden sonra detektörün en dış kısmında bulunan muon detektörlerinde algılanırlar. Muonların kütlesi elektrondan yaklaşık 200 kat fazla olması sebebiyle atomlarla elektriksel bir etkileşmeye girmezler. Bu nedenle elektromagnetik sağanak oluşturmazlar. Enerjileri 5 GeV civarında olan muonlar, bakır, çelik gibi metallerin her milimetresinde yaklaşık 1 MeV enerji kaybına uğradıkları için çok fazla enerji kaybetmeden

kalorimetreleri geçebilirler. Yüksek Enerji Dedektörlerinde değişik şekilde tasarlanmış muon deteksiyon sistemleri kullanılmaktadır. Müonlar ağır ($\sim 105,65$ MeV) ve uzun ömürlü ($\sim 10^{-6}$ s) parçacıklardır. Bu nedenle de detektörde çok temiz sinyaller verirler. Büyük Hadron Çarpıştırıcısındaki araştırmaların büyük bölümünde etkileşme ve bozunumlarda müonlar çıkmaktadır. Bu nedenle bunların büyük bir hassasiyetle ölçülmesi gereklidir [16].



Şekil 2.6 ATLAS dedektöründeki kalorimetreler

2.4 CMS (Compact Muon Solenoid - Kompak Müon Sarmalı)

Kompak Müon Sarmalı ise ölçüm tekniği itibari ile daha küçük boyuttadır. CMS, radyasyona karşı çok dirençli ve elektronik donanımı çok iyi olan, yüksek teslalı bir manyetik akım sarmalına sahip, parçaları ileri teknolojiyle üretilen, yüksek performanslı bir müon sistemi, elektromanyetik kalorimetre, çok iyi kalitede iz izleme sistemleriyle, enerji çözünürlüğü çok yüksek olan bir hadronik kalorimetreden oluşmuş, gelişmiş bir sistemdir. CMS, LHC’de en yüksek lüminositede bile çalışabilecek ve LHC’de araştırılması önerilen bütün fizik potansiyeline cevap verecek durumdadır. Tüm detektör parçaları geniş bir enerji spektrumunda etkileşimler sonucunda ortaya çıkan müonları, elektronları, fotonları ve hadron jetlerini tam olarak ölçecek şekilde tasarlanmıştır.

CMS dedektörü 14500 ton ağırlığında, 14,6 m yüksekliğinde, 24 m uzunluğundadır ve 4 teslalık bir süper iletken solenoid mıknatısa sahiptir. CMS'in mıknatıs sistemi 10 bin ton demir içermekte olup bu miktar Eyfel kulesinden daha fazladır. CMS'te de 3500'den fazla kişi çalışmaktadır. Türk bilim insanlarının yaygın olduğu deney bölgesinden biridir. Bu dedektörlerin veri üretme hızı hakkında bir fikir sahibi olmak için bir yılda üretecekleri verinin yazıldığı CD'lerin dünyadan aya yol olacağını bilmek yeterlidir. Bu verilerin bilgisayarlarda depolanması ve işlenmesi, veri analizi için ciddi bir sorun oluşturmaktadır. Bulunan çözüm ise elde edilen verilerin dünya üzerinde çok çeşitli ülkeleri ve kurumları içeren GRID adlı bilgisayar ağına dağıtılması ve gerektiğinde ilgili bilgisayarlardan okunarak kullanılması şeklindedir. GRID ağı aynı zamanda sağlık alanında da kullanılacaktır [17].

2.5. LHCb

LHCb, hadron fiziği özellikle de B-mezon (bir adet 'alt kuark-d' ile onunla aynı özelliklere sahip ancak ondan bin kat daha ağır olan 'güzel kuark-b' olarak tabir edilen diğer bir kuark ve bunları birbirine bağlayan gluondan oluşan hadron) fiziği araştırmaları yapacak olan bir dedektördür [17]. LHCb dedektörünün araştırmaları sonucunda evrende bulunan madde ve karşıt-madde arasındaki simetri kırılmasının aydınlatılacağı düşünülmektedir. Yukarıda da bahsettiğimiz gibi proton-proton çarpışmaları sırasında ortaya çıkacak olan özellikle de b kuarka sahip olacak parçacıkların çalışılması ve modellenmesi bu dedektör sayesinde sağlanabilecektir [18]. Bu çalışmalara ek olarak LHCb Dedektöründen toplanan veriler sayesinde "CP Violation-yük ve parite ihlali" ve "rare decays - seyrek bozunumlar" üzerine de veri analizi çalışmaları yapılabilecektir.

2.6. Alice

ALICE dedektörü, kuark-gluon plazmasını araştırmak üzere tasarlanmıştır. Kuarkların ve gluonların yüksek sıcaklık ve basınç altında proton gibi, B-mezon gibi hadronları oluşturmak yerine bir plazma oluşturmaları beklenmektedir. Büyük Patlama'nın hemen ardından evrenin bu halde olduğu düşünülmektedir ve ALICE evrenin bu evresini laboratuvar koşullarında oluşturacak ve analiz yapılmasına imkan sağlayacaktır.

ALICE bize evrenin bebeklik dönemi hakkında önemli bilgiler verecektir. Kuark-gluon plazmasının oluşmasını sağlayabilmek için proton yerine her biri 82 tane proton içeren kurşun iyonları çarpıştırılacaktır. LHC'nin kurşun-kurşun çarpıştırıcısı olarak çalışması proton-proton sürecine benzer şekilde yürütülmektedir bir farkla ki elektronların tamamen ayıklanması bir kaç adımda gerçekleştirilebilmektedir [17].

3. DEDEKTÖR MODELLEME

Simulasyon, gerçek bir problemin bilgisayarla yapılan modelidir. Başka bir deyişle teoriksel ya da gerçek fiziksel bir sisteme ait neden sonuç ilişkilerinin bir bilgisayar programına yansıtılmasıyla, değişik koşullar altında gerçek sisteme ait davranışların bilgisayar modelinde izlenmesini sağlayan bir yöntemdir. Dedektör simülasyonu ise doğal ve insan yapımı dedektörleri benzetme sanatıdır.

Bir simülasyon program paketinin genel yapısı iki ana bileşene ayrılmıştır. Bunlar, olay üretimine kadar giden fizik modellemesi ve dedektör simülasyonudur. Yeniden oluşturma programı hem simülasyon hem de gerçek veri akışı için ortaktır [19].

Gerçek problemler doğada rastgele ve karmaşıktır bu yüzden de çok sayıda değişken içerirler. Bu tür problemlerin analitik çözümlerini bulmak oldukça zordur. Çözüm için bir yaklaşım, sistemin bilgisayar modelinin kurulması ve Monte Carlo analizi gibi simülasyon teknikleri kullanılması gerekmektedir. Simülasyonlar önemlidir, çünkü çok pahalı olan gerçek bir olayın yerine dikkat isteyen deneylerdeki sistem parametrelerini değiştirerek sistemin davranışını gözlemlememize ve sonucu almamıza dayanır. Böylece simülasyonlar gerçek deneylerde meydana gelebilecek belirli olayların şartlarını deney yapılmadan oluşturmamıza ve bu olayların etkisini sanki olmuş gibi araştırmamıza imkan sağlarlar. Simülasyonda temel adım ardışık rasgele sayı üretmektir. Bu rasgele sayı üreticileri simülasyon çalışmalarında çok önemlidir. Tanımlayıcı istatistik, gözlenen (ölçülen) verilerin sınıflandırılması, temsil edilmesi ve indirgenmesi için grafiksel ve sayısal tekniklerle ilgilenir.

Böyle yöntemler gözlenen verilerin yorumlanmasında, hesaplanmasında ve onlardan sonuç çıkarılmasında yardımcı olmaktadır [19].

Dedektör modelleme programlarına PGS, ATHENA, GEANT4 gibi programları örnek verebiliriz. Bu tezde hızlı dedektör modelleme programı olan PGS, tam dedektör modelleme programı olan ATHENA ve olay üretici programı olan CompHEP programlarından detaylı olarak bahsedilecektir.

3.1 PGS

PGS (Pretty Good Simulation), basit, hızlı, fizibilite analizleri için uygun bir dedektör modelleme programıdır. Bu modelleme programı SHW adıyla 1998'de Tevatron (ABD'de Chicago şehrinin doğusundaki fermilab (Fermi Ulusal Hızlandırıcı Laboratuvarı) da bulunan dairesel bir parçacık hızlandırıcısı için yapılmıştır. Kendisine 150 GeV olarak yollanan proton ve antiprotonları 1.96 TeV enerjiye kadar hızlandırıp, 2 ayrı noktada çarpıştırır) Run2

SUSY/Higgs çalıştayında geliştirilmeye başlanmıştır. Asli yazarı John Conway (UC Davis)'dir. Son sürümü PGS 4 – 070423'dür [20].

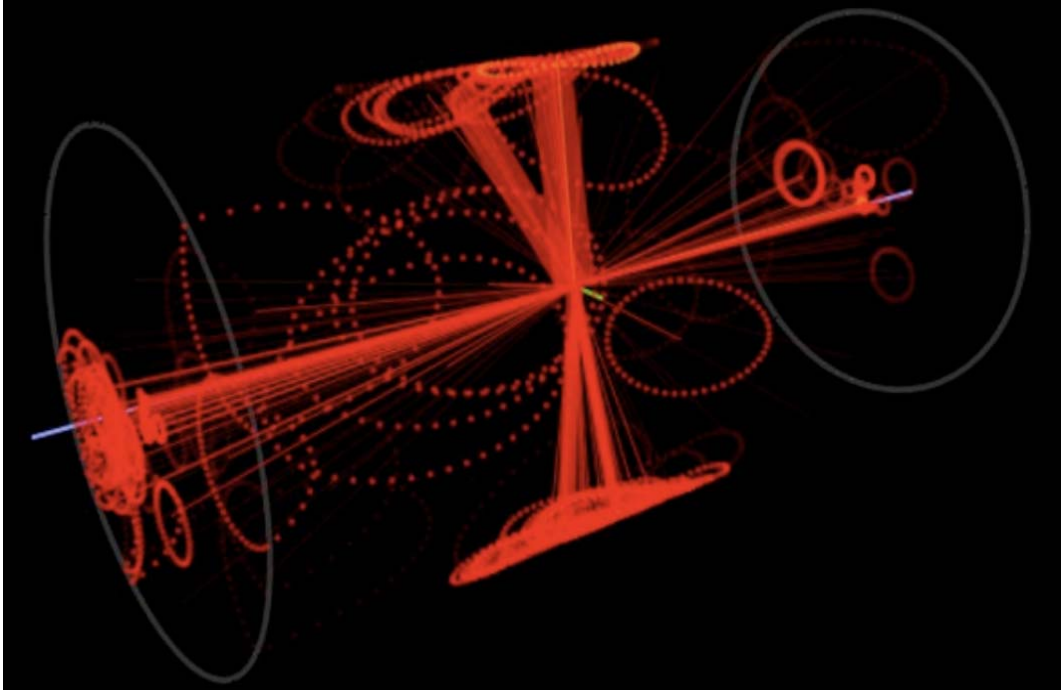
PGS programı Fortran programlama dili ile yazılmış, g77 veya gfortran ile derlenebilmektedir. Sadece standart STDHEP yüksek enerji fiziği kütüphanesine bağımlıdır. Bu kütüphane üretilen olayların kayıtlarını içerirken, STDTBL kütüphanesi ise parçacıkların özelliklerini (isim, kütle ve genişlik vb.) içermektedir. PGS'nin hemen hemen tüm görevleri tek tek fonksiyonlar halinde kütüphanede yer alır. Kütüphane fonksiyonlarını sırayla çağırır. İşlem tamamlanınca kullanıcının olayı çözümlene kodunu işleme sokar.

MadGraph ile gelen program (pythia-pgs_V2.0.26.tar.gz) LHC olimpiyatları için tasarlanmış ve standart hale getirilmiştir. Bu program tetikleyicinin maksimum işlemlerini uygular ve çıktıları .LHCO kütükleri biçiminde verir [21].

PGS eğer dedektörün sınırları içerisindeyse tüm son durum parçacıklarını işleme alır. Eğer parçacık yüklüyse doğrusal bir iz yaratır. (manyetik alan olmasına karşın eğilme yoktur.)

Kalorimetrede kuarkların izlerine jet denir ve uzaysal koni biçimindeki şekiller içerisinde eta ve phi şeklinde iki uzaysal koordinat değişkeni ile jetlerin yerleri, enerjileri momentumları belirlenir ve aranır. Koni içerisinde eta ve phi değerleri belirli karelere ayrılmıştır.

Parçacıklar hangi karelere yönlendiğinde orada enerjilerini yerleştirir ve o karedeki enerji doluluğu sinyalde kıvılcım'a sebep olur. Bu metod jetlerin özelliklerinin bulunması ve jetlerin yeniden oluşturulması için iyi bir yöntemdir.



Şekil 3.1 Koni temelli jet görüntüsü

PGS’te jetlerin dedekte edilmesi için Koni veya Çift bulma metodları kullanılır. Jetleri Koni temelli ya da Çiftleştirme temelli olarak işleme alır. Koni temelli jetlerde, en yüksek enerjili ve belirli bir minimum değere sahip olan kalorimetre hücresiyle başlar ve çevresine bir koni çizer. Koninin içine düşen hücrelerdeki enerjileri de ekler. Böylece bunun bir jet olup olmadığına bakar ve bu bir jet ise başa döner ve geri kalan hücreler için bunu tekrarla şeklinde çalışır. Çiftleştirme temelli jetler ise birbirlerine olan enine momentumları ışın eksenine göre olan enine momentumlarından belli bir kat küçük olan hücreleri birbirleriyle birleştirir, birleştirme yapılmayacak olana kadar devam eder [22].

3.2 Athena

ATHENA, LHC deneyinden sonra çok fazla miktarda ortaya çıkacak olan parçacıklar için; deney yapılmadan; sanki üretilmiş ve dedektörden geçmiş gibi işlem yapılacak olan parçacıkların dedekte edilmesine ait tüm karmaşıklığı çözmemizi sağlayacak tam dedektör modelleme programıdır.

ATHENA Linux işletim sistemi komutlarıyla çalışabilen bir programdır. Python yazılım dilini kullanır. ATHENA çalıştırılırken iki farklı yolla sonuçlara ulaşılabilir. Bunlar “**Kits**” ve “**Releases**” dir. Bunlardan “**Releases**” SSH yardımıyla CERN-LXPLUS hesabına bağlanarak gerekli kütüphanelerin kullanılmasıyla başlatılan çalışmadır. **Kits** seçilerek çalıştırma

durumunda TÜBİTAK-ULAKBİM aracılığı ile CERN'e ait gerekli kütüphanelerin yüklenme şartları ve ortamı oluşturulduğunda yapılan çalışmadır [23].

Öncelikle bu çalışmaların yapılabilmesi için bir lxplus hesabı ve grid'e iş göndermek için de CERN'den veya TÜBİTAK-ULAKBİM'den CERN-GRID SERTİFİKASI VE CERN ATLAS'tan ATLAS-VO SERTİFİKASI almak gereklidir. Bu sertifikalar bir yıllık geçerli olmak üzere alınmaktadır. Bu iki gereksinim sağlandıktan sonra ATLAS yazılımı paket yönetimi tarafından kullanılan CMT yazılım aracını lxplus hesabındaki dizinde derleyerek çalışmalara başlanabilir. Bunun için öncelikle hesap dizininde “cmthome” ve “scratch0” adında çalışma ve sonuçların konulacağı iki dizin yaratılır. Daha sonra “cmthome” dizininde gerekli bağlantıları kurabilmek için “requirements” adında bir dosya yaratarak (Bunun için herhangi bir editör kullanılabilir; nano, vi, kate gibi) içerisine gerekli bağlantı bilgileri eklendikten sonra dosya kaydedilip kapatılır.

```
cd $HOME
```

```
mkdir cmthome
```

```
cd cmthome
```

requirements dosyasının içeriği aşağıdaki şekilde düzenlenmeli.

```
set CMTSITE CERN
```

```
set SITEROOT /afs/cern.ch
```

```
set HOME /afs/cern.ch/user/...
```

```
macro ATLAS_DIST_AREA ${SITEROOT}/atlas/software/dist
```

```
macro ATLAS_TEST_AREA ${HOME}/scratch0
```

```
use AtlasLogin AtlasLogin-* $(ATLAS_DIST_AREA)
```

Şimdi de CMT ayarlarının kullanıcı hesabı için hazırlanması gerekir.

```
Source/afs/cern.ch/sw/contrib/CMT/v1r20p20090520/mgr/setup.sh
```

```
cmt config
```

Buraya kadar yapılan komut satırlarının birkez yapılması yeterli olacaktır. Bundan sonraki çalışmalar için bu komut satırlarının yapılmasına gerek yoktur. Bu adımlardan sonra ATHENA yazılımının hesabın olduğu yerde çalışır hale getirilmesi gerekir. Bunun için ana dizinde gerekli komutların “source” ismiyle kaynaklanması yeterli olacaktır.

cd \$HOME

Burada ATLAS dedektör programı olan ATHENA'nın ATHENA 15.4.0 versiyonu kullanılmaktadır. Bu yazılım 32 bitlik bilgisayarlar için yazılmıştır. Şimdi çalışmaların yapılacağı TEST alanının hazırlanması gerekir. Bu süreç için de, aşağıdaki komutlar yazılır. Böylece çalışmalar için \$TestArea isminde bir çalışma dizini yaratılmış olur.

Echo \$TestArea

mkdir \$TestArea

ATLAS - ATHENA yazılımını çalıştırmak için bütün ortam şartları sağlanmış olmaktadır. Oluşturulan \$TestArea alanına gidilerek burada "run" isminde bir dizin yaratılır ve bu dizine örnek olarak çalıştırılacak olan "HelloWorld.py" python dilinde yazılmış olan dosya kopyalanır, bu işlemden sonra dedektör programı olan ATHENA 15.4.0 programı çalıştırılmış olur.

cd \$TestArea

mkdir run

cd run

athena HelloWorldOptions.py

Burada bir Athena paketini derleyebilir, çalıştırabilir, modifiye edilebilir ve çıktıları kontrol edilebilir. Bu işlemler için "Test" alanına gidilmelidir.

cd \$TestArea

cmt show versions Control/AthenaExamples/AthExHelloWorld

cmt co -r AthExHelloWorld-01-02-00 Control/AthenaExamples/AthExHelloWorld

cd Control/AthenaExamples/AthExHelloWorld/

ls

Elde edilen dosyalar kontrol edildikten sonra eğer istenirse yeni paketlerde de değişiklik yapmak derlemek ve yeni bir iş çalıştırmak için aşağıdaki komut satırlarının uygulanması yeterli olacaktır.

```
cd cmt
```

```
cd ../cmt
```

```
cmt config
```

```
source setup.sh
```

```
gmake
```

Şimdi iş göndermek için ayarların tekrar yapılması ve yeni bir test alanı yaratılması gerekmektedir.

```
source cmthome/setup.sh -tag=15.4.0,AtlasProduction,setup,releases,32
```

```
cd ../
```

```
mkdir $TestArea
```

```
cd $TestArea
```

Bu kısım da olay üretiminin gerçekleştirilmesi ve çıktı dosyalarının oluşturulması gibi çalışmalar yapılabilir [24].

3.3. Comphep

CompHEP projesi 1989 yılında bir grup fizikçi ve Moskova Üniversitesi'nin Nükleer Fizik Enstitüsü programcılarında D.V Skobeltsyn tarafından bulunmuştur. Comphep yazılımının yazarı ise Alexander Pukhovdur [25]. CompHEP Feynman diagramlarının değerlendirildiği ve saçılma parçacıklarına ait tesir kesitlerinin, dallanma oranlarının, polarizasyon durumlarının, simetritlerinin gözlemlenebildiği, grafiklerin ve Feynman diagramlarının elde edildiği bir paket programdır. Yüksek enerjili parçacıklarla yapılan çalışmalarda iyi sonuçlar verir.

CompHEP içerisinde Standart Model'e, Standart Model ötesine ve diğer bazı temel parçacıklar için önerilen modellere ait (SUGRA) gibi modellerin çalışmasına izin verir. Program hızlı ve basittir. Ele alınan incelenen kanalların konularını hızlı sonuçlandırır. Fakat çok parçacıklı bozunmalara gidememe gibi eksiklikleri vardır. CompHEP'te çalışılan bir çarpışma veya saçılma kanalını inceleyebilmek için dört kısımda giriş yapmamız gerekir.

Bunlar Lagranjiyen (parçacıkların alışverişindeki ara kuvvetlerin Feynman diyagramı üzerindeki yansıması olan etkileşme köşelerini ifade eder.)

Constrait (parçacıkların kütlelerine, yüklerine, spinlerine gelen sınırlamaların programa girildiği kısımları ifade eder.), particle (parçacıkların özelliklerinin verildiği bölümdür.) ve parameters bölümünde ise çalışılan prosese ait (toplam enerji, yükler, parçacık kütleleri gibi) nicelikler içerilir [25].

CompHEP programı iki bölüm içermektedir: Sembolik ve Sayısal kısım.

Sembolik kısım C programlama dilinde yazılmakta ve çalışmaktadır. Fortran ve C kodlarını içerir. Bunlar daha sonra sayısal hesaplamalarda kullanılmaktadır.

Sayısal kısmın ise iki versiyonu vardır: bir tanesi Fortrande yazılan diğeri de C'de yazılan kısımdır. Bu iki hizmet hemen hemen aynıdır. Fakat C versiyonu daha kullanışlıdır. Sembolik kısmın içeriğinde ise gelen ve çıkan parçacıkların bozunumlarının girilmesi, Feynman diagramlarının üretilmesi, bu üretilen diyagramların şekilsel gösterilmesi ve LATEX'te karşılık gelen verilerin oluşturulması gibi kısımlar mevcuttur.

Sayısal kısmın içeriğinde; işlemlerde yer alan fiziksel parametrelerde (toplam enerji, yük, kütle, ...) değişiklikler yapılabilir. Parton yapısı ve QCD (Quantum Chroma Dynamic - Kuantum Renk Dinamiği) hesaplamaları için parametrelerin seçilmesi, farklı kinematik kriterlerin programa ve prosese girilmesini ifade eder. Kinematik işlemleri (özel uzay parametreleri) Monte Carlo yöntemi içinde tanımlanır, Monte Carlo etkileşim evreleri ise VEGAS Yöntemi tarafından üretilir. Hesaplanmış matrix öğelerinin değerleri bir dosyada saklanır, bir sonraki bozunma için girilen farklı parametrelerin dağılımları oluşturulur. En küçük hata payıyla grafikler oluşturulmuş olur [26].

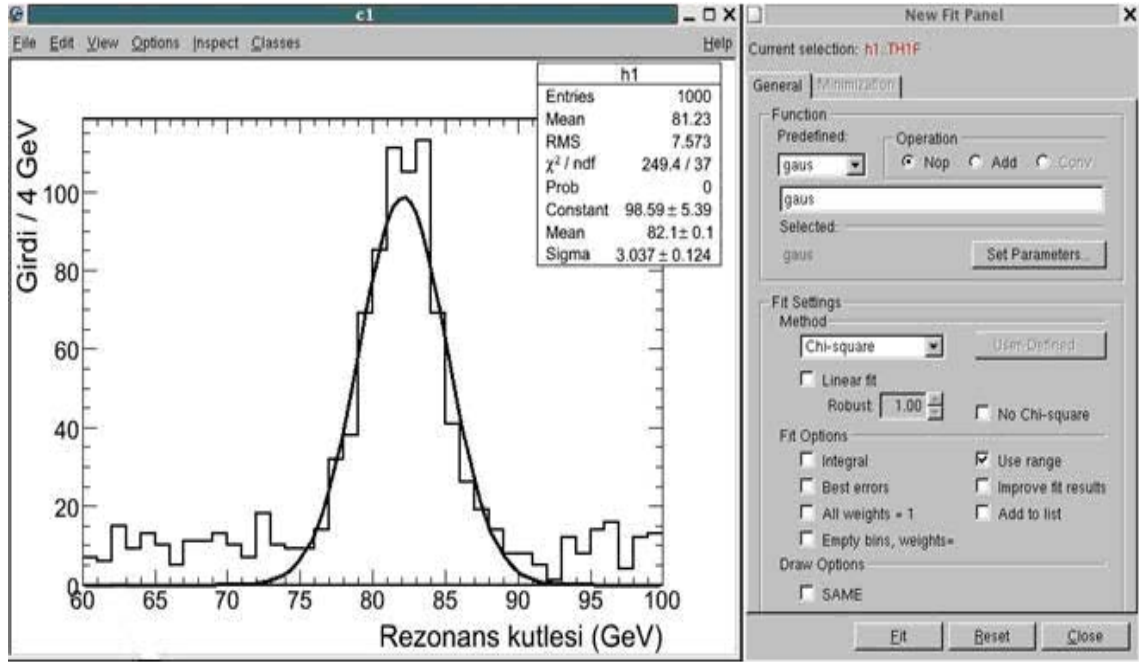
3.4 Root

ROOT Programı C++'ta yazılmış nesne-yönelimli bir program olup; her tür dedektör modelleme ve rastgele olay üreteç simülasyonlarının program çıktılarının analizine hem görsel hemde işlemsel olarak izin verir. Nesneye yönelik progama nesnelerini temel olarak kabul eden bir programdır.

ROOT programının altyapısı; Sınıf, Nesne, Metod fonksiyonu gibi dallardan oluşur. Sınıf bir sistemdeki bütünlüğü ifade ederken, nesne ise bir sınıftan yaratılmış belli bir durumu ifade etmektedir. Sınıflar programlarda gerçek nesnelere yaratmak amacıyla kullanılan bir yapıdır. Sınıf veri üyeleri ve üye fonksiyonları tanımlayarak nesnenin yapısını ve davranış biçimini ortaya koyar.

ROOT'ta önceden tanımlanmış (TMath::) veya tanımlayacağımız fonksiyonların histogramlarını oluşturmak, bu histogramlardaki grafikleri matematiksel fonksiyonların kullanıldığı uyum eğrisi ile matematiksel fonksiyonlara uyarlamak - fit etmek - mümkündür.

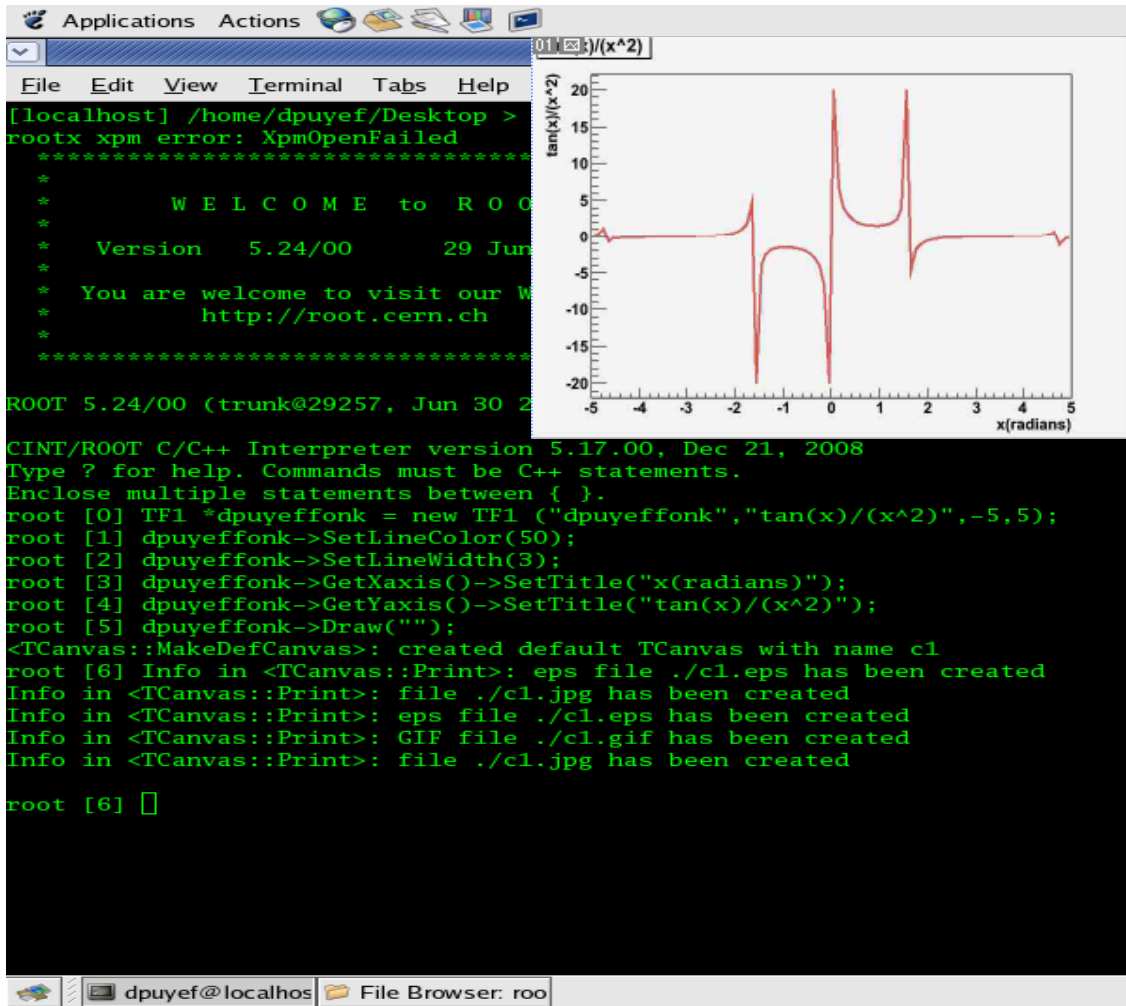
Histogram fit etmeye bir örnektir.



Şekil 3.2. Histogram örnek görüntüsü

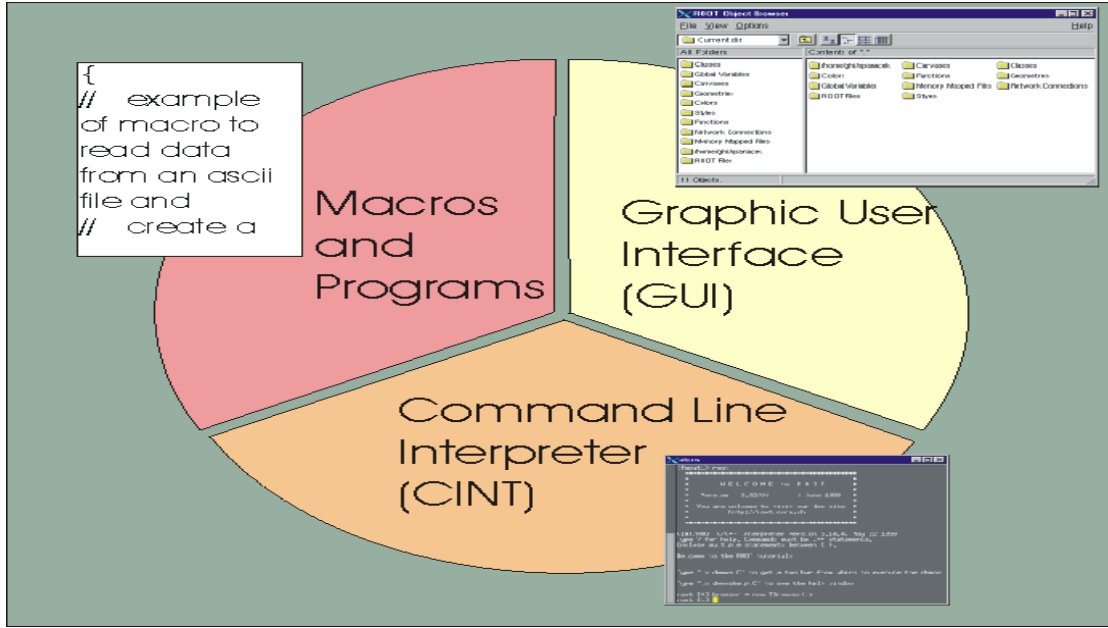
ROOT programı ile makro oluşturmak, betik yazmak ve bu betikleri çalıştırmakta mümkündür [27].

Örnek olarak $\tan(x) / (x^2)$ fonksiyonu örneğine bakılabilir.



Şekil 3.3 $\tan(x)/(x^2)$ fonksiyonu

Bununla birlikte ROOT açılışa yüklenen, gerek duyulduğunda yüklenen ya da özel amaçlı olarak yüklenen bir kütüphane yapısına sahiptir. ROOT programı C++ temeline dayanan bir program olmasına karşın birçok kavramı FORTRAN temelli bir istatistik grafik paket programı olan PAW'a (Physical Analysis Workstation - Fiziksel Analiz İstasyonu) dayanmaktadır [27]. Aynı zamanda bu program geliştiricilerin uygulama geliştirmek için ihtiyaç duyacakları şeyleri sağlayan araçlar toplamıdır.



Şekil 3.4 Root işlevselliği

Yorumlayıcı, histogram ve veri analiz ayrıştırma işlevleri, grafik kullanıcı ara yüzü (GUI ve HI) geliştirici, bölüm (class) kitaplığı, giriş / çıkış (I/O) işlevleri gibi bu çalışma ortamının bize sağladığı kolaylıklar ise; belli bir işlevsellik için daha az kaynak kod yazmamız gerekliliği, geliştirilen programın yüksek bir güvenilirliğe sahip olması, parçalanabilir özelliğe sahip olmasıyla da tekrar kullanılabilir parçalardan oluşan esnek bir yapıda olması şeklinde sıralayabiliriz. ROOT programının Linux işletim sistemi ve Windows işletim sisteminde çalışan versiyonları da mevcuttur. Yukarıda konusu geçen kullanım şekillerine birkaç örnek verebiliriz [27].

Aşağıda `root_v5.22.00.source.tar.gz` versiyonunun Linux işletim sistemindeki kurulum adımları verilmektedir.

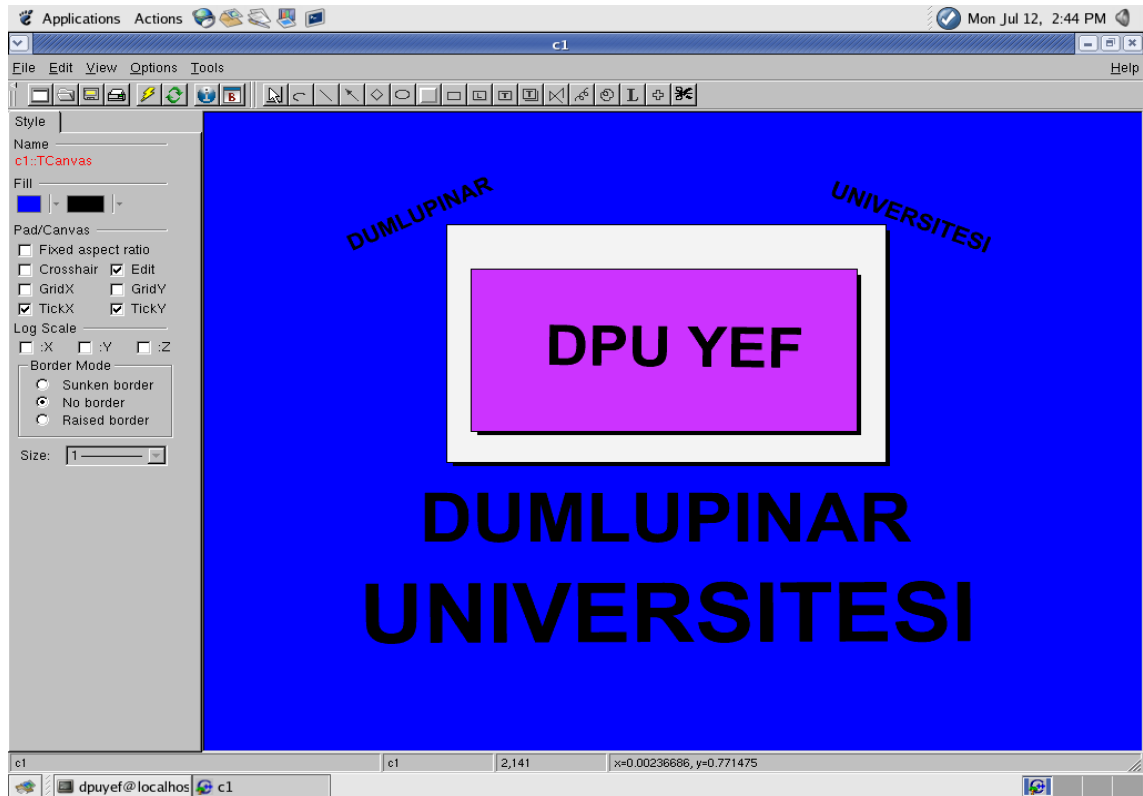
```
> tar -zxvf root_v5.22.00.source.tar.gz
> export ROOTSYS .....
> export PATH .....
> export LD_LIBRARY_PATH .....
> cd root
> ./configure
> make
> sudo make install
```

Yukarıdaki komut işlemlerinden sonra ROOT programı çalıştırılabilir. Tek yapılması gereken konsolda “root” yazmak ve böylece ROOT programını etkin hale getirmektir. Bu komut kullanıldığında root açılış sayfası ROOT logosu görünümüyle çalışmaya başlar.

Eğer ROOT logosu görülerek vakit harcanmak istenmiyorsa, o zaman yapılması gereken komut satırında “root.exe” yazmak ya da “root -l” komutunu kullanmak sorunu çözecektir.

ROOT komut satırını görüntülemek içinse root [0] ? ya da root [1] .h yazılması yeterlidir. Oturumu kapatmak için ise root [3] .q yazılmalıdır.

ROOT Programı, CMS ve ATLAS gibi birçok CERN deneyinde veri analizi için aktif şekilde kullanılmaktadır [28]. TCanvas kullanımına örnek olarak aşağıdaki şekil verilebilir.



Şekil 3.5 TCanvas uygulaması

Yukarıdaki TCanvas uygulamasına ait komut satırları aşağıdaki şekildedir.

```

{
//=====Macro generated from canvas: c1/c1
//===== (Thu June 7 19:50:37 2010) by ROOT version5.22/00
TCanvas *c1 = new TCanvas("c1", "c1",0,48,1105,951);
c1->Range(0,0,1,1);
c1->SetFillColor(4);
c1->SetBorderMode(0);
c1->SetBorderSize(1);
c1->SetTickx(1);
c1->SetTicky(1);
c1->SetFrameFillColor(19);
TPave *pave = new TPave(0.2586207,0.4512712,0.7270115,0.8241525,4,"br");
pave->Draw();
TPaveText *pt = new TPaveText(0.284287,0.5005599,0.6966394,0.7547592,"br");
Int_t ci; // for color index setting
ci = TColor::GetColor("#cc33ff");
pt->SetFillColor(ci);
pt->SetTextSize(0.03601695);
pt->Draw();
TLatex * tex = new TLatex(0.3642144,0.6035834,"DPU YEF");
tex->SetTextSize(0.09406495);
tex->SetTextAngle(359.2748);
tex->SetLineWidth(2);
tex->Draw();
tex = new TLatex(0.226158,0.3393057,"DUMLUPINAR");
tex->SetTextSize(0.1097424);
tex->SetLineWidth(2);
tex->Draw();
tex = new TLatex(0.1534968,0.1982083,"UNIVERSITESI");
tex->SetTextSize(0.1388578);
tex->SetLineWidth(2);
tex->Draw();
tex = new TLatex(0.1580381,0.7816349,"DUMLUPINAR");
tex->SetTextSize(0.03583427);
tex->SetTextAngle(25);
tex->SetLineWidth(2);
tex->Draw();
tex = new TLatex(0.6603088,0.8734602," UNIVERSITESI");

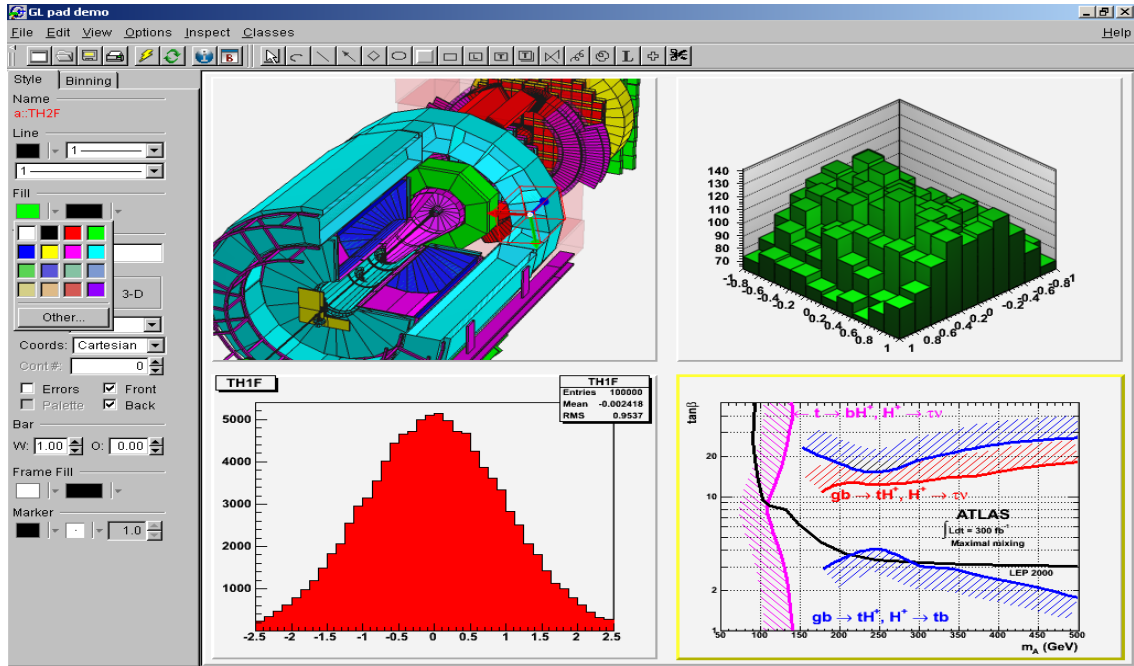
```

```

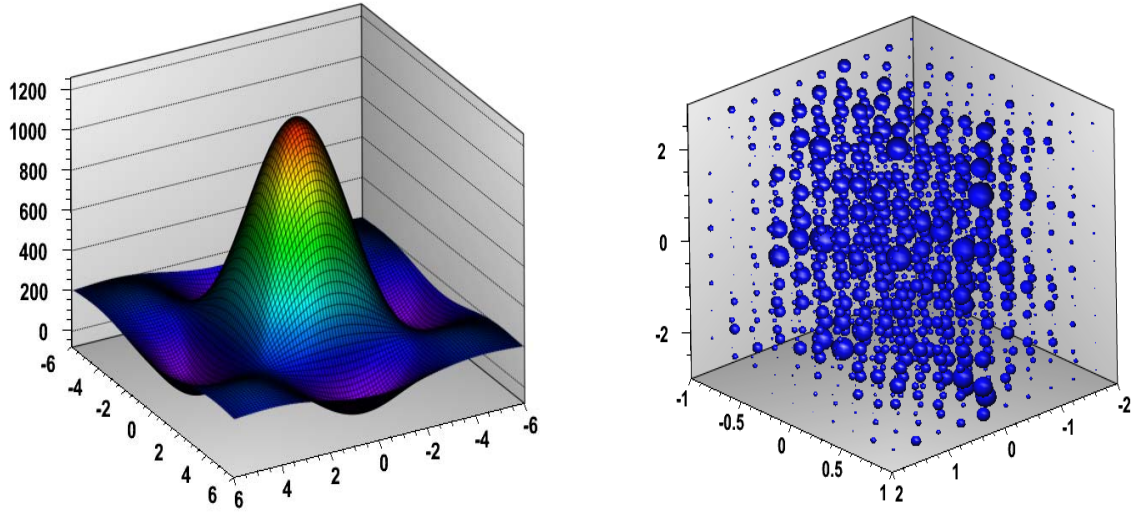
tex->SetTextSize(0.03807391);
tex->SetTextAngle(338.5435);
tex->SetLineWidth(2);
tex->Draw();
c1->Modified();
c1->cd();
c1->SetSelected(c1);
c1->ToggleToolBar();
}

```

Bütün bunların yanında ROOT programının arayüz programları ile kullanılmasıyla 2 ve 3 boyutlu grafikler de elde edilebilir. Bu grafiksel gösterimlere örnekler aşağıda verilmektedir. Çalışmanın niteliğine göre histogram, bar grafiği, dolgu grafiği gibi 2 ve 3 boyutlu grafikler de elde edilebilir.



Şekil 3.6 Boyutsal grafik gösterim örnekleri 1



Şekil 3.7 Boyutsal grafik gösterim örnekleri 2

4. DEDEKTÖR MODELLEME UYGULAMALARI

4.1 PGS ile Modellemeler

Bu bölümde belirli programların yardımıyla bir algıç benzetim arayüz programı olan pythia-pgs'te bir çarpışma prosesi sonucunda ortaya çıkan son durum parçacıklarını elde edip bunların analizlerini yapacağız. Öncelikle MadGraph programının kurulmasıyla işe başlanılabilir. Daha sonra MadGraph programı içerisinde kolaylık sağlaması açısından bir arayüz programı olan pythia-pgs programı ve bu programda yer alan ExRootAnalysis programları doğrudan bilgisayara kurulabilir veya PGS programının belirli bir sürümü de (PGS4) doğrudan kullanılabilir. Pratik olması açısından bu tezde bir arayüz programı olan pythia-pgs programı kullanılmıştır. Bütün bunların yanısıra ROOT programında hali hazırda kurulu olması gerekir. Şimdi MadGraph programında yeni bir çalışma dizini yaratılıp burada gerekli "Cards" kartların düzenlemeleri yapıldıktan sonra pythia-pgs arayüz programı yardımıyla PGS algıç programı çalıştırılmalıdır. Bunun için "Events" olaylar dizininde gerekli olan olay kütükleri yerleştirildikten sonra "../bin/run_pgs" komutunun çalıştırılması yeterli olacaktır. Daha sonra "Events" olayların dizininde .LHCO kütüğü elde edilecektir. Bu kütük ExRootAnalysis programı yardımıyla .root kütüğü haline getirilmelidir. Sonra bu Root kütüğü Root programı yardımıyla analiz edilip çözümlenecektir. Şimdi yukarıda bahsedilen işlem adımları enine momentumları 100GeV jetler (kuarkların dedektörde görülen izlerine jet denir) içeren STDHEP kütükleri 2 şer u, d, s, c, b, g partonlarından oluşmuş 6 olay kütüğünden üretilen uU, dD, sS, cC, bB ve gG jet ikili olay kütükleri için kullanılacaktır. Bir olay döngüsü kurularak, kütük içerisindeki b, c veya hafif olarak işaretlenmiş jetler sayılacak ve bunların doğru ve yanlış işaretleme yüzdeleri bulunacaktır. Yani hangi yüzdelerde b jeti başka bir jetle karıştırılmıştır, o durum belirlenecektir.

Programların kurulum adımları aşağıda verilmiştir.

```
cd MG_ME_V4.4.12/
```

```
tar -zxvf pythia-pgs_V2.0.26.tar.gz
```

```
tar -zxvf ExRootAnalysis_V1.0.6.tar.gz
```

```
cd ExRootAnalysis/
```

```
make
```

PGS (Pretty Good Simulation) Modelleme programında birçok dedektör için dedektörlerin özelliklerinin detaylı olarak tanımlandığı kartlar hazır bir şekilde bulunmaktadır. Bu kartlarda istenilen özelliklerde değişiklikler yapılabilir. Bu değişiklikler yapıldıktan sonra PGS algıç programı çalıştırılabilir.

Çizelge 4.1 PGS girdi kartı

ATLAS	! Parametre grubunun adı
81	! Kalorimetredeki eta hücrelerinin sayısı
63	! Kalorimetredeki phi hücrelerinin sayısı
0.1	! Kalorimetre hücrelerinin eta genişliği eta < 5
0.099733101	! Kalorimetre hücrelerinin phi genişliği
0.01	! EM kalorimetre çözünürlüğü (sabit terim)
0.1	! EM kalorimetre çözünürlüğü (*sqrt(E) terimi)
0.8	! Hadronik kalorimetre çözünürlüğü (*sqrt(E) terimi)
0.2	! Tetikleyicide kayıp ET çözünürlüğü
0.00	! Kalorimetre hücrelerinin verimsiz kenar kalınlığı
cone	! Jet bulma algoritması (cone veya ktjet)
3.0	! Kalorimetre kümeleme azami çekirdek enerjisi (GeV)
0.5	! Kalorimetre kümeleme azami omuz enerjisi (GeV)
0.70	! Kalorimetre kümeleme/ktjet koni büyüklüğü (delta R)
1.0	! İz sürücünün dış yarıçapı (m)
2.0	! Manyetik alan şiddeti (Tesla)
0.000005	! Sagitta çözünürlüğü (m)
0.98	! İz bulma randımanı
0.30	! En düşük iz pT'si (GeV)
2.5	! İz sürme eta kapsamı
3.0	! Elektron/foton eta kapsamı
2.4	! Muon eta kapsamı
2.0	! Tau eta kapsamı

**PGS Girdi
Kartı**

cd deneme/

./bin/newprocess

./bin/generate_events

Yukarıdaki komutlar yardımıyla MadGraph'ta 2'şer partondan elde edilen olay kütüğü kullanılarak pythia-pgs arayüz programıyla modelleme yapılmaktadır.

cd Events/

../bin/run_pythia ; ls

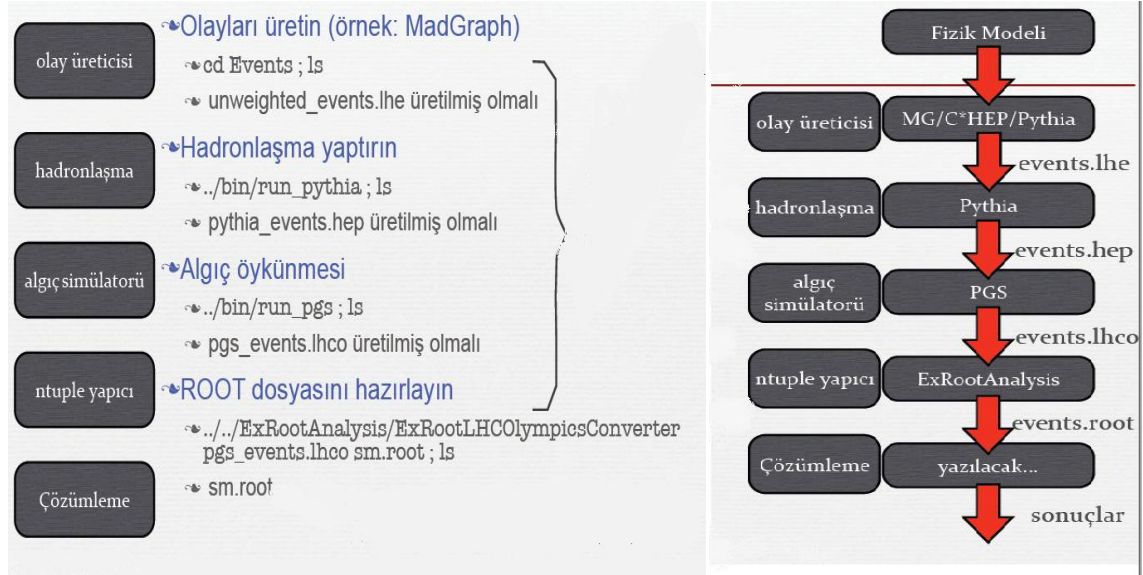
../bin/run_pgs

../../ExRootAnalysis/ExRootLHCOlympicsConverter pgs_events.lhco sm.root

vi sm.root

root sm.root

Çizelge 4.2 PGS ve olay üretim zinciri



Yukarıdaki 4.2 çizelgesinde Algıç öykünmesi kısmında ifade edilen .LHCO uzantılı kütüğün içeriği aşağıdaki çizelge 4.3 de gösterildiği şekildedir.

Çizelge 4.3 LHCO kütüğünün içeriği

LHCO Biçimi

* PGS'in önce LHC Olimpiyatları için hazırlanmış, düz ASCII dizgesindeki, son zamanlarda standart olan çıktı biçimi.

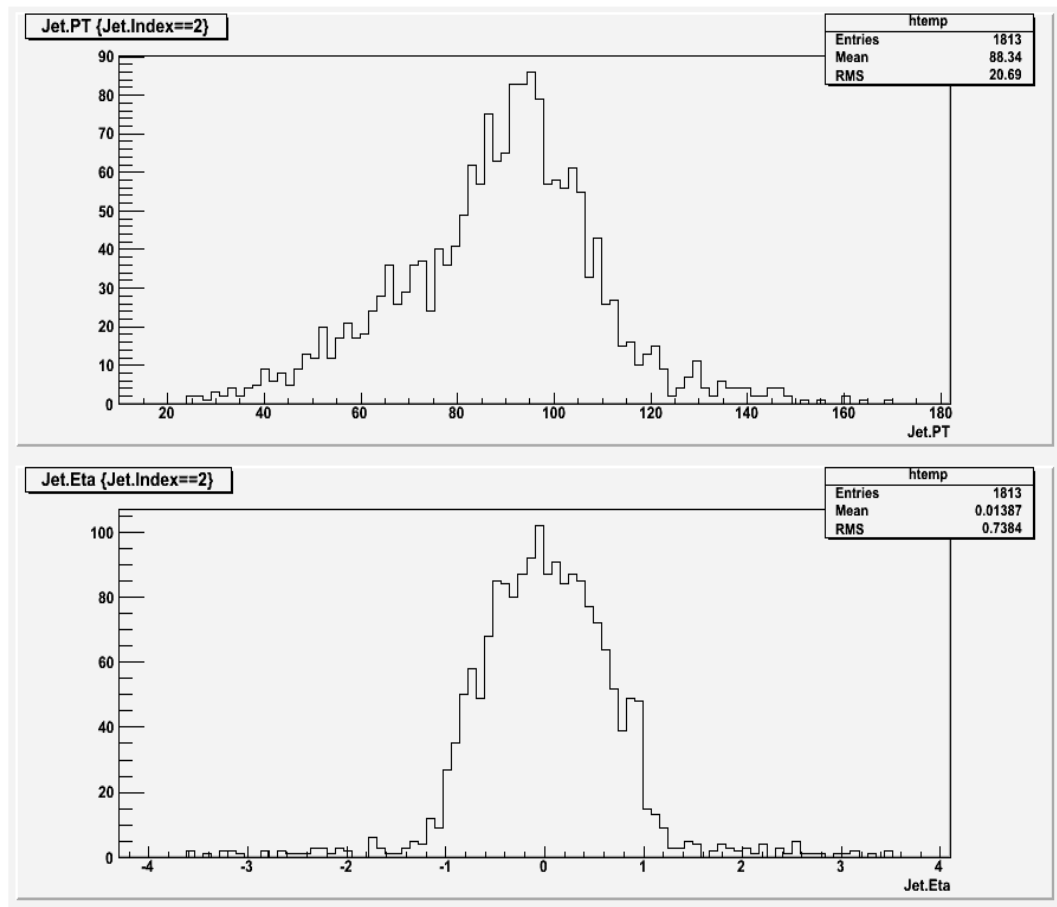
#	tür	eta	phi	pt	jkt1	#iz	btag	had/em	dum1	dum2
0		39	1043							
1	0	-1.350	3.341	26.11	0.00	0.0	0.0	0.00	0.0	0.0
2	1	-0.663	5.233	164.40	0.00	1.0	0.0	0.02	0.0	0.0
3	2	-0.589	4.675	147.62	0.11	-1.0	4.0	95.99	0.0	0.0
4	4	-0.629	4.998	308.94	85.25	12.0	2.0	0.33	0.0	0.0
5	4	-2.061	1.571	455.01	156.56	32.0	0.0	1.19	0.0	0.0
6	4	-1.954	2.699	24.07	2.38	32.0	0.0	11.60	0.0	0.0
7	4	1.149	5.756	6.23	1.93	2.0	0.0	3.73	0.0	0.0
8	6	0.000	3.521	28.86	0.00	0.0	0.0	0.00	0.0	0.0

Burada çizelge 4.3'te ilk sırada yer alan # karakteri altındaki sayılar 0,1,2,3,4,5,6... meydana gelen olay sayılarını göstermektedir. Tür olarak gösterilen kısmın altında kalan sayılar parçacığın türü hakkında bilgi vermektedir. 0 fotonları, 1 elektronları, 2 Muonları, 3 Çok yaygın olarak bilinen Hadronik ışınmaları, 4 jetleri, 6 ise kayıp enerjiyi temsil etmektedir. Bunun dışında yukarıdaki çizelgede eta, phi, pt, jtkl (jet kütlesi), btag, had/em gibi değerlere ait veriler yer almaktadır [29]. Burada tek tek hangi kısımların işaretlendiğine bakıp işaretleme yüzdelerinin hesaplanması yönünde not alınabilir ve sonuca ulaşılabilir ancak bu işlemin bu şekilde yapılması hem sıkıcı hem de inanılmaz derecede vakit kaybına yol açacaktır. Bu yüzden bu işlemi kısa yoldan yani ROOT programı yardımıyla grafiksel hale getirerek çalışmak daha avantajlı olacaktır.

BTAG işaretleme için ROOT programında yazılan kod satırları aşağıda verilmektedir. Öncelikle kullanılacak kütüphaneler tanımlanmaktadır.

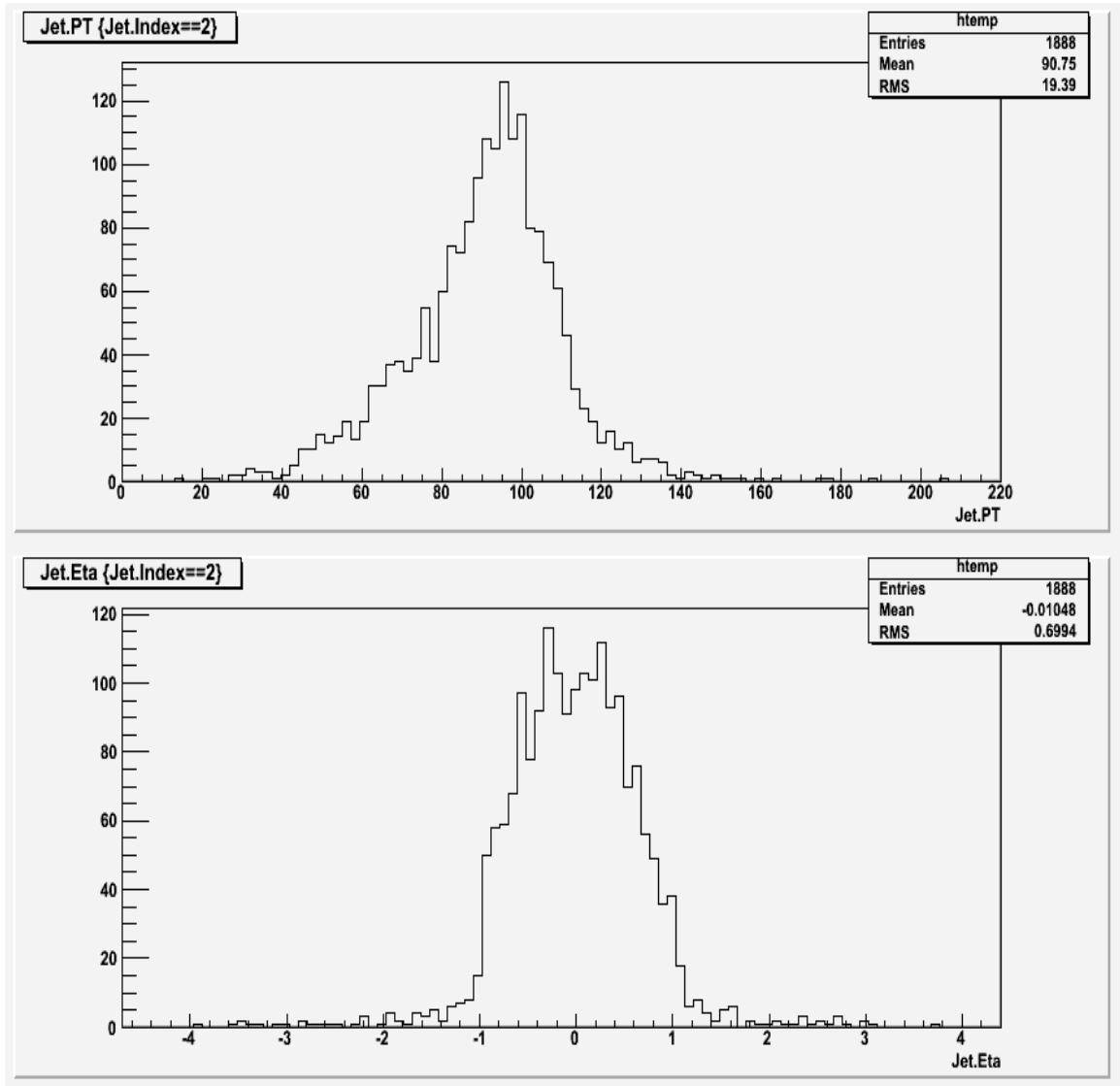
```
#include <iomanip> #include "TTree.h" #include "TStyle.h" #include "TCanvas.h"
#include <iostream> #include "TH1F.h" #include "TFile.h" #include "TF1.h"
#include <math.h> #include "TMath.h" #include "TCut.h" #include "TROOT.h"
#include <fstream> #include "TChain.h"
void jets(char* root_filename="pythia_events_BB.hep.root")
{
char title[200];
char fname[200];
// READ ROOT FILE
sprintf(fname,"%s",root_filename);
TFile *d1= new TFile(fname,"READ");
//JETS
TTree *jets=(TTree*)d1->Get("LHCO");
//CANVAS
sprintf(title,"c_Jet");
TCanvas* c_Jet= new TCanvas(title,title,900,700);
c_Jet->Divide(1,2);
c_Jet->cd(1);
// sadece iki jet li olaylar icin jet pT dagilimi grafik gosterimi
jets->Draw("Jet.PT","Jet.Index==2");
c_Jet->cd(2);
// sadece iki jet li olaylar icin jet Eta dagilimi grafik gosterimi
jets->Draw("Jet.Eta","Jet.Index==2");
}
```

Bu kod 6 olay kütüğü içinde yapılarak grafikler elde edilmiştir. Yukarıdaki kod satırlarında bu işlem `pythia_events_bB.hep.root` adlı olay kütüğü için yapılmıştır. Diğer olay kütükleri `pythia_events_cC.hep.root`, `pythia_events_dD.hep.root`, `pythia_events_gG.hep.root`, `pythia_events_sS.hep.root`, `pythia_events_uU.hep.root` içinde yapılacaktır. Böylelikle her bir kütük için iki jetli olayları seçen ve jetlerin dik momentumlarıyla birlikte pseudo-hızlılıklarını veren grafikler elde edilmiştir.



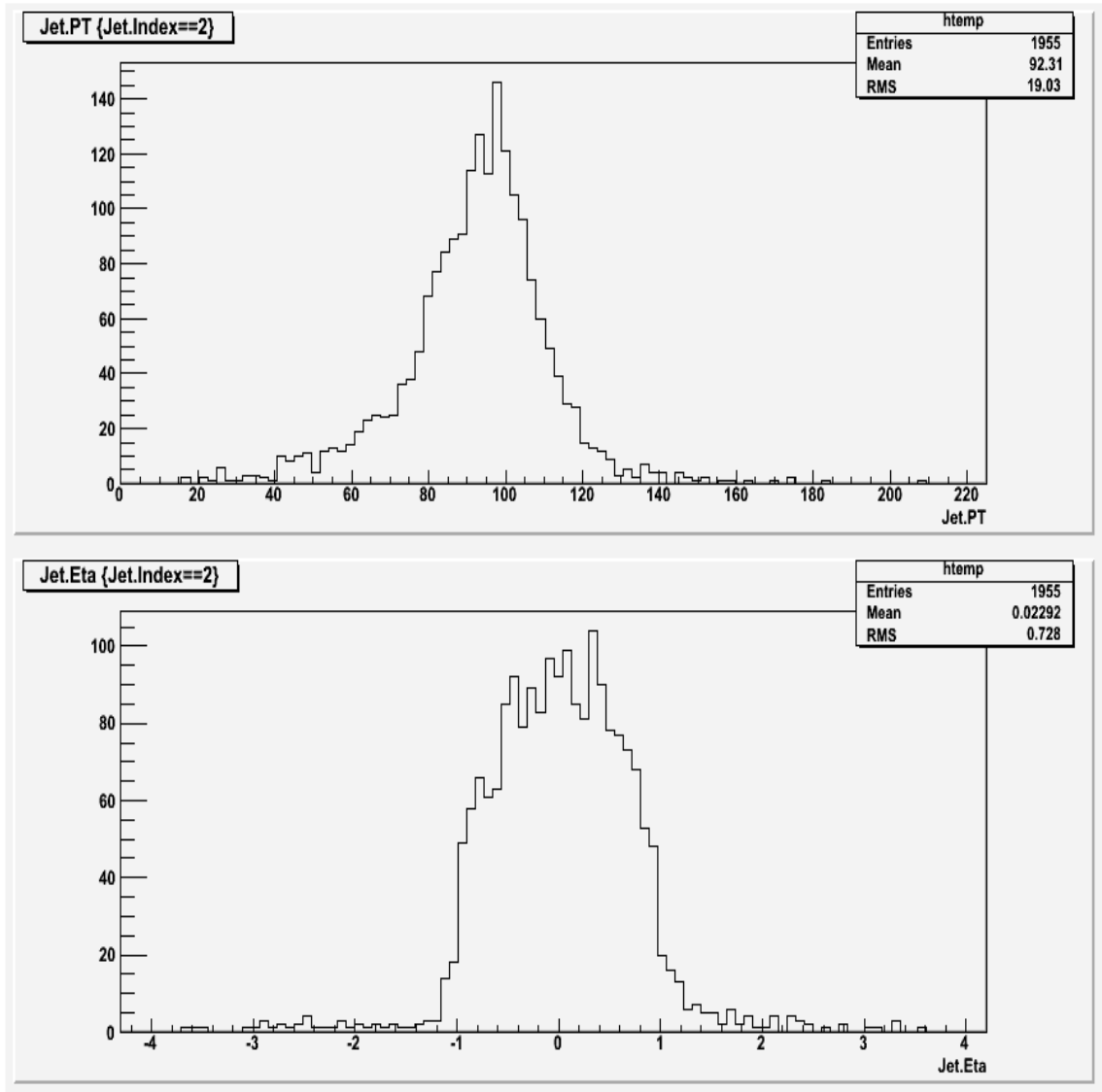
Şekil 4.1 `pythia_events_bB.hep.root` kütüğü için grafikler

Yukarıdaki grafiklerde sadece 2 jetli olaylar için Eta ve pT grafiklerinde “Entries” olay sayısı 1813 olarak gözlenmektedir. Olay sayısı ne kadar fazla olursa istatistikten dolayı elde edilmeye çalışılan verilere ulaşılması o kadar kolay olur. Yine grafiklerde “Mean” ortalama değerler pT grafiği için 88.34 iken Eta grafiği için bu değer 0.01387 olarak hesaplanmıştır. Bir eksandeki değerlerin ortalamasını gösteren “RMS-Root Mean Square” değerleride sırasıyla Eta ve pT için 0.7384 ve 20.69 olarak görülmektedir.



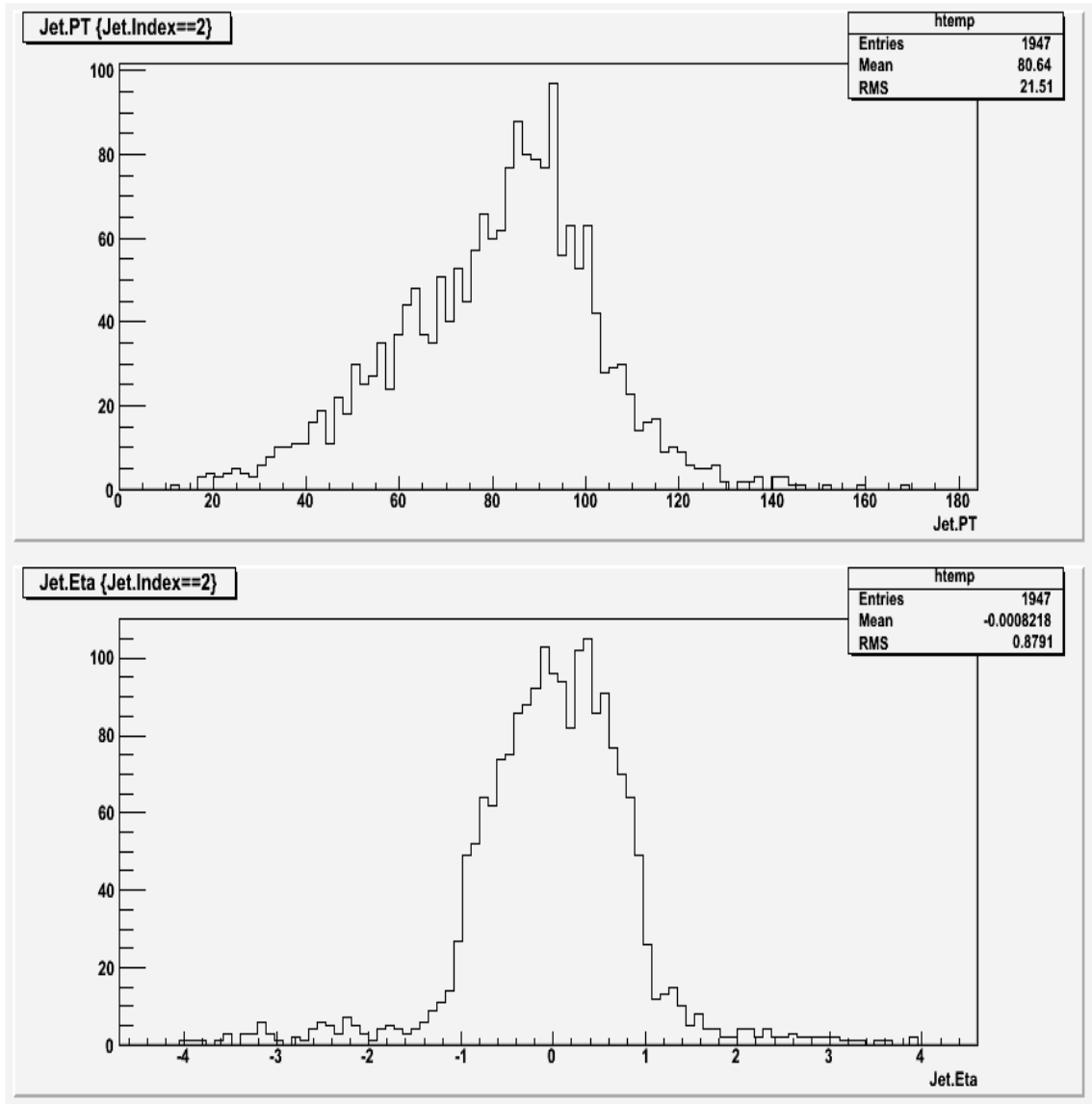
Şekil 4.2 pythia_events_cC.hep.root kütüğü için grafikler

Yukarıdaki grafiklerdeki “Entries” girilen değerler 1888 iken, “Mean” ortalama değerleri dik momentum pT için 90.75 Eta içinse -0.01048 dir. RMS değerleri ise pT için 19.39 Eta içinse 0.6994 olarak görülmektedir.



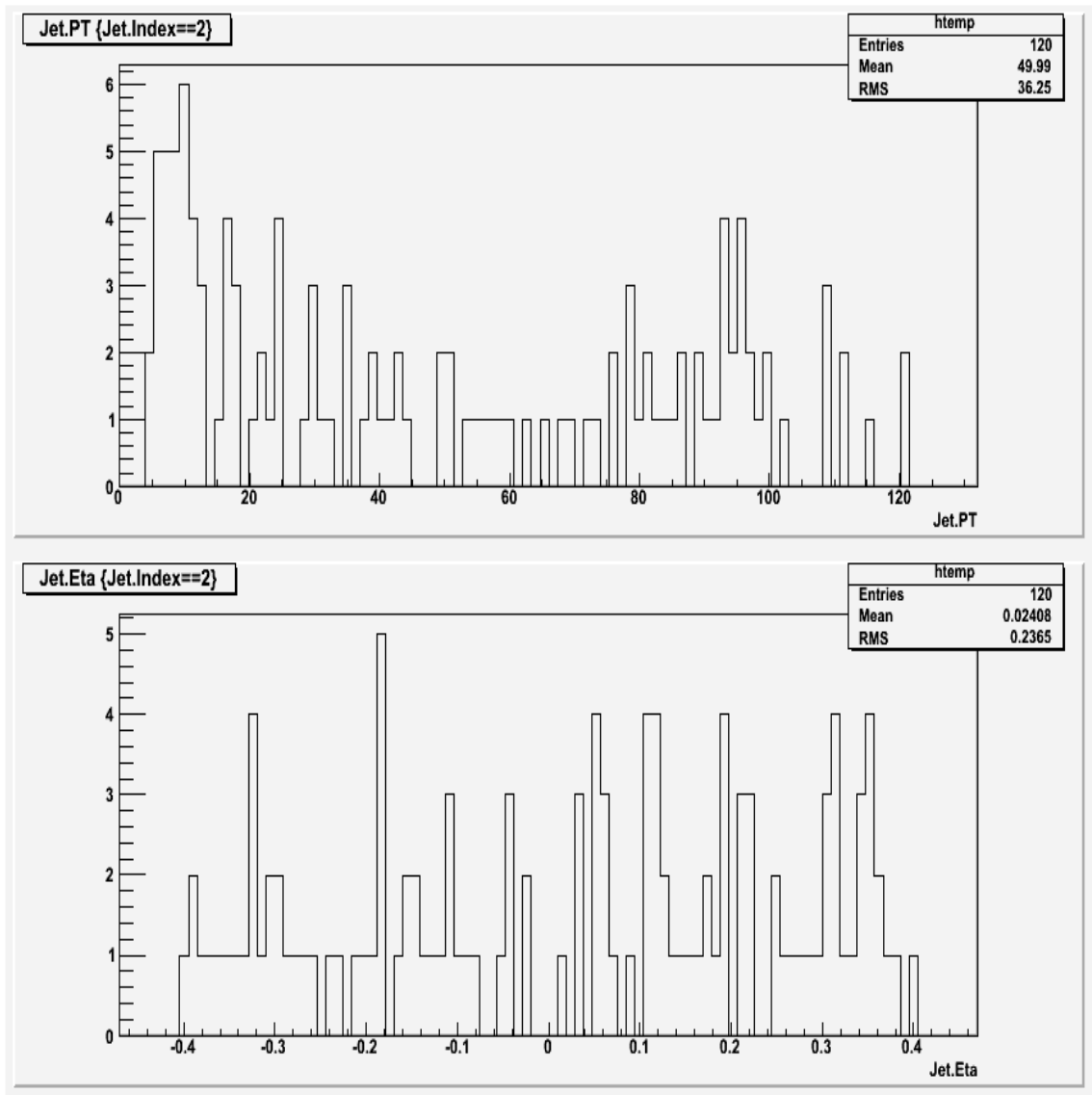
Şekil 4.3 pythia_events_dD.hep.root kütüğü için grafikler

pythia_events_dD.hep.root kütüğü için elde edilen grafiklerdeki Entries değerleri 1955 iken, Mean değerleri dik momentum pT için 92.31 Eta içinse 0.02292 dir. RMS değerleri ise pT için 19.03 Eta içinse 0.728 olarak görülmektedir.



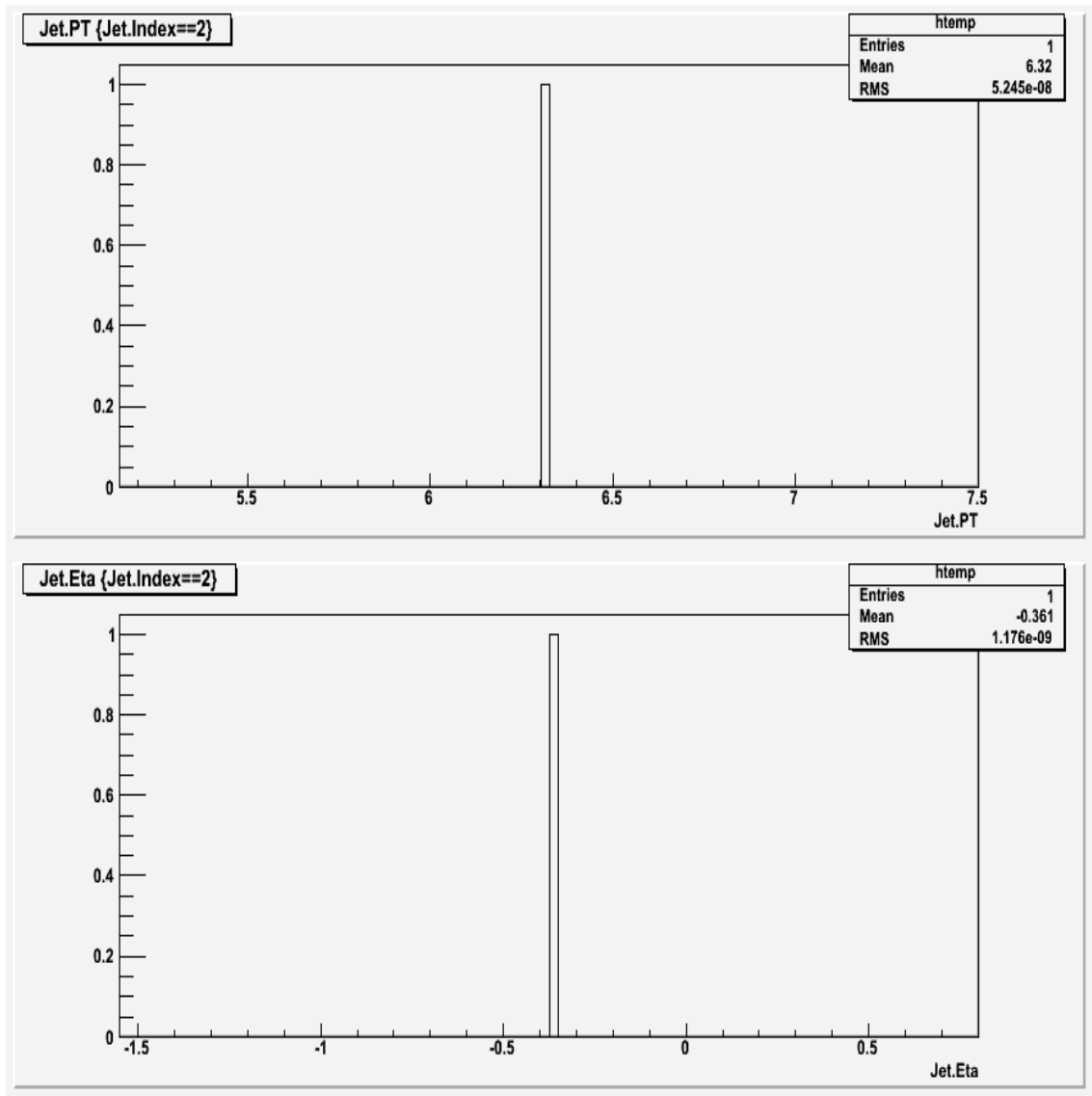
Şekil 4.4 pythia_events_gG.hep.root kütüğü için grafik

pythia_events_gG.hep.root kütüğü için Entries değerleri 1947 iken, Mean ortalama değerleri dik momentum pT için 80.64 Eta içinse -0.0008218 dir. RMS değerleri ise pT için 21.51 Eta içinse 0.8791 olarak görülmektedir.



Şekil 4.5 pythia_events_sS.hep.root kütüğü için grafik

Grafiğe ait Entries değerleri 120 iken, Mean ortalama değerleri dik momentum pT için 49.99 Eta içinse 0.02408'dir. RMS değerleri ise pT için 36.25 Eta içinse 0.2365 olarak görülmektedir.



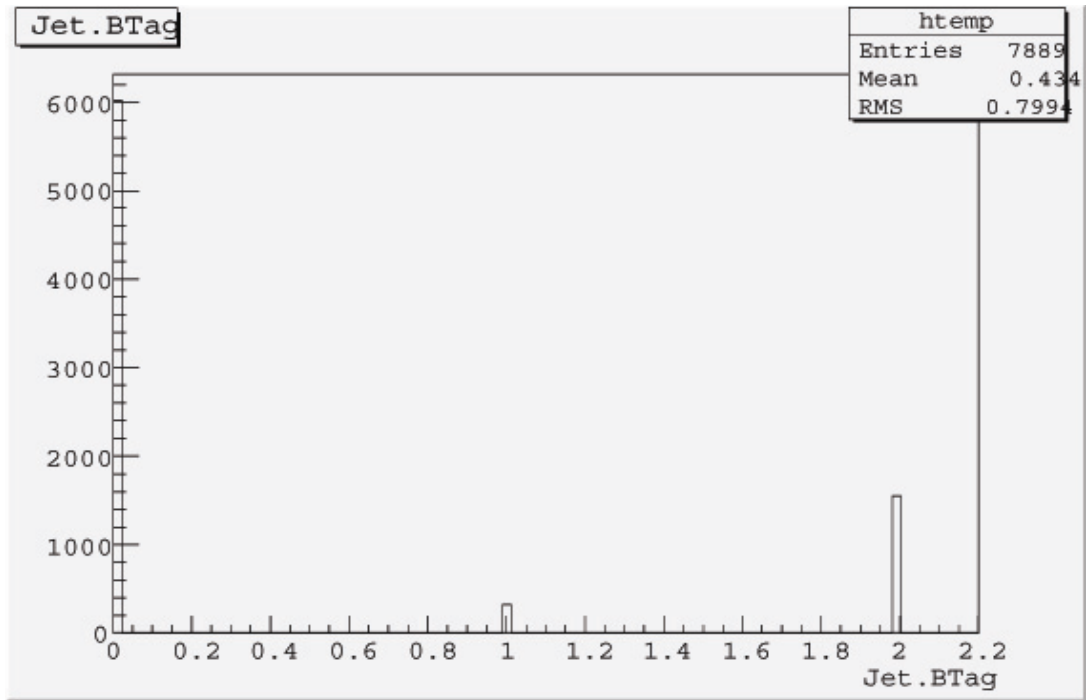
Şekil 4.6 pythia_events_uU.hep.root kütüğü için grafik

Grafiğe ait Entries değerleri 1 iken, Mean ortalama değerleri dik momentum pT için 6.32 Eta içinse -0.361'dir. RMS değerleri ise pT için 5.245e-08, Eta içinse 1.176e-09 olarak görülmektedir.

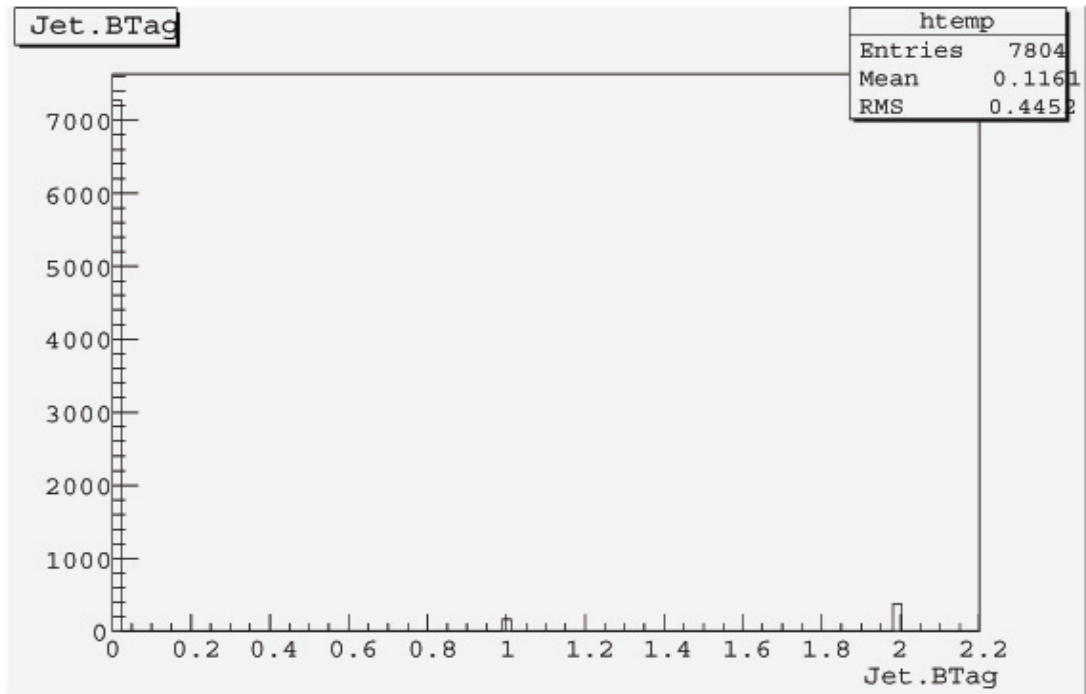
```

#include <iomanip>    #include "TTree.h"    #include "TFile.h"    #include "TH1F.h"
#include <iostream>    #include <math.h>    #include "TF1.h"    #include "TCut.h"
#include <fstream>    #include "TMath.h"    #include "TStyle.h"    #include "TChain.h"
#include "TROOT.h"    #include "TCanvas.h"
void jets(char* root_filename="pythia_events_bB.hep.root")
{
//gROOT->ForceStyle();
char title[200];
char fname[200];
// READ ROOT FILE
sprintf(fname,"%s",root_filename);
TFile *d1= new TFile(fname,"READ");
//*****
//                JETS
TTree *jets=(TTree*)d1->Get("LHCO");
//CANVAS
sprintf(title,"olaysayisi_Jet");
TCanvas *olaysayisi_Jet= new TCanvas(title,title,500,300);
olaysayisi_Jet->Divide(1);
olaysayisi_Jet->Divide(2);
olaysayisi_Jet->Divide(3);
olaysayisi_Jet->cd(1);
jets->Draw("Jet.Mass","Jet.BTag==0");
olaysayisi_Jet->cd(2);
jets->Draw("Jet.Mass","Jet.BTag==1");
olaysayisi_Jet->cd(3);
jets->Draw("Jet.Mass","Jet.BTag==2");
}

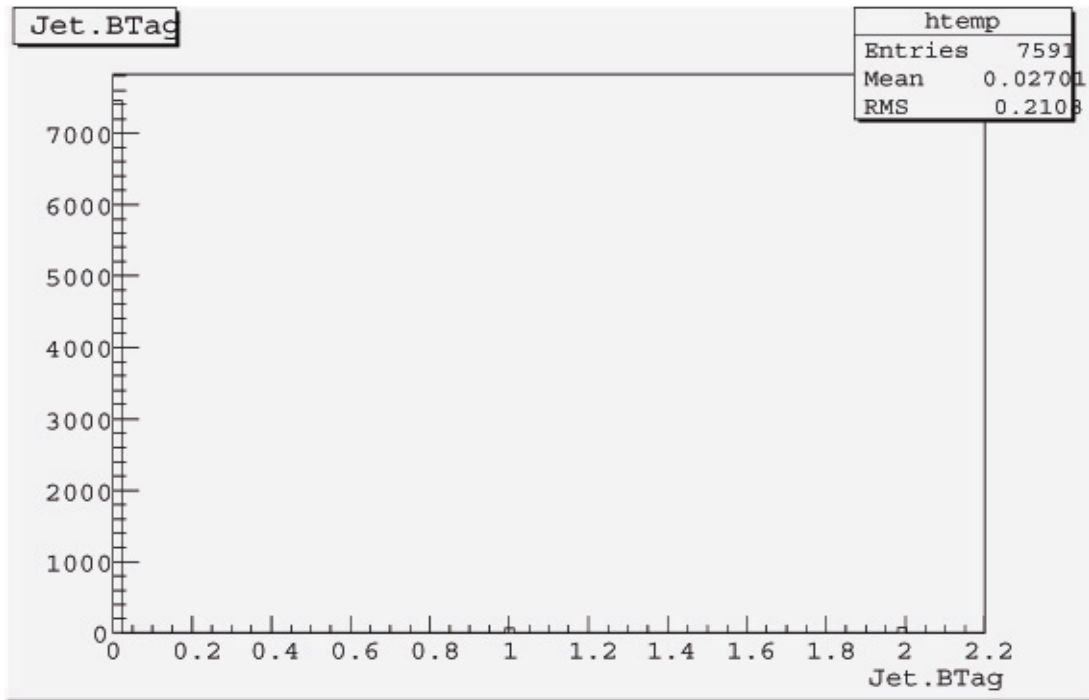
```



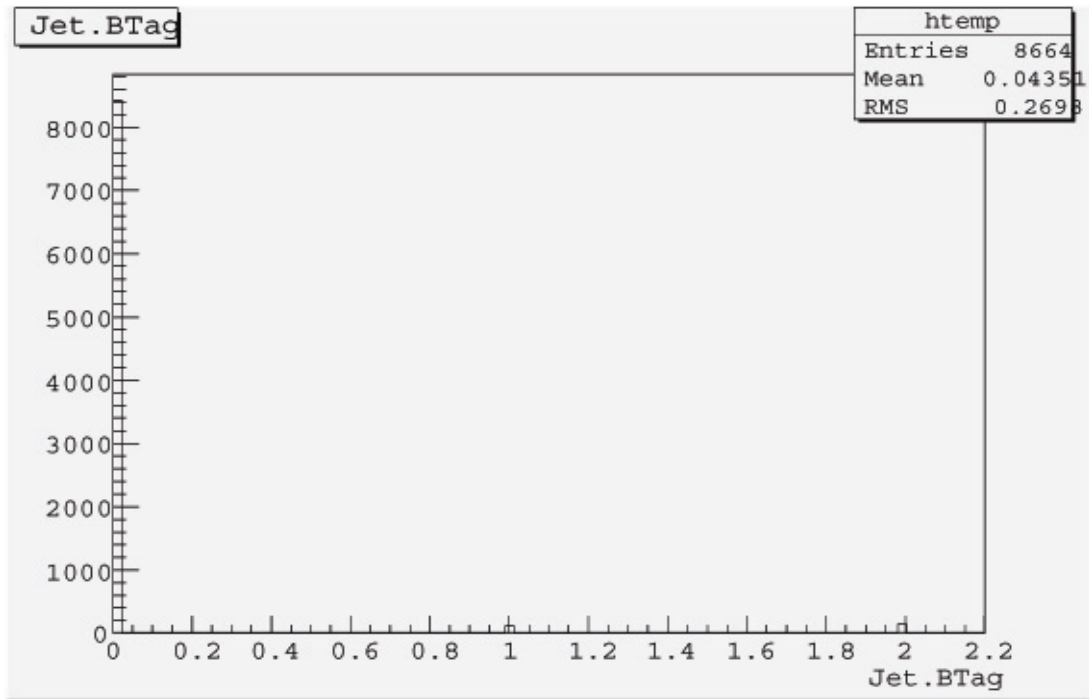
Şekil 4.7 bB_BTAG işaretleme oranları grafiği



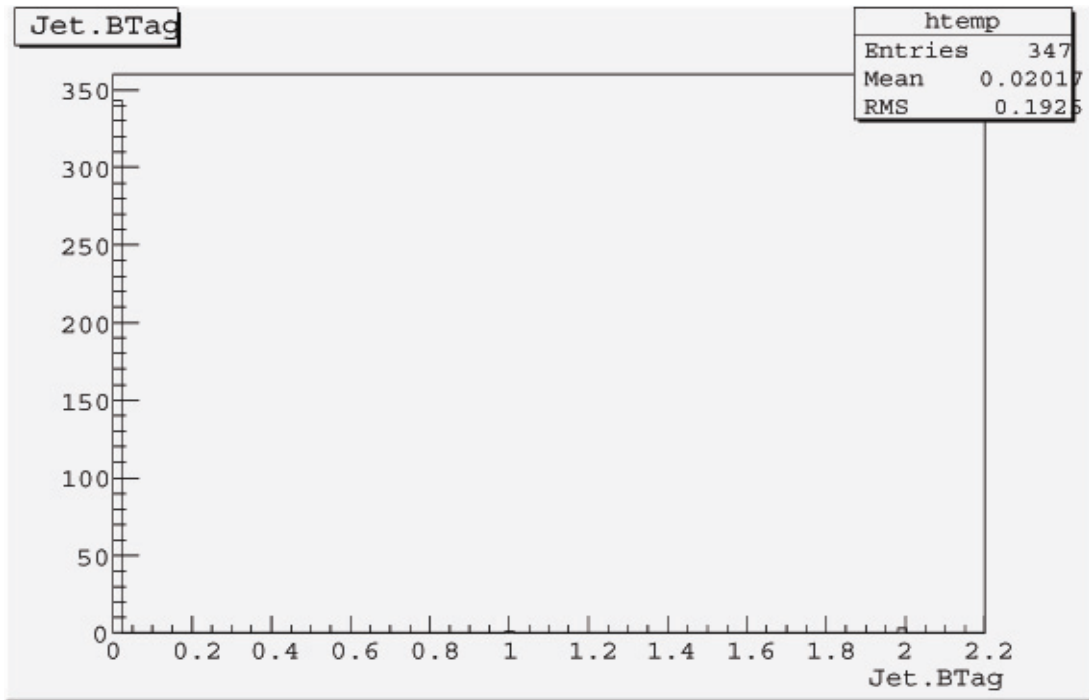
Şekil 4.8 cC_BTAG işaretleme oranları grafiği



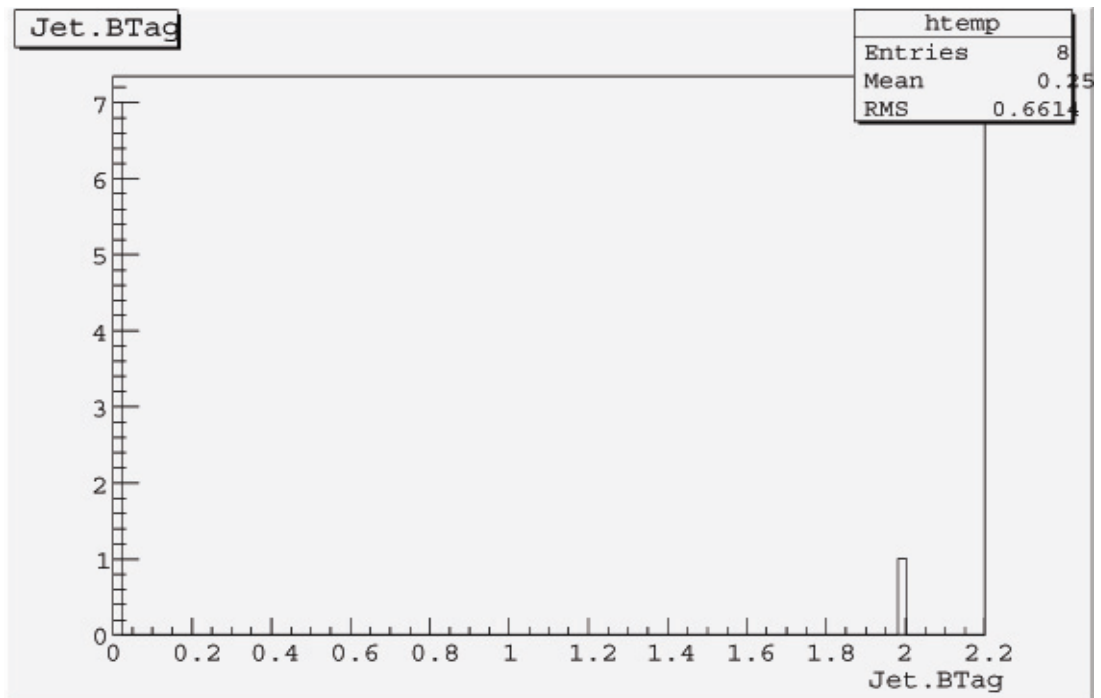
Şekil 4.9 dD_BTAG işaretleme oranları grafiği



Şekil 4.10 gG_BTAG işaretleme oranları grafiği



Şekil 4.11 sS_BTAG işaretleme oranları grafiği



Şekil 4.12 uU_BTAG işaretleme oranları grafiği

Bir olay döngüsü kurarak kütükler içerisindeki b, c veya hafif olarak işaretlenmiş jetleri sayarak elde edilen verilerden yararlanılıp yine bir olay örgüsü kurulup yukarıdaki doğru ve yanlış işaretleme yüzdelerinin bulunacağı grafikler elde edilmiş oldu. Bu grafikler incelenirken bilinmesi gereken grafiklerde verilerin oluşturduğu aralıkların ne anlama geldiğidir. Burada 0 bölgesindeki verilerin oluşturduğu kısım, yanlış işaretlenme oranını vermektedir. 1 değeri ise serbest jetler için işaretlemeyi gösterirken, 2 ise serbest jetler için doğru işaretleme değerini vermektedir.

Bu grafikler incelendiğinde, ilk iki grafiğe bakılacak olursa şekil 4.7 grafiğinde toplam olay sayısı 7889 iken b-jet için yanlış işaretleme yüzdesi ise yaklaşık olarak ~%76 civarındadır. Böylece doğru işaretleme yüzdesinin ~%20 civarında olduğu görülür. Bu değerler hesaplanırken yani yanlış işaretleme yüzdesi bulunurken, toplam olay sayısı ve yanlış işaretleme miktarından yararlanarak (0 bölgesindeki miktar üzerinden) yüzde oranı üzerinden yanlış hesaplama yüzdesi hesaplanmaktadır. Aynı şekilde doğru işaretleme yüzdesi bulunurken yine toplam olay sayısı ve doğru işaretleme miktarı (2 bölgesindeki miktar üzerinden) kullanılarak doğru işaretleme yüzdesi hesaplanır. Her bir grafik için bu işlemler yapılır ve böylece b, c ve hafif jetler için doğru ve yanlış işaretleme yüzdeleri bulunmuş olur. Diğer şekillere bakılacak olursa şekil 4.8'deki c-jet için durum b jete göre daha düşük değerlerdedir ve toplam olay sayısı 7804 iken yanlış işaretleme yüzdesi ~%92 civarındayken doğru işaretleme yüzdesi ise ~%1.92 civarındadır.

Diğer hafif jetler için bu durum sırasıyla d, g, s ve u için, doğru ve yanlış işaretleme yüzdeleri:

d için yanlış işaretleme yüzdesi ~%97 iken doğru işaretleme yüzdesi ~%4 civarındadır.

g için yanlış işaretleme yüzdesi ~%96 iken doğru işaretleme yüzdesi ~%3.5 civarındadır.

s için yanlış işaretleme yüzdesi ~%98 iken doğru işaretleme yüzdesi ~%1.4 civarındadır.

u için yanlış işaretleme yüzdesi ~%87 iken doğru işaretleme yüzdesi ~%15 civarındadır.

Bu durum hem istatistiğin az olmasından hem de jetlerin tekil olarak günümüz bigilerine göre halen iyi bir çözünürlükte gözlenemediği sonucuna dayanır. Tekil olarak kuarkları gözlemek için daha yüksek enerjilere ve daha farklı dedektör yapılarına ihtiyaç vardır ve bu çıkan sonuçlardan görülmektedir.

4.2 Athena İle Uygulamalar

Burada örnek olarak $\sim 100 Z \rightarrow \mu\text{on}$, μon bozunumu sürecine bakılarak, ışınlar çalışılarak, çıktı dosyaları elde edilecek ve incelenecektir. Eğer bu kısımda bu komutlar ilk kez kullanılıyorsa “csc_evgen08new_trf.py -h” sayesinde komutlar hakkında yardım alınabilir. Şimdi “ $Z \rightarrow \mu\text{on}$, μon ” ’nun iş seçimi için gerekli olan dosyalar elde edilmelidir. Bunun için “get_files” komutu ile aşağıda belirtilen komut satırını uygulamak yeterli olacaktır.

```
get_files -jo MC8.105145.PythiaZmumu.py
```

Daha sonra MC8.105145.PythiaZmumu.py dosyası istenilen editör yardımıyla açılıp içerisine alt kısmına “evgenConfig.minevents = 100” komut satırı eklenir. Bu komut satırı olay sayısını vermektedir ki burada 100 olay için bu işlemler yapılmaktadır.

Artık program çalıştırılabilir. Bu işlem çok kısa olup sadece birkaç dakikada sonuç dosyaları elde edilebilir.

```
csc_evgen08new_trf.py 105145 1 -1 1324354657 ./MC8.105145.PythiaZmumu.py  
/tmp/$USER/105145.pool.root
```

Üretilen olay çıktıları basit bir ATHENA algoritması yazılarak sonucun doğruluğu çek edilebilir. İhtiyaç duyulan dosya çalışma alanına kopya edildikten sonra içerisindeki girdi dosyasında, “mc.event.pool” olan dosyanın ismi “/tmp/user_name/105145.pool.root” olarak değiştirilmelidir.

```
get_files -jo readGenEventFromPool.py
```

```
athena readGenEventFromPool.py > dumpmc.log
```

```
Sun Aug 16 01:25:25 CEST 2009
```

```
Preloading tcmalloc.so
```

```
Py:ConfigurableDb INFO Read module info for 4034 configurables from 9 genConfDb files
```

```
Py:ConfigurableDb INFO No duplicates have been found: that's good !
```

```
Py:Athena INFO including file "AthenaCommon/Bootstrap.py"
```

```
Py:Athena INFO including file "AthenaCommon/Atlas.UnixStandardJob.py"
```

```
Py:Athena INFO including file "readGenEventFromPool.py"
```

```
Py:Athena INFO including file "AthenaCommon/runbatch.py"
```

```
ApplicationMgr INFO Updating ROOT::Reflex::PluginService::SetDebug(level) to level=0
```

```
ApplicationMgr SUCCESS
```

```
=====
```

Welcome to ApplicationMgr \$Revision: 1.77 \$
 running on lxplus203.cern.ch on Sun Aug 16 01:25:36 2009

```

=====
ApplicationMgr  INFO Successfully loaded modules : AthenaServices
ApplicationMgr  INFO Application Manager Configured successfully
ApplicationMgr  INFO Updating ROOT::Reflex::PluginService::SetDebug(level) to level=0
ServiceManager  DEBUG Initializing service StatusCodeSvc
StatusCodeSvc   INFO initialize
ServiceManager  DEBUG Initializing service DataModelCompatSvc
DataModelCompatSvc  DEBUG Service base class initialized successfully
.....
.....
.....
.....
.....
.....
.....
.....
/tmp/ymehmet/105145.pool.root Debug --> Deaccess DbContainer  READ    [ROOT_All] ##Links
/tmp/ymehmet/105145.pool.root Debug --> Deaccess DbContainer  READ    [ROOT_All] ##Shapes
/tmp/ymehmet/105145.pool.root Debug --> Deaccess DbContainer  READ    [ROOT_All] ##Params
RootDBase.close Debug I/O READ  Bytes: 243629
RootDBase.close Debug I/O WRITE Bytes: 0
RootDBase.close Debug I/O OTHER Bytes: 0
Domain[ROOT_All] Info -> Deaccess DbDatabase  READ    [ROOT_All] A4D464E9-B886-DE11-
9658-000423D664CE
/tmp/ymehmet/105145.pool.root Debug --> Deaccess DbContainer  READ    [ROOT_Tree]
CollectionTree(EventInfo_p3/McEventInfo)
/tmp/ymehmet/105145.pool.root Debug --> Deaccess DbContainer  READ    [ROOT_Tree]
CollectionTree(McEventCollection_p4/GEN_EVENT)
/tmp/ymehmet/105145.pool.root Debug --> Deaccess DbContainer  READ    [ROOT_Tree]
POOLContainer_DataHeader
Domain[ROOT_All] Info > Deaccess DbDomain  READ    [ROOT_All]
reaccessed.mc.event.pool Info Database being retired...
reaccessed.mc.event.pool Debug --> Deaccess DbContainer  CREATE [ROOT_All] ##Links
reaccessed.mc.event.pool Debug --> Deaccess DbContainer  CREATE [ROOT_All] ##Shapes
reaccessed.mc.event.pool Debug --> Deaccess DbContainer  CREATE [ROOT_All] ##Params
RootDBase.close Debug I/O READ  Bytes: 0

```

```

RootDBase.close Debug I/O WRITE Bytes: 244209
RootDBase.close Debug I/O OTHER Bytes: 0
Domain[ROOT_All] Info -> Deaccess DbDatabase CREATE [ROOT_All] 42B4006C-2088-DE11-
BADF-000423D986F4
reaccessed.mc.event.pool Debug --> Deaccess DbContainer CREA/UPDA [ROOT_Tree]
CollectionTree(EventInfo_p3/McEventInfo)
reaccessed.mc.event.pool Debug --> Deaccess DbContainer CREA/UPDA [ROOT_Tree]
CollectionTree(McEventCollection_p4/GEN_EVENT)
reaccessed.mc.event.pool Debug --> Deaccess DbContainer CREA/UPDA [ROOT_Tree]
POOLContainer_DataHeader
Domain[ROOT_All] Info > Deaccess DbDomain UPDATE [ROOT_All]
XMLFileCatalog Info There were no updates in the catalog
PoolXMLFileCatalog Info XercesC termination number 0
.....
.....
.....
.....
.....
.....
.....
.....
.....
ChronoStatSvc.f... INFO Service finalized succesfully
ServiceManager DEBUG Finalizing service DataModelCompatSvc
ServiceManager DEBUG Finalizing service StatusCodeSvc
EventSelector DEBUG Calling destructor
IoComponentMgr DEBUG Calling destructor
MetaDataSvc DEBUG Calling destructor
CoreDumpSvc DEBUG Calling destructor
AthenaSealSvc DEBUG Calling destructor
AthDietLoaderSvc DEBUG Calling destructor
ApplicationMgr INFO Application Manager Finalized successfully
ApplicationMgr INFO Application Manager Terminated successfully
Py:Athena INFO leaving with code 0: "successful run"

```

Daha önce elde edilen pool dosyası,

checkFile.py /tmp/ymehmet/105145.pool.root çek edilebilir

Dedektör simülasyonu ve yeniden üretimi için “csc_simul_trf.py” komutu kullanılabilir. Şimdi “/tmp/ymehmet” dizinine gidilerek dedektör simülasyonu yapılmaktadır. Son olarak “ntuple.root” dosyası programın çalışması sonrasında elde edilerek analizler yapılmaya başlanabilir.

```
cd /tmp/ymehmet
```

```
csc_simul_trf.py 105145.pool.root hits.pool.root rdo.pool.root 2 0 1324354656 ATLAS-  
GEO-06-00-00 100 1000
```

```
csc_reco_trf.py rdo.pool.root esd.pool.root aod.pool.root ntuple.root 2 0 ATLAS-GEO-06-  
00-00 DEFAULT
```

```
checkFile.py
```

Yukarıdaki komutlarla yapılanlar, yani üretilen dosyalar çek edilebilir.

Şimdi ATHENA kullanılarak farklı bir süreç için, bir işin fizik analizi şu şekilde yapılabilir.

Çalışma alanı yeniden ayarlanmalıdır. Ve çalışma alanına gidilerek çalışma paketleri çek edilmelidir.

```
cd $TestArea
```

Örnek olarak “J/psi → müon, müon” süreci için ATHENA çalıştırılıp analiz yapılabilecektir ancak bundan önce gerekli olan dosyalar çalışma alanına kopyalanmalıdır.

```
cd PhysicsAnalysis/AnalysisCommon/UserAnalysis/UserAnalysis
```

```
cd ../src/
```

```
cp /afs/cern.ch/atlas/maxidisk/d36/jpsi_mumu/AnalysisSkeleton.cxx .
```

```
cd ../cmt/
```

```
cp /afs/cern.ch/atlas/maxidisk/d36/jpsi_mumu/requirements .
```

ve dosyalarımızı derleyebiliriz artık,

```
cd ../cmt
```

```
cmt config
```

```
gmake
```

Şimdi de çalışma için gerekli olan iş yazılımlarının çalışma alanına kopyalanması gereklidir. Sonrasında iş şu şekilde çalıştırılır.

```

cd ../run
cp /afs/cern.ch/atlas/maxidisk/d36/jpsi_mumu/filelist.py .
cp /afs/cern.ch/atlas/maxidisk/d36/jpsi_mumu/jobOptions.py .
athena jobOptions.py

```

Böylece “AnalysisSkeleton.root” dosyası elde edilir. Elde edilen bu root uzantılı dosyadan sonra analiz macro oluşturulmalıdır. ROOT analiz programı çalıştırılıp histogramlar elde edilip analiz yapılmaya başlanabilir.

```

root
.x rootScript.C
{
// Style setting
gROOT->Reset();
gROOT->SetStyle("Plain");
gStyle->SetFillColor(0);
gStyle->SetFuncColor(kRed);
gStyle->SetFuncWidth(1.0);
gStyle->SetOptStat(11111);
gStyle->SetOptFit(11111);
// Make a canvas
TCanvas c1;
// ROOT-blah-blah needed for release 14
//gSystem->Load("libCintex");
//ROOT::Cintex::Cintex::Enable();

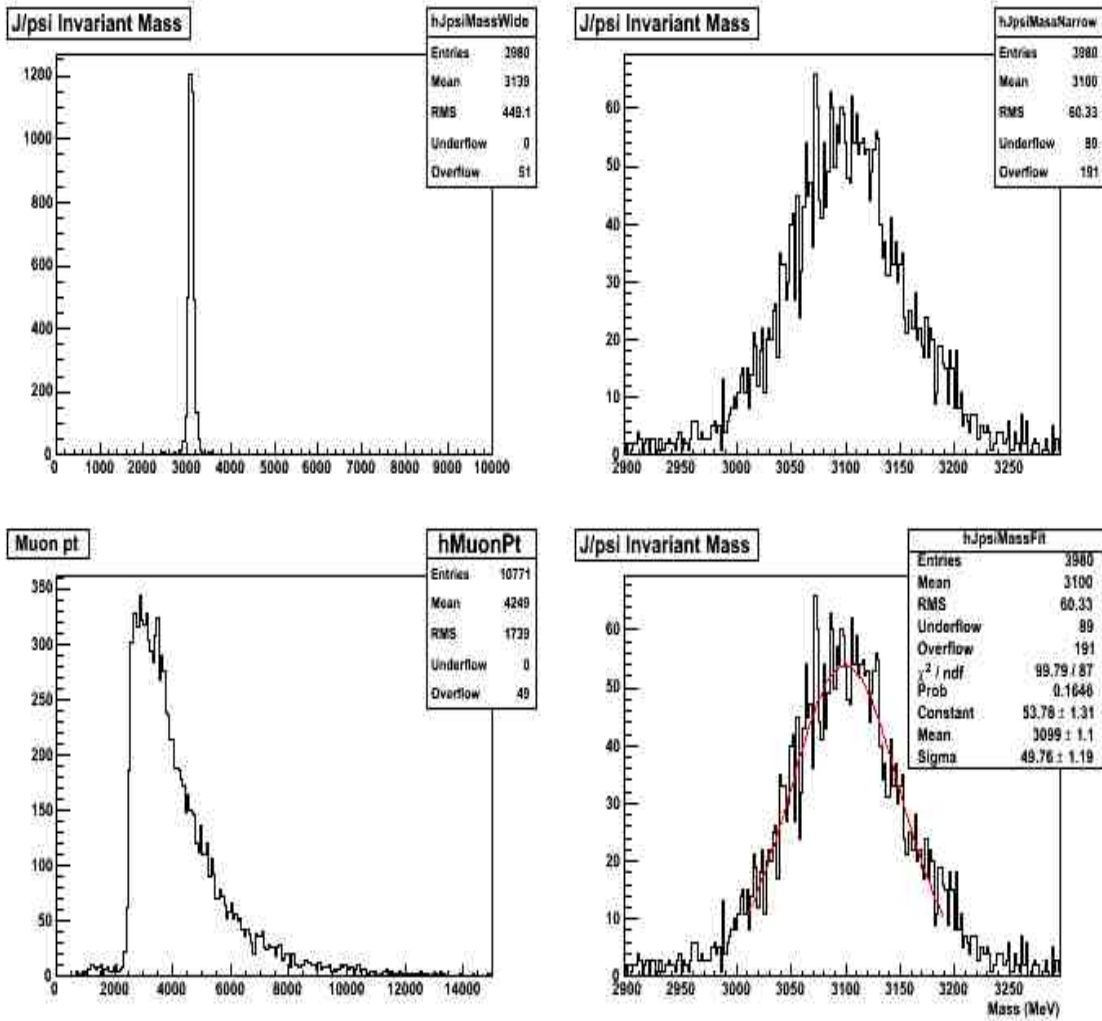
// Open file
TFile f("AnalysisSkeleton.root");
// Get the number of entries in branch
Int_t nEntriesNtuple = (Int_t)MYTREE->GetEntries();
std::cout << "Number of entries in the n-tuple is " << nEntriesNtuple << std::endl;
// Declare variables to plot
std::vector<double>* m_jpsiMass;
std::vector<double>* m_muonPt;
// Assign variable to the n-tuple
MYTREE->SetBranchAddresses("DiMuonInvariantMass",&m_jpsiMass);
MYTREE->SetBranchAddresses("MuonPt",&m_muonPt);
// Declare histograms

```

```

    TH1F *hJpsiMassWide = new TH1F( "hJpsiMassWide", "J/psi Invariant Mass",200, 0.0,
10000.0 );
    TH1F *hJpsiMassNarrow = new TH1F( "hJpsiMassNarrow", "J/psi Invariant Mass",200,
3096.92-200.0, 3096.92+200.0 );
    TH1F *hJpsiMassFit = new TH1F( "hJpsiMassFit", "J/psi Invariant Mass",200, 3096.92-
200.0, 3096.92+200.0 );
    TH1F *hMuonPt = new TH1F( "hMuonPt", "Muon pt",200, 0.0, 15000.0 );
    TH1F *hMuonEta = new TH1F( "hMuonEta", "Muon Eta",200, 0.0, 3.0 );
    // Fill histograms with numbers
    for(Int_t i=0; i<nEntriesNtuple; ++i) {
        MYTREE->GetEntry(i); // Fills all assigned variables in one go, for ith event
        int nMuons = m_muonPt->size();
        int nJpsis = m_jpsiMass->size();
        for (Int_t j = 0; j<nJpsis; ++j) {
            hJpsiMassWide->Fill((*m_jpsiMass)[j]);
            hJpsiMassNarrow->Fill((*m_jpsiMass)[j]);
            hJpsiMassFit->Fill((*m_jpsiMass)[j]);
        }
        for (Int_t k=0; k<nMuons; ++k) {
            hMuonPt->Fill((*m_muonPt)[k]);
        }
    }
    // Draw histograms
    c1.Divide(2,2);
    c1.cd(1);
    hJpsiMassWide->Draw();
    c1.cd(2);
    hJpsiMassNarrow->Draw();
    c1.cd(3);
    hMuonPt->Draw();
    c1.cd(4);
    hJpsiMassFit->Draw();
    // Do fit
    hJpsiMassFit->Fit("gaus","Q","",",3010,3190);
    hJpsiMassFit->GetXaxis()->SetTitle("Mass (MeV)");
    gStyle->SetOptFit(111111);
    hJpsiMassFit->Draw();
}

```



Şekil 4.13 $J/\psi \rightarrow$ müon, müon sürecinin ROOT histogramları

Yukarıdaki $J/\psi \rightarrow$ müon, müon sürecine ait “ J/ψ mass” dar ve geniş kütle dağılımına sahip grafikleri, muon p_T 'sine ait grafiği ve son olarak ta J/ψ kütle dağılımına sahip matematiksel fonksiyonlara uyarlanmış, fit edilmiş ROOT histogramları elde edilmiş olur.

4.3. COMPHEP ile $P,P \rightarrow t_p, T_p$ Bozunumu İçin Yapılan Modelleme Çalışmaları

CompHEP (Computational High Energy Physics - Hesaplamalı Yüksek Enerji Fiziği), bozunum ve saçılma durumlarında parçacık işlemleri için deneylerle elde edilebilir nicelikleri yüksek enerjilerde modelleme yaparak hesaplayan bir programdır. Program Moskova State Üniversitesinde geliştirilmiştir [30]. CompHEP'in kullanımı basittir ve parçacık fiziğini incelemede teorik hesaplarla uyumlu sonuçlar vermektedir. CompHEP, güçlü, kullanımı hızlı fakat bazı limitleri olan bir programdır.

CompHEP'in parçacık içeriğinde: SM, Unitary ve Feynmann ayarları; MSSM (Minimal Supersymmetric Model - Minimum Süpersimetrik Model), Unitary ve Feynmann ayarı; SUGRA ve GMSB, Unitary ayarı gibi modeller vardır. CompHEP'te değişkenlerin, bilinen parçacıkların özelliklerinin tanımlandığı, Lagrangian ifadesinin yazıldığı, ve bu lagrangian'da etkileşen parçacıkların kendilerinin ve ara parçacıklarının tanımlandığı, Lagrangian'da köşe faktörlerinin belirtildiği - Lorentz Çarpanının yazılabildiği kısımlar vardır. CompHEP'te verilen probleme göre belirtilen ifadelerin hem sinyal hem de fonlar için tanımlanması gerekir. Fon, sinyal ile aynı son durum parçacıklarını taşıyan fakat tamamen farklı bir başlangıç ve oluşum sürecine veya süreçlerine ait çalışmaları içerir. Herhangi bir sürecin eksiksiz olarak incelenebilmesi, sinyalin yanısıra bütün fonların ayrı ayrı incelenmesiyle mümkün olur. CompHEP'te proses ve alt prosesler seçilebilir. Seçilen her bir prosesin fiziksel anlamına ve çarpışma kinematığına göre çeşitli kriterler getirilebilir. Bu kriterler bazen CompHEP'te çalışılan prosesin diverjans durumunu (fotonlu süreçlerde bulunur ve sonsuza gitmeyi ifade eder) yok etmek için de yararlıdır. CompHEP'te olaylar çıktı halinde yazdırılabilir ve grafik çıktısı alınabilir. SUSY içeren modellerde CERNLIB'in programa dahil edilmesi gerekir.

CompHEP'te olayların sonuçlarını hesaplamak için bozunma ve saçılma durumundan uygun olan seçilir. Seçilen kanal için diyagramlar görülür ve bu diyagramlardan gerekli olanlar Latex çıktısı halinde alınabilir; gereksizler silinebilir. Seçilen diyagramların bozunma genişlikleri ve tesir kesitleri C++ programlama diliyle derlenerek sembolik olarak hesaplanır ve bu hesaplar yazdırılabilir. Sonra sayısal hesaplar için yeni bir pencere açılır; uygun alt proses ve gerekli kriterler seçilerek Vegas veya Simpson Algoritmaları yardımıyla olaylar integrallenir. Grafikselt çıktı elde etmek için istenilen değişken seçilerek sınırlar ayarlanır. Liste çıktı sonuçları için $X^2 < 1$ olan sonuçlarla hatalar elde edilmeye çalışılmalıdır. Çünkü bu sonuçlar beklenen değerler için hatası düşük olan sonuçlardır. CompHEP'te istenilen çoklukta olaylar üretilebilir.

Çalışılan proses, rezonansta üretim ve bozunum süreçleri ise Feynman diyagramları bölümünde sadece çalışılan süreç seçilip diğerleri silinir. Bu durum fonlar açısından çalışma kolaylığı sağlar.

Proton çarpıştırıcısı (LHC), 14 TeV'lik kütle merkezi enerjisinde $L_{\text{int}} = 100 \text{ fb}^{-1}$ ışıkta üretilen son durum parçacıkları için istatistiksel kesinlik (statistical significance) değeri,

$$SS = \frac{\sigma_{s+B} - \sigma_B}{\sqrt{\sigma_B}} \sqrt{\varepsilon \cdot BR \cdot L_{int}} \quad (4.1)$$

denklemlerle hesaplanır. Buradaki ε ve BR değeri b kuarklı süreçlerde dahil edilir ve değerleri yaklaşık olarak BR için 0.2, ε için de 0.5'tir. ε detekte etme verimi, BR de sürecin dallanma oranını gösterir. Olay sayıları incelenen olay için elde edilen tesir kesiti ile ışınlık değerinin çarpılmasından elde edilir:

$$N_s = \sigma_s \cdot L_{int} \quad (4.2)$$

İstatistiksel kesinlik değeri, Yüksek Enerji fiziği süreçlerinde 5 sayısından büyük değerde elde edilebiliyorsa çalışılan kanalın oluşma şartlarının kesinlikle oluşabileceği anlamını taşımaktadır. Değer küçüldükçe çalışılan kanalın oluşma olasılığı azalmaktadır. LHC deki Dördüncü Aile Kuarklarının tekli üretimleri için, üç farklı parametrisasyon (PI, PII, PIII) üzerinden hesaplamalar yapılacaktır. Yapılacak olan hesaplamalar üç farklı parametrisasyonda 200 – 800 GeV kütle aralığında yapılacaktır.

CompHEP'te seçilen değerler tablodaki gibidir.

Çizelge 4.4 Tesir kesiti hesaplamaları tablosu

Table 1 değerleri	Table 2 değerleri
Kapa=1	Kapa=1
Lambda=1000	Lambda=10000
GG=1.2136	GG=1.2136
PDF=10	PDF=10
Mass=200-800	Mass=200-800
QCDScale=200-800	QCDScale=200-800
Itmax=20	Itmax=20

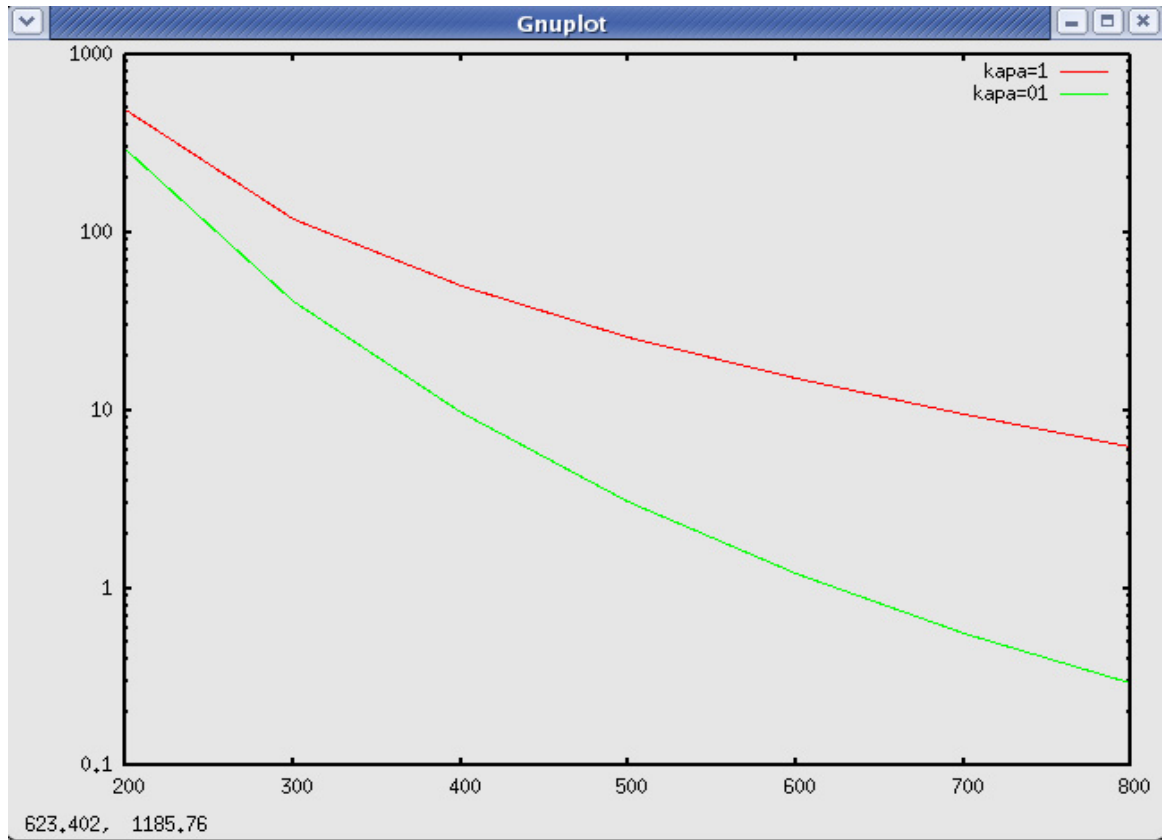
Burada Kapa ve Lambda deęerleri yeni fizik sakalarını ifade etmektedir. PDF ise Parton Distribution Function- Parton daęılım fonksiyonu alıřılan bir proses iin momentum transferine gre belli tanımlı bir boylamsal momentum deęerinde paracıęı bulma olasılık yoęunluęudur. Itmax, hesaplamaları tekrarlama oranıdır.

izelge 4.5 Rezonans Tesir kesiti (pb) ve t' kuarkın bozunma geniřlikleri $\kappa/\Lambda=1 \text{ TeV}^{-1}$

Mass (GeV)	200	300	400	500	600	700	800
$\sigma(\text{pbpb} \rightarrow \text{tpTp})$	484.41	119.87	49.656	25.957	15.232	9.5395	6.2249

izelge 4.6 Rezonans Tesir kesiti (pb) ve t' kuarkın bozunma geniřlikleri $\kappa/\Lambda=0.1 \text{ TeV}^{-1}$

Mass (GeV)	200	300	400	500	600	700	800
$\sigma(\text{pbpb} \rightarrow \text{tpTp})$	291.95	41.240	9.6354	3.0552	1.2017	0.5588	0.2953



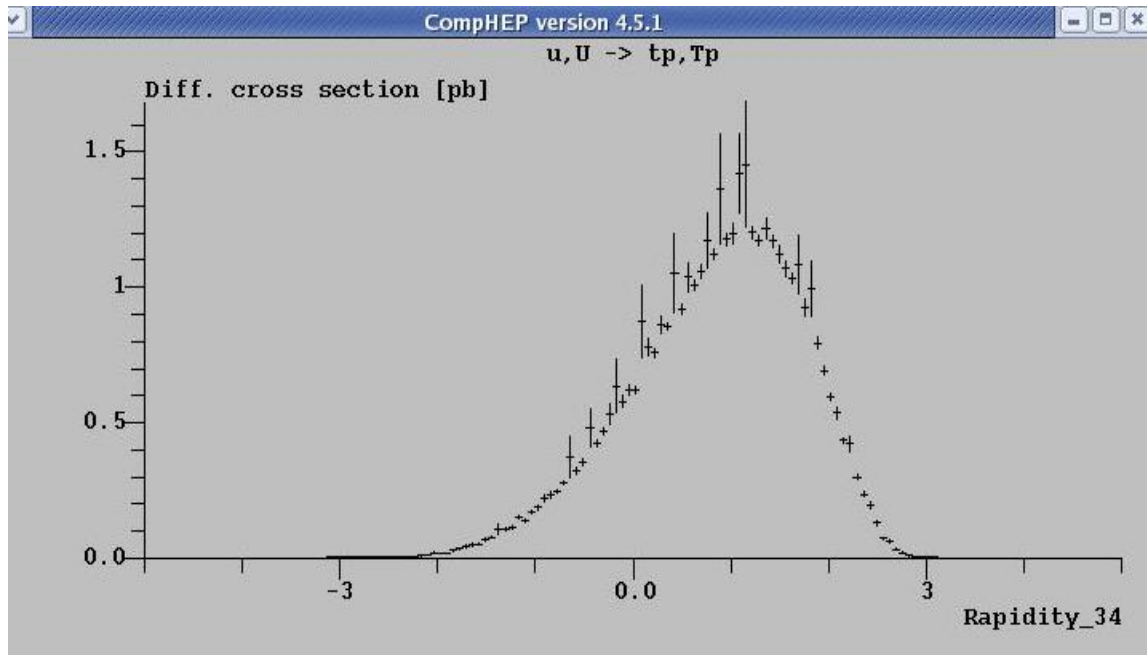
Őekil 4.14 Tesir kesitinin ktleye baęlı grafięi

Aşağıdaki iki süreç için fon (background) değeri "SM Unitary Gauge" de hesaplanmıştır.

$$Pp \rightarrow W^+ W^- Z \quad \text{kanalı için} \quad | \quad \sigma_b = 1.0717e - 01$$

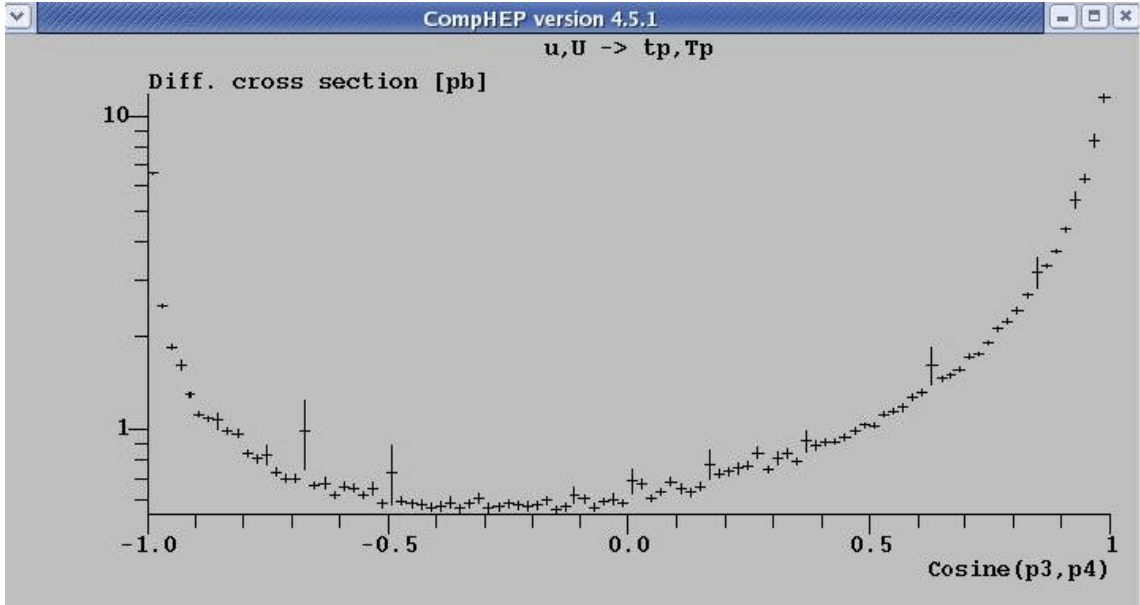
$$Pp \rightarrow W^+ W^- b b \quad \text{kanalı için} \quad | \quad \sigma_b = 5.3223e + 01$$

tesir kesitleri fonlar için bulunmuştur. Bu fonlardan W^+, W^-, b burada b jetleri fazla miktardadır. Bu değer kinematik dik momentum kriteri koyarak fonda yaklaşık %60 değer kaybına yol açmaktadır. Dik momentum kriteri, merkezsiz bölgede yumuşak jetleri gözönüne almamak için 50 GeV civarında konulmuştur. Bu durum sinyalin fonlardan daha yüksek tesir kesiti ile elde edilmesini sağlar.



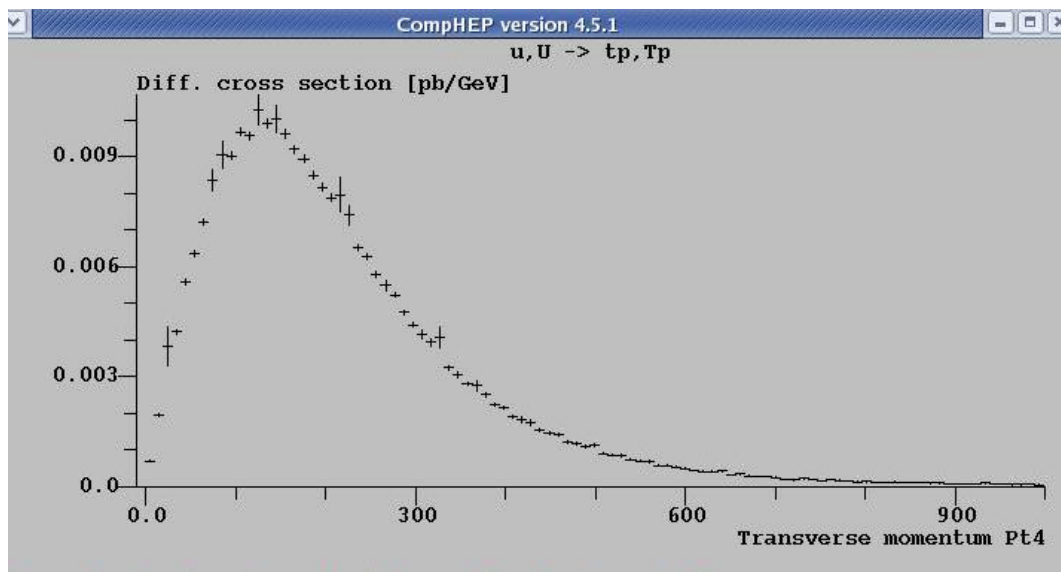
Şekil 4.15 Rapidity histogramı

$P, P \rightarrow tp, Tp$ kanalı için diferensiyel tesir kesitine göre rapidity değişimi. Histogramdan görüldüğü gibi Dördüncü Aile Kuarkları ATLAS dedektörünün merkezsiz bölgesinde daha çok yoğunlaşmışlardır - 0 noktası civarında.



Şekil 4.16 Cosine histogramı

$P,P \rightarrow tp, Tp$ kanalı diferensiyel tesir kesitine göre parçacıkların açisal değişimi. Yukarıdaki şekilden de açıkça görüldüğü gibi çarpışma farklı yönlerde asimetrik bir çarpışmadır. Açısal değişim -1 ve 1 aralığında 0 bölgesine daha yakındır. Sınırlarda 1'e yakındır çünkü gelen protonlar aynı yönde veya 180 derece açı yapacak şekilde daha fazla dağılım gösterirler.



Şekil 4.17 Transverse Momentum histogramı

Yukarıdaki $P,P \rightarrow t\bar{t}$, $T\bar{T}$ kanalındaki $T\bar{T}$ için diferensiyel tesir kesitine göre dik momentum histogramı. Dik momentum olduğu için enerji ve momentum çok yüksek değerlere çıkamamaktadır. Histogramdan anlaşılacağı gibi parçacıklara aktarılan enerji yaklaşık olarak 150 MeV ile 350 MeV aralığındadır.

Bu sonuçlara göre LHC’de $t\bar{t}$ ve $T\bar{T}$ çift üretimi bol miktarda beklenmektedir. Birçok fonlar üzerinde çalışılmış ve sinyal’e göre hepside azaltılabilmektedir. En inatçı olan fon tezde bahsedilen W^+,W^-,b fonudur, bu fon dik momentum kriteri sonucunda elenebilmiştir.

5. SONUÇLAR VE TARTIŞMA

Bu tezde yapılan dördüncü SM u_4 kuarkının aranma çalışmaları ile parçacık fiziği yüksek enerji modelleme programı CompHEP; dedektör modelleme programları-ATHENA, PGS ile bu programlarının analizlerini yapmamızı sağlayan ROOT programları sayesinde CERN-LHC fiziği hakkında geniş bir bilgi ağına sahip olunmuştur. CompHEP'le yapılan u_4 'ün üretilmesi ve bozunması çalışmaları sayesinde Standart Model ayarları, Lagrangian terimleri, matematiksel ve istatistiksel integral alma yöntemleri, histogramlama teknikleri hakkında bilgi sahibi olduktan sonra $P,P \rightarrow t\bar{p},T\bar{p}$ kanalının CompHEP'le modellenmesi yapılabilmektedir

Sonuç olarak Dördüncü Standard Model u_4 kuarkları tekli ve çiftli üretimleri proton-proton ya da diğer deyişle hadron-hadron çarpıştırıcılarında bol miktarda üretilebilecektir. Gelecekte inşa edilmesi düşünülen e-p veya da lineer çarpıştırıcılar olan e-,e+ çarpıştırıcılarında Dördüncü Standard Model Fermiyonlarının (leptonlar ve kuarklar) farklı çarpıştırıcılarda aranması ve bulunması çalışmaları Standard Model'in eksikliklerinden birini tamamlama anlamında önem kazanmaktadır. Çünkü Dördüncü SM ailesinin varlığı birçok kozmolojik ve astrofiziksel sonucu içerir.

KAYNAKLAR DİZİNİ

- [1] Zafaritos – Fizikte ve Mühendislikte Modern Fizik
- [2] <http://arxiv.org/abs/hep-ph/0004271> Erişim tarihi: 09.03.2007
- [3] <http://arxiv.org/abs/hep-ex/0007043> Erişim tarihi: 09.03.2007
- [4] http://atlas-service-eneews.web.cern.ch/atlas-service-eneews/2009/news_09/news_cernstate.php
- [5] Bilim ve teknik dergisi Nisan 2007 – TÜBİTAK
- [6] <http://www.w3.org/People/Berners-Lee/WorldWideWeb.html> Erişim tarihi: 27.03.2007
- [7] www.taek.gov.tr/uluslararası/cern/cern.htm
- [8] www.cern.ch/lhc
- [9] <http://www.taek.gov.tr/tr/sss/129-hizlandirici-fizigi/533-cern-lhc-atlas-cms-clic-alice-nedir.html>
- [10] <http://public.web.cern.ch/public/en/lhc/ATLAS-en.html>
- [11] <http://atlas.ch/> Erişim tarihi: 17.04.2007
- [12] THE TRACKER PROJECT Technical Design Report, 1998, CERN/LHCC 98-6.
- [13] library.cu.edu.tr/tezler/6409.pdf
- [14] Electromagnetic Calorimeter Technical Design Report, CERN-LHCC 97-033, December 1997.
- [15] Hadronic Calorimeter Technical Design Report, CERN-LHCC 97-031, June 1997.
- [16] Muon Technical Design Report, CERN-LHCC 97-031, 1997; Phd.Thesis G.Bruno, University of Padova, İtaly.
- [17] <http://www.tuba.gov.tr/>
- [18] <http://lhcb.web.cern.ch/lhcb/> Erişim tarihi: 17.05.2007
- [19] <http://thm.ankara.edu.tr/tac/YAZOKULU/yazokulu1/dersnotlari/ocakir1.pdf>
- [20] <http://www.physics.ucdavis.edu/~conway/research/software/pgs/pgs.html>
- [21] <http://www.physics.ucdavis.edu/~conway/research/software/pgs/pgs4-olympics.htm>
- [22] Hızlandırıcı ve Parçacık Fiziğinde Bilgisayar Uygulamaları Yaz Okulu (Adana)
- [23] http://wiki.grid.org.tr/index.php/ATHENA_Framework_Giriş_ve_örnek_Uygulamalar

KAYNAKLAR DİZİNİ (devamı)

[24] <https://twiki.cern.ch/twiki/bin/view/Atlas/RegularComputingTutorial>

[25] <http://arxiv.org/hep-ph/9908288> v2 Erişim tarihi: 11.02.2000

[26] [http:// theory.npi.msu.su/comphep](http://theory.npi.msu.su/comphep)

[27] Rene Brun/CERN, Fons Rademarkers,Suzanne Panacek/FNAL, Damir Buskulic/Universite de Savoie/LAPP, Jörn Adamczewski/GSI, Marc Hemberger/GSI, Nick West/Oxford (ROOT, Users Guide 3.02c (October, 2002)

[28] Hızlandırıcı ve Parçacık Fiziğinde Bilgisayar Uygulamaları Yaz Okulu (Adana)

[29] <http://wwwth.cern.ch/lhcOlympics/lhcolympicsII.html>

[30] <http://theory.sinp.msu.ru/comphep> Erişim Tarihi: 24.12.2006