

DÖRDÜNCÜ STANDART MODEL AİLESİ
FERMİYONLARININ ÜRETİMİ VE
MODELLEME ÇALIŞMALARI

Yüksek Lisans Tezi

Gamze KİBAR

Eylül - 2008

**DÖRDÜNCÜ STANDART MODEL AİLESİ FERMİYONLARININ ÜRETİMİ VE
MODELLEME ÇALIŞMALARI**

Gamze KİBAR

Dumlupınar Üniversitesi
Fen Bilimleri Enstitüsü
Lisansüstü Yönetmeliği Uyarınca
Fizik Anabilim Dalında
YÜKSEK LİSANS TEZİ
Olarak Hazırlanmıştır.

Danışman: Yrd. Doç. Dr Hatice DURAN YILDIZ

Eylül – 2008

KABUL ve ONAY SAYFASI

Gamze KİBAR'ın YÜKSEK LİSANS tezi olarak hazırladığı “Dördüncü Standart Model Fermiyonlarının Üretimi ve Modelleme Çalışmaları” başlıklı bu çalışma, jürimizce lisansüstü yönetmeliğin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

/ /2008

Danışman Üye : Yrd. Doç. Dr. Hatice Duran Yıldız

Üye : Prof. Dr. Atalay Küçükburşa

Üye : Doç. Dr. Eyüp Tel

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

Prof. Dr. M. Sabri ÖZYURT
Fen Bilimleri Enstitüsü Müdürü

DÖRDÜNCÜ STANDART MODEL AİLESİ FERMİYONLARININ ÜRETİMİ VE MODELLEME ÇALIŞMALARI

Gamze KİBAR

Fizik Anabilim Dalı, Yüksek Lisans Tezi, 2008
Tez Danışmanı: Yrd. Doç. Dr. Hatice DURAN YILDIZ

ÖZET

Bu tezde temel parçacıklar, temel parçacıklar arasındaki etkileşmeler ve Standart model hakkında detaylı bilgiler verilmiş ve temel parçacıklardan olan dördüncü Standart Model ailesi fermiyonlarının gözlenebilmesi üzerine modelleme çalışmaları sunulmuştur.

CERN-LHC Deneyi Temmuz 2008'de başlayacak olan dünyanın en büyük Yüksek-Enerji Fiziği deneyidir. ATLAS Deneyide LHC Hızlandırıcısının en büyük deneylerinden birisidir. ATHENA, tam ATLAS Dedektörü modelleme programıdır. LHC Deneyi düşük aydınlıkta çalışmaya başladıktan sonra ortaya çıkacak olan çok fazla miktarda parçacık için deney yapılmadan; sanki üretilmiş gibi ve dedektör etkilerini de içerecek şekilde üzerine işlem yapılarak; parçacıkların üretim ve bozunumlarını incelemek ve böylece, parçacık özelliklerinin dedekte edilmesine ait tüm karmaşıklığı çözmemizi yazılan modelleme programı sağlayacaktır. Bu çalışmada ATHENA ve PYTHON dilleri kullanılarak $\sqrt{s}=14$ TeV'lik kütle merkezi enerjisinde ve $L = 10^{33} \text{cm}^{-2}\text{s}^{-1}$ ışınlıkta ATLAS dedektörünün tam modellemesi Standard Model fermiyon ailesi için yapılmıştır. Sonuçlar ROOT programı ile analiz edilmiştir.

Anahtar kelimeler: ATHENA, ATLAS Deneyi, Dördüncü Standart Model Ailesi, LHC, PYTHIA, ROOT,

THE PRODUCTION AND MODELING OF FOURTH STANDARD MODEL FAMILY FERMIONS

Gamze KİBAR

Department of Physics, Masters Thesis, 2008

Thesis Advisor: Assist. Prof. Dr. Hatice DURAN YILDIZ

SUMMARY

In this thesis, detailed information about the interaction of base particles and Standard model is given and works has been done for fourth Standard Model family fermions about how to observe the fundamental particles.

CERN-LHC is the world's biggest High-Energy experiment that will start July 2008. ATLAS Experiment is one of the biggest experiments of LHC. ATHENA is full ATLAS detector modeling program. LHC experiment will cause the examining of the production and corruption of the particles to be processed and by this way to solve all the complexity that belong to particle features For the vast amount of particles that will come up after starting under soft light as if they are produced and they include the effects of detector without doing an experiment. In this work full modelling of ATLAS detector is done for Standard Model fermions family by using ATHENA and PYTHON languages at $\sqrt{s}=14$ TeV center of mass energy and $L = 10^{33} \text{ cm}^{-2}\text{s}^{-1}$ luminosity. The results are analyzed with the ROOT program.

Key words: ATHENA, ATLAS Experiment, Fourth Model Family, LHC, PHYTHIA, ROOT

TEŐEKKÜR

Bu alıŐma sűresinde bana her tűrlű desteęi, emeęi, sabrı gűsteren, yardımlarını benden esirgemeyen, alıŐmayı sevdiren, bilgi ve tecrűbelerini benimle paylaŐan ok deęerli danıŐman hocam Sayın Yrd. Do. Dr. Hatice DURAN YILDIZ' a teŐekkűrű bir bor bilirim.

Tez alıŐmalarım boyunca CERN ATLAS Deneyinde Dűrdűncű Aile Kuarklarının Siműlasyon alıŐmaları ve Veri Analizi isimli, 105T442 numaralı proje tarafından burs saęlayan TBAG (Temel Bilimler AraŐtırma Grubu) – TŪBİTAK' a ve alıŐmalarımda destek aldıęım TAEK (Tűrkiye Atom Enerjisi Kurumu)'e teŐekkűr ederim.

Ayrıca maddi ve manevi destekleri iin her zaman yanımda olan aileme űzellikle annem Gűlay KİBAR'a, her tűrlű sıkıntıda yanımda olan arkadaŐım Erman ETİN ve tűm arkadaŐlarıma teŐekkűrler.

Gamze KİBAR

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	iv
SUMMARY	v
ŞEKİLLER DİZİNİ.....	viii
ÇİZELGELER DİZİNİ	ix
SİMGELER VE KISALTMALAR	x
1. TEMEL PARÇACIKLAR VE STANDART MODEL.....	1
1.1. Temel Parçacıklar Arasındaki Etkileşmeler.....	2
1.2. Temel Parçacıkların Orijini.....	6
1.3. Standart Model ve Ötesi	6
2. TEMEL PARÇACIKLAR NASIL GÖZLEMLENİR?	9
2.1. Hızlandırıcılar ve LHC	9
2.2. CERN’deki Büyük Deney ve Dedektörler.....	11
2.3. ATLAS Dedektörü	13
2.3.1. İç Dedektör.....	15
2.3.2. Kalorimetre.....	16
2.3.3. Müon Sistemi.....	16
2.3.4. Magnet Sistemi.....	16
3. DÖRDÜNCÜ STANDART MODEL FERMİYON AİLESİ	18
3.1. Fermiyon Ailesinin Sayısı ve Teorik Yaklaşım	18
3.2. Dördüncü Ailenin Gerekliliği	19
4. ATHENA DEDEKTÖR MODELLEME PROGRAMI.....	24
4.1. SSH (Secure SHell Linux Uzaktan Erişim Programı)’nın Yüklenmesi.....	24
4.2. ROOT Grafik Analiz Programı.....	25
4.3. Dördüncü Aile Fermiyonlarının ATHENA ve PYTHON Yazılım dillerini kullanarak elde edilmesi.....	26
4.4. PYTHIA Programı.....	29
4.5. gamze_gen.out Özet Dosya içeriği	37
4.6. Liste Yorumu.....	43

İÇİNDEKİLER(Devamı)

	<u>Sayfa</u>
4.7. Programa Etki Eden Kinematik ve Geometrik Kriterler	44
5.BİTWİSE OPERATÖRLER.....	47
5.1. Bitsel Değil Operatörü.....	47
5.2. Bitsel Kaydırma Operatörleri.....	47
5.3. Bitsel Ve Operatörü.....	48
5.4. Bitsel Özel Veya Operatörü.....	48
5.5. Bitsel Veya Operatörü.....	49
5.6. Bitsel Operatörlere İlişkin İşlemler Atama Operatörleri.....	50
5.7. Bir Sayının Belirli Bir Bitinin 1 Yapılması.....	50
5.8. Bir Sayının Belirli Bir Bitinin 0 Yapılması.....	50
5.9. Bir Sayının Belirli Bir Bitinin Değerinin Test Edilmesi.....	51
5.10. Bir Sayının Belirli Bir Bitini Ters Çevirmek.....	51
5.11. Cygwin Yapısı.....	56
6. SONUÇ ve DEĞERLENDİRME	60
KAYNAKLAR.....	61

ŞEKİLLER DİZİNİ

<u>Sekil</u>	<u>Sayfa</u>
1.1. Proton Yapısı	3
1.2. Nötron Yapısı	3
1.3. Pion Yapısı	3
1.4. Mezon Yapısı.....	4
1.5. Baryon Yapısı.....	5
2.1. LHC Çemberi.....	11
2.2. Alice Dedektörü	12
2.3. CMS Dedektörü	12
2.4. LHCb Dedektörü	13
2.5. ATLAS Dedektörü.....	13
2.6. ATLAS Dedektör Büyüklüğünü CERN' deki bir binayla karşılaştırma.....	14
2.7. ATLAS Dedektörü İç Yapısı.....	14
2.8. ATLAS Koordinat Sistemi	17
4.1. ROOT Çalışma Menusu.....	26
4.2. Enerji Dağılım Histogram Grafiği	35
4.3. Parçacık x eksenli Momentum Grafiği.....	36
4.4. Dedektör İçindeki Eta Sınırları.....	44
4.5. Üretilen Parçacıkların Dedektöre Göre x yönündeki momentumları.....	46
4.6. Monte Carlo ile Üretilen Parçacıkların Etkileşme Vertekslerinde Davranışları.....	46
4.7. Üretilen Parçacıkların Dedektöre Göre z Yönündeki Momentum Dağılımı.....	51
5.1. Yazılan Komutlarla ve Cygwin Yapısı kullanarak elde edilen grafik.....	60

ÇİZELGELER DİZİNİ

<u>Çizelge</u>	<u>Sayfa</u>
1.1. Kuarklar ve Leptonlar	2
3.1. Leptonlar ve Kuarklar	18
3.2. LHC'de Dördüncü Aile Üretimi.....	19
4.1. Diferansiyel Tesir Kesintinin Maksimum Araştırılmasının Sonuçları	33
4.2. Parçacık/Parton Veri Çizelgesi.....	34
4.3. Tesir Kesiti ve Olay Sayılarının İstatistiği	34

SİMGELER VE KISALTMALAR

SİMGELER

L	Lagrangian, ışıklık
J^μ	Elektromanyetik Akım Operatörü
α_{em}	İnce Yapı Sabiti
Θ_w	SM' in Weinberg açısı
η	Skaler Higgs alanının boşluk beklenen değeri
M^0	Kütle Matrisi
p_T	Dik Momentum
E_T	dik enerji
a_F	fermion Higgs etkileşme sabiti

KISALTMALAR

AÇIKLAMA

SM	Standart Model
SUSY	Süper simetri
DMM	Demokratik Kütle Matrisi
LHC	Büyük Hadron Çarpıştırıcısı
ATLAS	Toroidal Büyük Hadron Çarpıştırıcısı Aleti
CERN	Avrupa Nükleer Araştırma Organizasyonu
QED	Kuantum Elektro Dinamik
GUT	Birleştirme Teorisi
ALICE	A Large Ion Collider Experiment
CMS	Compact Muon Selenoid
LAr	Likit Argon Kalorimetre

1. TEMEL PARÇACIKLAR VE STANDART MODEL

Bilim adamlarının yıllar boyunca merak ettikleri en önemli konulardan biriside maddeyi oluşturan en temel parçacığın ne olduğu sorusudur. Günümüzde bu sorunun cevabı hala aranmakta olup, fizik alanında önemli bir yer kaplamaktadır. Çok uzun bir dönem maddeyi oluşturan en temel parçacık atom zannedilmiştir ve bölünemez olarak düşünülmüştür. Atomlar elektron kabuklarını oluşturan iki kısımdan oluşur. Bu iki kısım elektron ve çekirdektir. Çekirdek proton(p) ve nötron(n)'dan oluşur. Proton ve nötron iki tür kuarktan oluşur, bunlar alt ve üst kuarktır. Ayrıca, alt kuark “u” ile üst kuark ise “d” ile gösterilir.

İlk temel parçacık olan elektronu Thomson 1897 yılında keşfetmiştir. Onun ardından Rutherford 1910 yılında atom çekirdeğini bulmuştur. Yine Rutherford 1918 yılında çekirdeğin pozitif yüklü protonlardan oluştuğunu göstermiştir. Araştırmaların bu seviyesinde değişik elementlerin ortak yapı taşları “elektron, proton ve nötron” olarak üç taneydi.

Fakat yeni buluşlar yapıldıkça, bu seviyenin daha da karmaşık olabileceğine dair kanıtlar birikmekteydi. Elektron, proton ve nötrondan hariç farklı parçacıkların olabileceği ortaya çıktı. Anderson 1932 de elektronun anti parçacığı olan pozitronu keşfetti. Bu parçacığın elektronla aynı kütle ve spine sahip, ancak yükünün pozitif olduğu görüldü. Maddenin yapısında pozitron yoktur fakat reaksiyonlarda ortaya çıktığı için pozitron da temel parçacık olarak kabul edilmelidir.

Çekirdeği oluşturan parçacıklardan nötronun kütlesi protonun kütesinden daha büyük olduğu için ve bu kütle farkının durgun bir elektron oluşmasına yeterli olduğu için beta bozunması ile nötrino dediğimiz diğer bir temel parçacık oluşur. Bu parçacığın var olabileceğini Pauli 1930 yıllarında ortaya atmıştır. Nötrinoda pozitron gibi maddenin yapısında yer almaz fakat uzayda yıldızlardaki çekirdek reaksiyonlarında nötrinolar bol miktarlarda oluşur. Bir diğer temel parçacık 1930 yıllarında bulunan fotonudur. Elektromanyetik etkileşmeyi taşıyan taşıyıcı fotonlardır, fotonlarda maddenin yapısında bulunmazlar.

1930 yıllarında maddenin üç tür temel parçacıktan oluştuğu şeklindeki düşünceden bir süre sonra vazgeçilmiştir. Çünkü 1960 yıllarının ortalarında bu karmaşıklığa bir çözüm getirilebileceği anlaşıldı. Aslında temel olarak düşünülen parçacıkların bir bütün olarak düşünülmesi gerektiği öngörüldü.

Fizikte temel parçacıklarla ilgilenen alan “yüksek-enerji fiziği” dir. Yüksek-enerji fiziği denilmesi adından da anlaşılacağı üzere diğer fizik alanlarından çok daha yüksek enerjilerle ilgilenmesidir.

$$P=h/\lambda \quad (1.1)$$

Formülünden anlaşılacağı üzere parçacıkları ve kuvvetleri araştırmak için kısa mesafelere uzanmak yüksek enerji ve momentum gerektirir.

1.1 Temel Parçacıklar Arasındaki Etkileşmeler

Temel parçacıklar fermiyonlar başlığı altında iki gruba ayrılır. Bunlar kuarklar ve leptonlardır. Günümüzde ispatlanmış bilinen lepton sayısı altı tanedir. Bunlar elektron, müon, tau ve bunların nötrinolarıdır. Ayrıca bunların anti parçacıkları da vardır. Müonun yükü elektronun yüküne eşit, kütlesi 105,7 mev/c dir. Taunun kütlesi ise 1784 mev/c dir. Bu parçacıkların nötrinoları yüksüzdür. Tau ve müonun da anti parçacıkları vardır.

Saydığımız altı tane leptondan sadece elektron normal madde yapısında yer alır. Çünkü elektron en küçük kütleli parçacıktır. Bozunarak dönüşebileceği başka madde olmadığı içinde kararlıdır ve temel parçacıktır. Bunun yanı sıra μ (müon), τ (tau) ve nötrinolar kararsız oldukları için normal bir maddenin parçası olarak düşünülemezler.

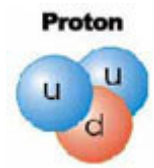
Çizelge 1.1: Kuarklar ve Leptonlar

Nesiller	1	2	3
Leptonlar	e ν_e	μ ν_μ	τ ν_τ
Kuarklar	u D	c s	t b

Altı kuark anti kuarkları ile birlikte üç guruba ayrılır. Kuarklar tek başlarına asla gözlemlenemez. Kuarklar "hadron" olarak bilinen parçacıklar içerisinde hapis olmuştur diyebiliriz. Protondaki ve elektrondaki gibi kuarklar elektrik yüküne sahiptir. Bununla beraber elektrik yükleri kesirlidir (2/3 veya -1/3, -2/3 ve 1/3 anti kuarklar için). Ve parçacıkları oluştururlarken yük daima tamsayıdır.

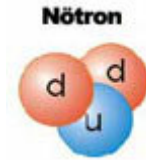
Kuarklar parçacıkların yükü tamsayı olacak biçimde bir araya geldiğinden kuarkların her türlü kombinasyonu mümkün değildir. Hadronların iki şekli vardır, bunlar baryonlar ve mezonlardır. Üç kuarkın bir araya gelmesi ile baryonlar, bir kuark ve bir anti kuarkın bir araya gelmesi ile mezonlar oluşur. Baryonlara iki örnek proton ve nötrondur.

Proton iki up kuark ve bir down kuarkın bir araya gelmesi ile oluşur. Şekilden görüldüğü gibi her bir kuarkın yükü toplanıp proton için yük +1 elde edilir.

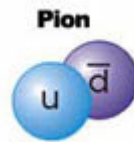


Şekil 1.1: Protonun Yapısı

Nötron iki down kuark ve bir up kuarktan meydana gelir. Kuarkların yüklerini tekrar toplarsak 0 olan nötronun yüküne ulaşırız.



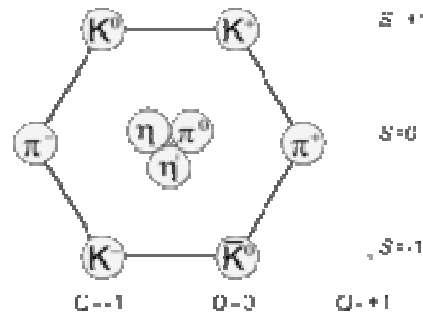
Şekil 1.2: Nötronun Yapısı



Şekil 1.3: Pionun Yapısı

Mezon için bir örnek piondur. Pion bir up bir de down kuarkın bir araya gelmesi ile oluşur. Mezonlar parçacık ve anti parçacık kombinasyonu olduğundan kararsız bir yapı gösterirler ve çok hızlı bozulurlar.

Bir mezon kuark anti-kuark ve bağlayıcı gluonların birleşmesinden oluşur. Örneğin, bir üst kuark ve bir anti-alt kuark dan oluşmuş (+1) yüklü piondur. (π^+) Mezonun anti-parçacığı kuark ve anti-kuark değişimi ile oluşur. Bir mezon renksiz bir nesnedir. Bunun sebebi kuark ve anti-kuarkları zıt renk yüklerine sahiptir. Ayrıca mezonlar tam sayı spinli parçacıklardır. ($S=0,1,2, \dots$) mezonların sayıları oldukça fazladır. Başlıca mezonlar pionlar (Π), eta parçacığı (η), rho parçacığı (ρ), omega parçacığı (ω), kappa parçacığı (K) dır. Her bir mezon parçacığı pozitif, negatif, nötr olmak üzere üç tanedir. Mezonlar çok kararsızdır. Yukawanın teorisine göre Π mezonları atom çekirdeği içerisinde protonlar ve nötronlar tarafından sürekli alınıp verilir bu sayede nükleonlar birada dururlar.

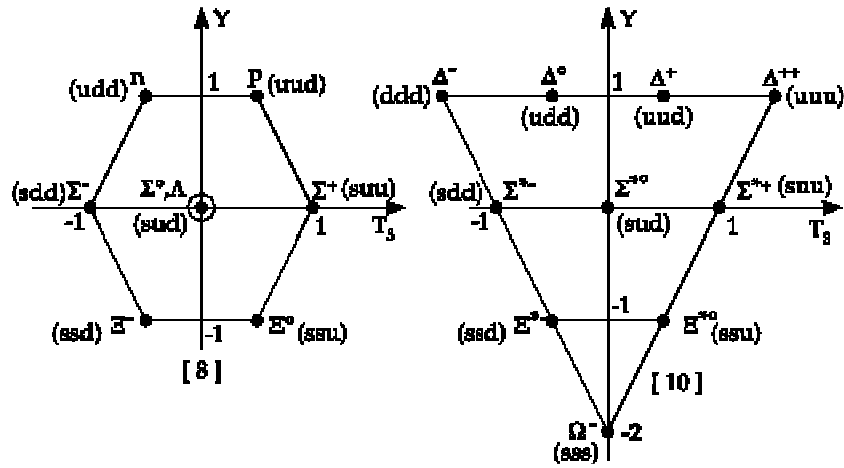


Şekil 1.4. Mezon yapısı

Baryonlar üç kuarkın bir araya gelmesinden oluşmaktadır. Kuarklar üçlüler halinde bir araya gelerek, daha pek çok parçacık oluştururlar. Baryonlar ailesinin bilinen, yaklaşık 120 çeşit üyesi vardır. Bazılarında üç kuarkın spini de aynı yöndedir. Bu durum, toplam spini ($3/2$) olan, daha ağır veya yüksek enerjili baryonların oluşmasına imkân sağlar. Ayrıca baryonları ayrı ayrı gözlemleyebiliriz.

Dolayısıyla, parçacıkların hepsinde; elektrik yükü (e^-)'nin tamsayı katları halinde iken renk yükü nötr, spin ise ($1/2$) veya ($3/2$) dir. En hafif baryon protondur (p). Protonun kütlesi ($938,3$) MeV/c^2 , yükü ($+e$) ve spini ($1/2$) dir. Proton, boş uzayda kararlı olan tek baryondur.

Diğer baryonların hepsi protondan daha fazla kütleye sahiptirler. Bir diğer baryon ise nötrondur (n). Nötronun kütlesi $(939,6) \text{ MeV}/c^2$, spini $(1/2)$ olan, yüksüz bir baryondur. Çekirdek içinde kararlı olan nötron boş uzayda kararsızdır. Boş uzayda β bozunmasıyla bir proton, bir elektron ve bir anti-nötrinoya dönüşür. Ortalama ömrü $(889) \text{ sn}$ dir. Daha ağır olan lamda (Λ), sigma ($\Sigma^+, \Sigma^0, \Sigma^-$), ksi (Ξ^0, Ξ^-) ve omega (Ω^-) baryonlarının hepsi $(10^{-9}) \text{ sn}$ den küçük ortalama ömürlerle değişik biçimde bozunarak, sonunda hep bir proton veya nötron haline gelirler.



Şekil 1.5. Baryon yapısı

Bilinen hadronların çoğu proton ve nötronun uyarılmış durumlarıdır. Bu yaklaşımla proton ve nötronun daha temel parçacıklardan oluştuğu söylenilebilir. Bu düşünce 1960 yıllarında Gell-Mann ve Zweigin birbirlerinden bağımsız olarak kuark modelini ortaya atmalarına neden olmuştur. Bu modele göre tüm hadronlar kuark adı verilen parçacıkların bağlı durumları olarak ortaya çıkar. Kuarklar için bugüne kadar yapılan araştırmaların sonuçlarına göre otuz altı tane kuark mevcuttur. Ama bunlar deneysel olarak gözlenmemiştir. Bugün bilinen kuark sayısı altı kuark, altı anti kuark olmak üzere on iki tane ve bunların üçer renk taşıma durumuna göre toplam kuark+anti kuark sayısı otuz altı tane olmaktadır.

Dört tane temel kuvvet söz konusudur. Bunlar çekirdek kuvveti, elektromanyetik kuvvet, zayıf kuvvetler ve kütle çekim kuvvetleridir.

- i. Çekirdek Kuvveti: En şiddetli kuvvettir. Baryon ve mezonlara etkilidir. Kuarkların proton ya da nötron içinde birada tutulmasını sağlayan, proton ve nötronların atom çekirdeği oluşumunu sağlayan kuvvettir.
- ii. Elektromanyetik Kuvvet: Çekirdek kuvvetinin yanında çok küçüktür. Elektrik ve manyetik alanlar sayesinde olur. Sonsuz menzili vardır atomda elektronların çekirdeğe bağlanmasını sağlar.
- iii. Zayıf Kuvvetler: Parçacıkların bozunumlarında görevlidir. Çekirdek kuvvetlerinin yanında elektromanyetik kuvvetten bile daha zayıftır.
- iv. Kütle-Çekim Kuvvetleri: Bütün parçacıklar arasında etkilidir. Güneş sistemini bir arada tutan kuvvettir, menzili sonsuzdur, şiddeti diğerlerine göre çok zayıftır bu nedenle ihmal edilebilir.

1.2 Temel Parçacıkların Orijini

Maddeye kütle kazandıran nesne Higgs olarak bilinen Higgs alanıdır. Bu alanda Higgs'in boşluktaki beklenen değeri anlık simetri kırınımından sonra W^+ , W^- , Z^0 bozonlarına kütle kazandırmaktadır. Teoride Higgs mekanizması ayar değişmezliği ve anlık simetri kırınımına dayanır. Higgs bozonu, Goldstone bozonunu yok ederek extra polarizasyon durumuyla ve serbestlik derecesiyle ortaya çıkar.

1.3 Standart Model ve Ötesi

Parçacıkların içyapılarını arayan bilim adamları yaptıkları araştırmalarla çeşitli modeller önermişlerdir. Önerilen modeller arasında evrendeki maddeleri ve kuvvetleri tanımlamak için bir model tanımlamışlardır. Bu model standart model (SM) dir [1]. Standart modele göre, yüzlerce parçacık ve karmaşık etkileşmeler birkaç temel parçacık ve etkileşme ile açıklanabilir, bizlere maddelerin davranışlarını açıklar. Bu modele göre iki çeşit parçacık vardır. Bunlar madde parçacıkları ve kuvvet taşıyıcı parçacıklardır. Her temel kuvvet bir taşıyıcı parçacıkla taşınmaktadır. Bu model, kuarklar ve leptonlar ile kuvvet taşıyıcıları yardımıyla madde ve maddenin yapısı hakkında merak edilen pek çok soruyu cevaplamaktadır. Bunun yanı sıra, fermiyonik ailelerin tekrarlanması, fermiyon kütleleri, Higgs mekanizması ve karışım açıları gibi bazı önemli sorular bu model ile açıklanamamaktadır. Bu gibi sorular fizikçileri yeni ve daha yüksek enerjili hızlandırıcılar yapmaya zorlamış, daha yüksek enerjili çarpışmalarla bu soruların cevaplanabileceği düşünülmüştür. Böylece Standart Model'in ötesinde ne gibi bir yeni fiziğin olduğu araştırılmaktadır. Standart Model ötesinde öngörülen bir model, Standart Model'

de bulunmayan yeni parçacıkları ve bunlar arasındaki yeni etkileşmeleri içerir. SM ötesinde öngörülen modeller; Büyük Birleştirme Teorisi (GUT), Süper Simetri (SUSY), Sicim Teorisi (String Theory) ve Kompozitliklerdir.

Büyük Birleştirme Teorisi yaklaşık 10^{15} GeV mertebesindeki bir enerji ölçeğinde güçlü ve elektro-zayıf kuvvetleri birleştirebilmeyi öngörmektedir.

Süper simetri, bugün gözlenen parçacıkların süper eşlerinin olduğunu ve bunların spinlerinin parçacıkların spinlerinden 1/2 kadar farklı olduğunu söyler [1].

Sicim (String) Teorisi, evrenin 10 boyutta titreşen çok küçük sicimlerden meydana geldiğini söyler. Sicim Teorisi kütle çekim kuvvetini diğer kuvvetlerle birleştirir.

Kompozit modellerde ise bugün temel olarak bildiğimiz kuark ve leptonların, daha temel olan parçacıkların bağlı durumu olduğunu öngörür.

Standart Model'de lepton ve kuarklar çok küçük mesafelere kadar (~ 10 cm) içyapısı olmayan parçacıklardır. Lepton ve kuarkların 3 aile olarak tekrarlanması bu parçacıkların daha temel alt bileşenlerden oluştuğu fikrini ortaya koymaktadır. Kompozitlik modeline göre, bugün temel olarak bildiğimiz parçacıkların da bir içyapısı olduğu düşünülür. Buna göre kuark ve leptonlar, preon denilen daha temel alt bileşenlerin bağlı durumlarıdır. Eğer kuarklar ve leptonlar daha temel bileşenlerden meydana gelmişlerse, bu bileşenlerin bağlanma enerjileri ölçeğinde, kuarklar ve leptonlar arasında yeni etkileşmeler açığa çıkarır. Λ Kompozitlik ölçeğinin çok altındaki enerjilerde bu etkileşmeler $1/\Lambda$ nın kuvvetleri ile bastırılmıştır. Bu yaklaşım uyarılmış durumların da olabileceğini öngörmektedir. Kuark ve lepton kompozitliği için en ikna edici kanıt kuark ve lepton taban durumlarının üzerinde yer alan uyarılmış durumların keşfi olacaktır. Kompozit modellerde spin $-1/2$ fermiyonlar en düşük radyal ve yörüngesel uyarımlardır. En düşük fermiyonik uyarımları spininin araştırılması preon yapısı hakkında bilgi verecektir. Kompozit modellerde öngörülen yeni parçacıklar olarak uyarılmış fermiyonlardan bahsedebiliriz. Uyarılmış fermiyonların kütlelerinin (m^*); verilen deneysel sınırlara göre statik lepton özelliklerinden Λ kompozitlik ölçeğinin (~ 1 TeV) altında olamayacağı anlaşılmıştır.

Sonuç olarak standart model dediğimizde

12 lepton(6lepton+6antilepton)

36 kuark(6 kuark+6 anti kuark) x 3

8 gluon

W^+ , W^- , Z^0 , gama ve

SM Higgs bozon

TOPLAM= 61 tane temel parçacık olduğu bilinmelidir.

SM'in bazı eksiklerine değinmemiz gerekirse, fakat bundan önce Çizelgede verilen Higgs bozonunu açıklamak gerekir. SM'deki boşluğu doldurmak için Higgs bozonu olarak adlandırılan kuramsal bir parçacık önerilmiştir. Bu parçacık elektrozayıf simetriyi bozan bir mekanizma sağlamaktadır. Buna göre Higgs parçacığı "Higgs alanının" kuantumlarıdır. Higgs parçacığı, alan sıfırdan farklı bir değerde iken en düşük enerjili durumda olur. Boş uzay olarak algıladığımız, bu alanın en düşük enerji düzeyinde olduğu durumdur. Alan kavramından parçacık kavramına geçecek olursak "boş uzay" tamamıyla Higgs parçacıklarıyla doludur. Higgs parçacığının spini sıfırdır. Tüm parçacıklar (fermionlar, ayar bozonları ve Higgs'in kendisi) kütlelerini Higgs'le olan etkileşmeleri sonucunda alırlar. Bugüne kadar Higgs parçacığı gözlenmemiştir [2].

SM'in cevaplayamadığı bir takım sorunlarda mevcuttur. Örneğin, SM bir parçacığın kütlelerini tahmin edemiyor, bileşik bir model değildir (Kuantum Renk Dinamiği–QED ile elektrozayıf teori birleştirilmemiştir), fermiyon aileleri tekrarlanmakta ve sayıları kesin olarak bilinmemektedir. Ayrıca SM kuarklar ve leptonlar temel parçacık mıdır yoksa bu parçacıklardan da alt parçacıklar var mıdır, madde–anti madde çelişmesini ve kara madde nedir sorularını cevaplayamıyor. Yukarıda merak edilen sorular SM'in problemleri olmakla birlikte, daha doğru modelin ne olacağının gelecekte kurulacak yüksek enerjili çarpıştırıcılarla ve hassas ölçümlerle belirlenebileceği düşünülmektedir. Şimdiye kadar üç kuark ve üç lepton çiftinin olduğunu gördük. Bu çiftlerden her biri bir aile oluşturur. Doğada yalnızca birinci aileden parçacıkları elde ediyoruz. Bu kısımda akla diğer soru gelmektedir: diğer iki aileye neden ihtiyaç duyulmaktadır? SM bu soruya da cevap verememektedir. SM'in cevaplayamadığı sorular için farklı modellerin ileri sürüldüğünü ve bu modellere yukarıda kısaca değinmiştik.

2.TEMEL PARÇACIKLAR NASIL GÖZLENİR?

Yüksek enerji fiziği deneylerinde, parçacıkların çok küçük kütleli oluşları ve enerjilerinin fazla miktarda olması nedeniyle çok uzun mesafelere gidebilen parçacıklar gözlenmeye çalışılmaktadır. Bu gözlemleri yapabilmek içinse hızlandırıcılara gerek duyulmaktadır.

2.1 Hızlandırıcılar ve LHC

Parçacık fiziğinde deney çalışmaları yapabilmek ve bu deneyleri değerlendirebilmek için parçacıklara büyük enerjiler vermek gerekir. Parçacıkları hızlandırmak demek, onların kinetik enerjilerini artırmak demektir. Farklı parçacık hızlandırıcıları olsa da temelde hepsi aynı şekilde çalışır. Eğer yüklü bir parçacık bir elektrik alana girerse bir kuvvet etkisinde kalır. Pozitif yüklü parçacıklar alan yönünde hızlanırken negatif yüklü parçacıklar alana ters yönde hızlanır. Hızlandırıcıların çalışma prensibinin en özet hali budur. Parçacık hızlandırıcıları iki çeşittir.

Lineer Hızlandırıcılar: Proton ve ağır iyon linakları

Elektron lineer hızlandırıcıları

Dairesel Hızlandırıcılar:

Betatron: Sabit yarıçap üzerinde indüksiyon yoluyla uyarılan elektrik alanın yine sabit yarıçaplı yörüngelerde parçacıkları hızlandırma ilkesine dayanarak çalışır.

Mikrotron: Aynı RF kaviteden geçirilerek her defasında daha büyük yarıçaplar taranarak hızlandırma ilkesine göre çalışır.

Siklotron: D şeklindeki karşılıklı iki magnet arasındaki boşlukta uygulanan RF hızlandırma geriliminden yararlanılarak parçacıkları spiral yörüngede hızlandırarak çalışır.

Sinkrotron: Dairesel yörünge boyunca her defasında RF kavitenin frekansının eş zamanlılık koşulu ayarlanacak şekilde artırılması ilkesine dayanır. Sabit yarıçaplı yörüngelerde dipol magnetler aracılığı ile tutulan ve yine magnetler aracılığı ile odaklamanın yapıldığı hızlandırıcı türüdür.

Hızlandırıcılar için önemli tutulan iki parametre aşağıda verilmektedir:

L: Luminositi ve Ecm: Kütle merkezi enerjisi

$$R = L \sigma \quad (2.1)$$

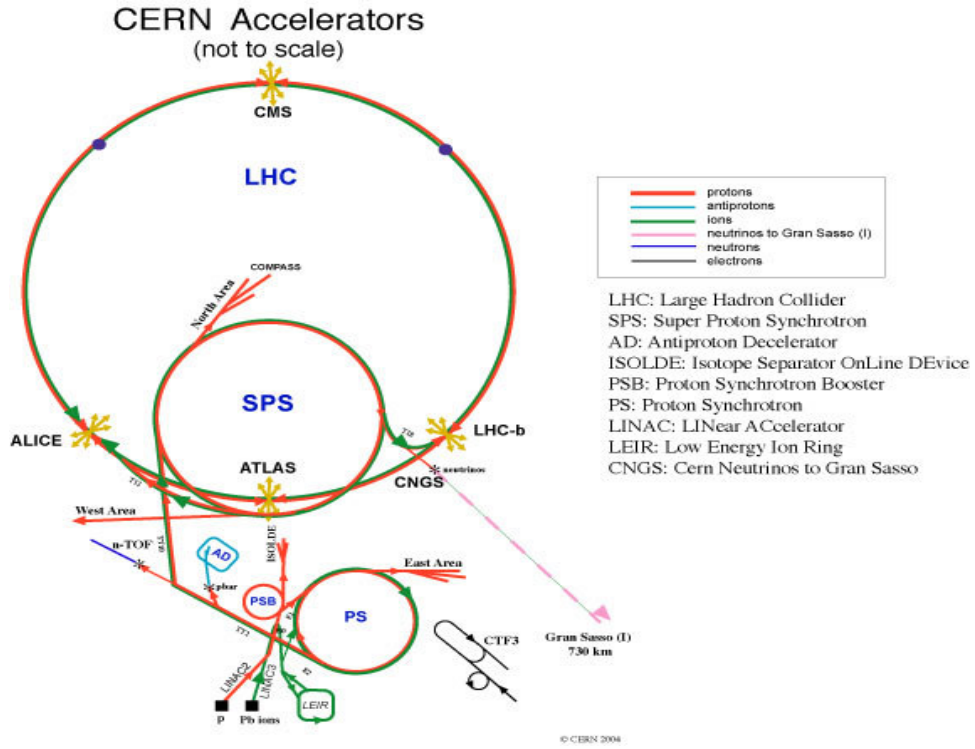
Çarpışan demetler: Hadron Çarpıştırıcıları, Lepton Çarpıştırıcıları, Lepton-Hadron Çarpıştırıcıları şeklinde gruplanır.

Alt-süreç düzeyinde TeV skalasına ulaşmanın dört yolu vardır:

- Halka tipli hadron çarpıştırıcıları
- Lineer elektron-pozitron çarpıştırıcıları
- Halka tipli müon çarpıştırıcıları
- Linak-halka tipli lepton-hadron çarpıştırıcıları.

2008 yılında çalıştırılması planlanan LHC' de proton-proton ve çekirdek-çekirdek çarpışmalarında alt-süreç düzeyinde ilk defa TeV skalasına ulaşılabilecektir. Proton-proton çarpışmalarının incelenmesine yönelik iki büyük detektörün, ATLAS ve CMS, yapımı tamamlanmak üzeredir. Burada öncelikli amaç Higgs bozonunun (zayıf etkileşmeleri taşıyan ara bozonlara ve temel fermiyonlara kütle sağlayan Higgs mekanizmasının ön gördüğü skalar parçacık) bulunmasıdır. Eğer Standart Model doğru ise ve Higgs bozonunun “görünmez” bozonum modlarının baskın olması gibi özel durumlar söz konusu değilse, bu parçacık LHC'nin ilk çalışma yılında gözlenecektir.

Protonların çok yüksek enerjilerde çarpıştırılması ile amaçlanan evrenin oluşumunun başlangıcı olarak bilinen Büyük Patlamadan sonra saniyenin milyarda biri gibi çok kısa zaman aralığındaki şartların benzerlerini laboratuvar ortamında yaratmak ve bu şekilde evrenin şu andaki durumunu ve işleyiş mekanizmalarını anlamaya çalışmaktır. Hızlandırıcılarda çok yüksek enerji ve çok yüksek yoğunluktaki parçacık demetlerinin sıklıkla çarpışmalarının sonucunda sürprizlerin de beklenmesi doğaldır. LHC deneyinin en önemli ve bilim dünyasında ilgiyle beklenen tarafı budur [3,4]. Diğerleri CMS deneyi, LHCb deneyi, Alice deneyi ve Totem deneyi'dir. ATLAS ve CMS genel amaçlı, LHCb b-fiziği üzerine, Alice ağır iyon fiziği ve Totem ise toplam tesir kesiti ölçümü üzerinedir. Bu dört dedektör biraz daha yakından incelenirse;



Şekil 2.1: LHC Çemberi

2.2 CERN 'deki Deney ve Dedektörler

LHC İngilizce "Large Hadron Collider" kelimelerinin başharflerinden oluşan kısaltmadan oluşmuştur. Anlamı "Büyük Hadron Çarpıştırıcısı" demektir. CERN'de 2008 yılında devreye girip yüksek enerjili parçacık fiziği deneyleri yapılmasına imkân verecek bir projedir. LHC çalıştığı zaman 7 TeV'lik (TeV=Tera elektron Volt) yüksek enerjili proton demetleri, her saniyede 40 milyon kez çarpışacaktır. Bu çarpışmalar yerin yaklaşık 100 m derinliğinde çevresi 27 km olan bir halkada gerçekleşecektir. LHC'de protonlar halkanın etrafına yerleştirilmiş yaklaşık 10,000 süperiletken mıknatıs tarafından yönlendirilecek ve zıt yönlerde dönen protonlar kütle referans sisteminde 14 TeV'de çarpışacaktır. Çarpışma noktalarındaki detektör sistemleri bu çarpışmaları izleyecek ve kaydedecektir. Parçacık fiziği deneylerinde küçük kütleli oluşları ve yüksek enerjileri nedeniyle çok uzun mesafelere gidebilen parçacıklar gözlenmeye çalışılmaktadır. Bu parçacıklardan bazıları kalınlığı metrelerce olan çok yoğun malzemelerden, geçerek ilerlemektedirler. Dolayısı ile detektörler, yoğun malzemelerden

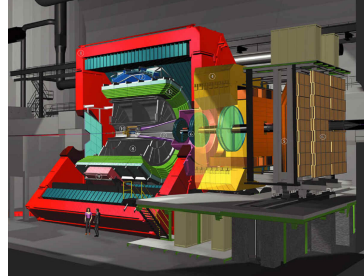
çok büyük hacimlerde inşa edilmektedir. Bu yolla bu parçacıklar durdurulabilir ve incelenebilirler [1].

LHC de süperiletken teknolojiyle $B= 10$ Tesla'ya ulaşmak mümkün görünmektedir. LHC'de ilk üç yıl $L=1033\text{cm}^{-2} \text{s}^{-1}$ luminositide(ışınlık) daha sonra $L=1034\text{cm}^{-2} \text{s}^{-1}$ luminositide çarpışmalar sağlanacaktır.

$$L = \frac{N^2 k_b f \sigma}{4\pi \epsilon_n \beta^*} F \quad (2.2)$$

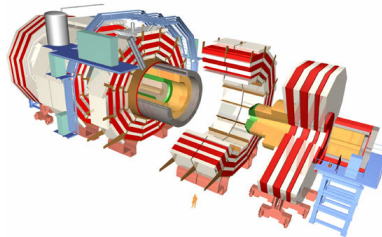
Yukarıda saymış olduğumuz Atlas deneyi, CMS deneyi, LHCb deneyi, Alice deneyi ve Totem deneylerini yakından inceleyelim.

· **ALICE** (A Large Ion Collider Experiment), quark-gluon plazma denilen yeni bir fazın oluşması beklenen çok yoğun enerji yoğunluklarında maddelerin fiziğini incelemeyi hedeflemektedir.



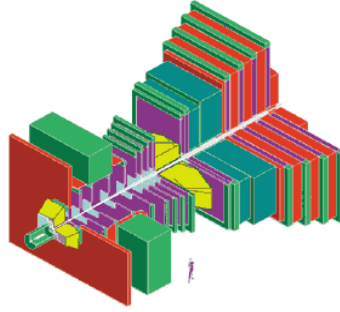
Şekil 2.2: Alice Dedektörü

· **CMS** (Compact Muon Solenoid) Süper simetri için kanıt bulmak amacıyla yüksek enerjide yeni bir fizik keşfedecektir.



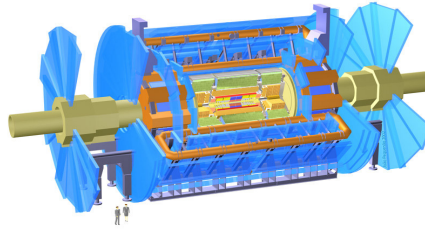
Şekil 2.3: CMS Dedektörü

· **LHCb** Yükn bozulması ve parite simetrisi üzerine çalışmada yoğunlaşmıştır. Bu etki, evrenin doğuşundaki madde/karşıit madde dengesizliğinden sorumlu olabilir. Aşağıdaki şekilde bu dört dedektörün LHC çemberine nasıl yerleştiğı ayrıntılı bir şekilde gösterilmektedir.



Şekil 2.4: LHCb Dedektörü

· **ATLAS** (A Toroidal LHC ApparatuS) maddelerin doğasını ve evrenimizi şekillendiren temel kuvvetleri keşfedecektir.



Şekil 2.5: Atlas Dedektörü

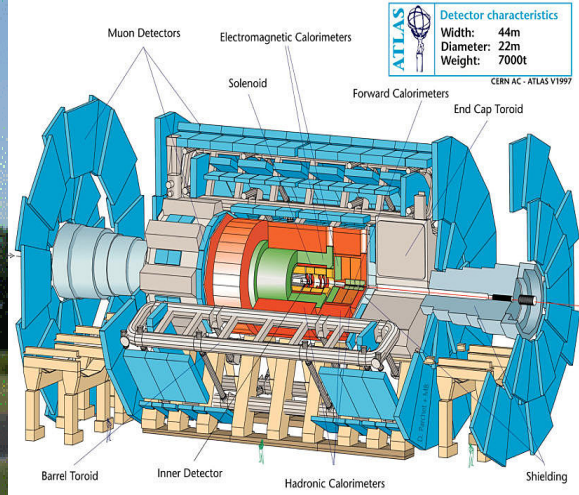
2.3 ATLAS Dedektörü

ATLAS (A Toroidal LHC ApparatuS) detektörü, LHC çemberinin ilk noktasında konumlandırılmıştır. ATLAS detektörü, şimdiye dek gözlenmiş veya gözlenmemiş birçok parçacığın izlerini, enerjilerini, momentumlarını ölçecek şekilde genel amaçlı olarak tasarlanmıştır. LHC nin çarpışma enerjisi olan 14 TeV ve ışınlığı olan 10^{34} p/cm²/s daha önce ki deneylerde ulaşılmamış özelliklerdir. Bu zor şartlar, ATLAS deneyini şimdiye dek yapılmış bütün parçacık fiziğı deneyleri arasında en büyüğü ve en karmaşığı olmaya itmiştir. ATLAS deneyi Standart Modeli deneysel olarak tamamlamak için gereken Higgs bozonunu keşfetmek

dışında, top kuark ile ilgili detaylı arařtırmalar yapmayı da amaçlar. Ayrıca yeni fizik modelleri (Süper simetri, BBT, Ek Boyutlar), mikro kara delikler, evrendeki madde-anti madde oransızlığı da arařtırılacak konular arasında yer almaktadır. [4]



Şekil 2.6: ATLAS ın büyüklüğünü CERN laboratuvarıyla karşılařtıran temsili resim.



Şekil 2.7: ATLAS Dedektörünün iç yapısı

ATLAS Dedektörü çarpıřtırıcılarda kullanılan katmanlı bir yapıya sahiptir. En iç kısımda çarpıřmadan çıkan yüklü parçacıkların izlerini belirleyen bir iz dedektörü, sonra elektronların ve fotonların enerjilerini bırakacakları bir elektromanyetik kalorimetre, onun dışında proton, nötron, pilyonlar gibi kuvvetli etkileşen parçacıkların enerjilerini bırakacakları bir hadron kalorimetresi ve en dışta zayıf etkileşen müonları algılamak üzere müon odacıkları bulunmaktadır. ATLAS Dedektörünün uzunluğu 44m, çapı 22m ve kütlesi de 7000 ton kadardır. Ayrıca ATLAS deneyinde 34 ülkenin 150 üniversitesi ve arařtırma kurumundan 2000 bilim adamı çalışmaktadır. Projenin maliyeti işletme giderlerinin dışında 550 milyon İsviçre Frankı veya 430 milyon dolar civarındadır.

Yukarıdaki şekilden de görüleceği gibi ATLAS Dedektörü dört ana kısımdan oluşmaktadır:

İç Detektör – yüklü parçacıkların momentumlarını ölçer

Kalorimetre – parçacıkların taşıdığı enerjileri ölçer

Müon Spektrometresi – müonları ölçer ve etiketler

Magnet Sistemi – yüklü parçacıkları momentum ölçümü için

Dedektörün iç yapısını biraz detaylandırmak gerekirse;

2.3.1 İç Detektör

- **Piksel Detektör**

Bir piksel sensör her biri 50 x 400 mikron olan 46,080 piksel içeren 16.4 x 60.8 mm silikon katmandır. Piksel dedektörün çember kısmı, yarıçap pozisyonu 50.5 mm, 88.5 mm ve 122.5 mm olan üç farklı silindirik katmana sahiptir. Bu üç çember katman, 20° 'lik azimutsal açıyla eğimli eş nokta çizgilerinden yapılmıştır. Bu katmanların her birinin içinde ayrı olarak 22, 38 ve 52 nokta çizgisi vardır. Her nokta çizgisi 13 piksel modülün bir araya getirilmesiyle oluşur. Uç bölgelerin her birinde üç disk vardır. Bir disk, her bir sektörde 6 modül olmak üzere 8 sektörden oluşmaktadır.

- **Yarıiletken Detektör –Semiconductor Tracker- (SCT)**

SCT sistemi, orta yarıçap uzaklığında her iz için momentum ölçümlerine, çarpışma parametresine ve doruk noktaya katkıda bulunan hassas ölçümlerin yapılması amacıyla dizayn edilmiştir. Çember SCT 'nin 8 katmanlı silikon mikro şerit dedektörlerinden her biri 6.36 x 6.40 cm den oluşur. Çember modüller, 30.0, 37.3, 44.7 ve 52.0 cm yarıçap uzaklıklarında karbon-fiber silindirlerle bağlanmıştır. SCT $|\eta| < 2.5$ aralığını kapsar.

- **Geçiş Radyasyonu Detektörü –Transition Radiation Tracker- (TRT)**

ATLAS 'ın dış detektörü örgü detektör ve geçiş radyasyonu detektörünün kombine bir halidir. Çember kısım 56 cm ve 107 cm yarıçap uzunlukları arasında yaklaşık 150 cm uzunluğunda 52,544 eksenel örgü içerir. Dış başlıklar 64 cm ve 103 cm (iç) ile ayrı olarak 48 cm ve 103 cm (dış) yarıçap uzunlukları arasında toplam 319,488 eksenel örgü içerir.

TRT, yüklü parçacık izlerini $|\eta| < 2.5$ ve $p_T > 0.5$ GeV aralığında, 0.170 mm çözünürlükte yaklaşık 36 iki-boyutlu ölçümü destekler.

2.3.2 Kalorimetre

- **Likit Argon Kalorimetre (LAr)**

Likit Argon kalorimetre birkaç bileşene ayrılmıştır: elektromanyetik örnekleme kalorimetresi, hadronik kalorimetre ve uç kalorimetre. Ek olarak, kalorimetre önünde enerji

kaybını düzeltmeye yardım etmek için elektromanyetik kalorimetrenin önünde LAr' ın bir katmanını içeren ön örnekleycilere sahiptir.

- **Silindir Kalorimetre**

Silindirik kalorimetre, çeliği yutucu materyal olarak kullanan büyük bir hadronik örnekleme kalorimetresidir. Bu kalorimetre merkezi genişliği $|\eta| < 1.7$ dir. Silindirik kalorimetre, iç yarıçapı 2280 mm ve dış yarıçapı 4230 mm olacak şekilde silindirik bir yapıya sahiptir. Bu yapı da 5640 mm uzunluğunda merkezi çember ve iki adet 2910 mm genişletilmiş çemberlere sahip alt bölümlere ayrılmıştır.

2.3.3 Müon Sistemi

Büyük çember toroid sistemi ile çevrili çember bölgede ($|\eta| < 1.0$), çarpışma eksenini etrafındaki hassas 'Monitored Drift Tubes' (MDTs) ve hızlı 'Resistive Plate Chambers' (RPCs) yapılarının üç katmanı ile müonların ölçümü yapılır. Büyük η genişliklerindeki bölgelere de dikey olarak üç katmanlı bölme yerleştirilmiştir. Burada, 'Thin Gap Chambers' (TGCs) yapıları tetikleme için kullanılmıştır. Yüksek parçacık akışının, daha fazla radyasyon toleranslı 'Cathode Strip Chamber' (CSC) teknolojisine ihtiyaç duyduğu uç başlıkların ve $|\eta| > 2.0$ durumu dışında müonlar için belirlenen ölçümler MDTs ile tekrar yapılır.

ATLAS müon sisteminin çemberinde, yarıçap uzaklıkları 5, 7.5 ve 10m olan, çarpışma eksenine ortak merkezli üç silindir içine müon bölmeleri kurulmuştur.

2.3.4 Magnet Sistemi

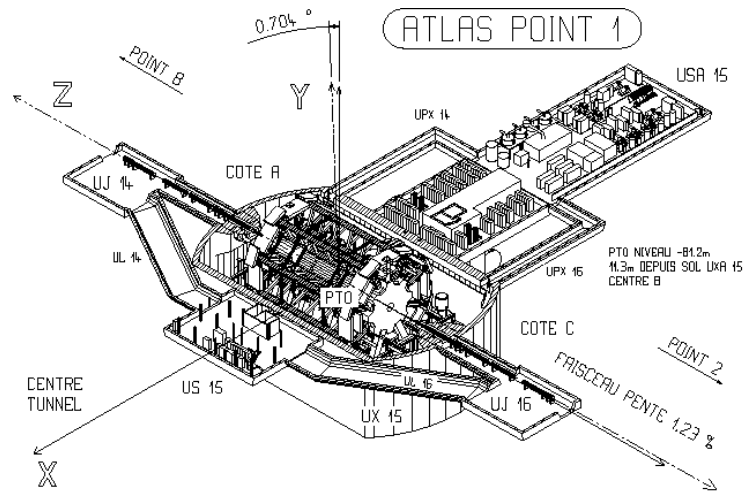
- **Merkezi Selenoid**

Merkezi ATLAS selenoidi 5.3m uzunluğa ve 2.4 m oyuğa sahiptir. İletken madde, alüminyum stabilizatörün merkezine yerleştirilmiş düz süperiletken içermektedir. Merkezi detektör hacmi içinde en yoğun manyetik alan 2.6 Tesla olacak şekilde, 2 Tesla'lık alan oluşturmak için tasarlanmıştır.

- **Toroid Magnet**

ATLAS Toroid Magnet sistemi 8 Barrel bobininden oluşur. Son başlık bobin sistemleri, Barrel Toroid 'lerin radyal kaplama oluşturmasıyla uyumlu olması için 22.5° döndürülmüştür.

ATLAS dedektörünün koordinat sistemi sağ eli bir sistemdir. x-ekseni LHC halkasının merkezini gösterecek şekilde, z-ekseni demet yönünü takip eder, y-ekseni yukarıyı gösterir. Birinci noktada pozitif z, % -1.23 eğimle sekizinci noktaya yönelmiştir. Azimutsal açı $\phi = 0$ pozitif x-eksenini karşılar ve + z-ekseninin içine bakacak şekilde saat yönünde artar. Polar açı θ + z-ekseninden açılmayı gösterir. Şekilde ATLAS koordinat sistemi detaylı şekilde verilmiştir.



Şekil 2.8 : ATLAS Koordinat sistemi

Enine momentum, p_T , LHC çarpışma eksenine dikey çizgi şeklinde görülen momentum olarak belirtilir.

ATLAS Deneyinin başlamasından önce, deney başladıktan sonra ne bulunacağı hakkında fikir sahibi olmamız gerekmektedir. Bu yüzden ATLAS Deneyinde çalışan bilim insanları ATLAS Dedektörünü kısmen ve hızlı olarak modellleyen (ATLFAST) ve Tam olarak modellleyen (ATHENA) gibi çeşitli modelleme programları yazmışlardır. ATLFAST dedektörü hızlı modellediği ve detaylara girmediği için kabaca ve hızlı sonuç vermektedir. Fakat ATHENA dedektörün en ince ayrıntısına kadar dedektörü modelleyen bir programdır. Bu tezde ATHENA kullanarak üretilen parçacıklar modellenmektedir. Parçacık üretici olarak PYTHIA programı kullanılmıştır.

3. DÖRDÜNCÜ STANDART MODEL FERMİYON AİLESİ

Standard modeli ilk bölümde açıklarken yetersiz kaldığı bazı kısımlar olduğundan bahsetmiştik. Bunlar özetle

- ❖ Temel fermiyon kütleleri ve karışımları.
- ❖ Fermiyon ailelerinin varlığı ve sayıları.
- ❖ Tek sağ-taraf parçacıkları ve zayıf izospin çiftlere sahip sol taraf parçacıklarının tayini olarak belirlenebilir.

Her biri sahip olduğu ayar grubu ile tanımlandığından dolayı SM şiddetli, zayıf ve elektromanyetik kuvvetleri tam olarak birleştiremez.

Bu soruları çözmek için SM ötesinde farklı yaklaşımlar önermiştik: elektrozayıf simetrisinin genişletilmesi, büyük birleşik teori (GUT), süpersimetri (SUSY) ...

Demokratik kütle simetrisi (DMM) yaklaşımı birinci problemi çözmek için genişletilmiştir.

Fermiyonlar arasındaki bağlaşımları ve kütleleri bulmak için DEMOKRATİK KÜTLE MATRİSİ (DMM) modeli önerilmiştir. Bu modele göre dördüncü SM ailesinin olması gerektiği ortaya çıkmıştır.

3.1 Fermiyon Ailesinin Sayısı ve Teorik Yaklaşım

Aşağıdaki Çizelgeyi üç fermiyon ailesi için giriş kısmında vermiştik. Dördüncü fermiyon ailesinin eklenmesi durumunda ise,

Çizelge 3.1: Leptonlar ve Kuarklar

Nesiller	1	2	3	4.Aile(?)
Leptonlar	e ve	μ $\nu\mu$	τ $\nu\tau$	l_4 ν_4
Kuarklar	u D	c S	t b	u_4 d_4

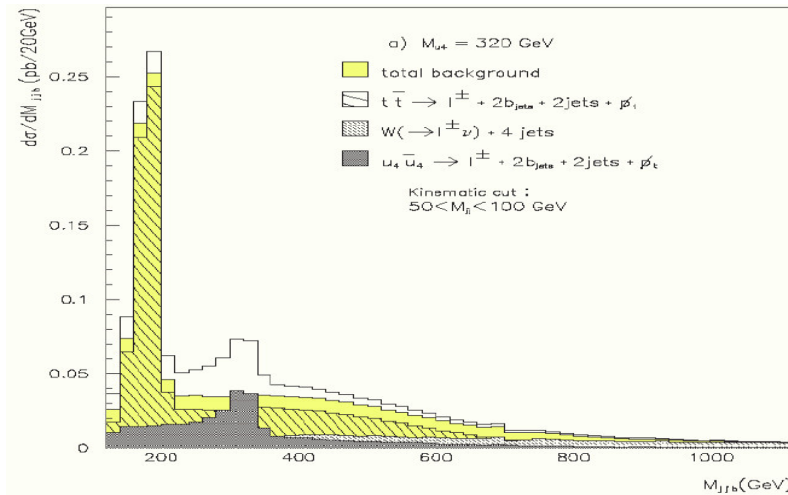
Yukarıdaki Çizelgede verdiğimiz dördüncü SM fermiyon ailesi varsa bunlar çift üretim süreçleri ile LHC’de çok sayıda üretilebilecektir.

- DMM yaklaşımına göre, $m_4:300-700$ GeV
- Tevatron verilerine göre, $m_4 > 199$ GeV (%95 CL.)

Fazladan SM aileleri, hadron çarpıştırıcılarında gluon füzyonu aracılığı ile Higgs bozonu üretim tesir kesitlerinde de önemli bir artışa neden olacak ve bu etki Tevatron’ da bile kolayca gözlemlenecektir.

LHC’ de ise dördüncü aile gözlemlenmesi beklenmektedir. 700 GeV kadar kütleli $Q_4 = (u_4, d_4)$ kuarklarının sinyallerini gözlemlenmek mümkün olabilecektir.[5]

Çizelge 3.2. LHC de dördüncü aile üretimi



3.2 Dördüncü Ailenin Gerekliliği

Higgs alanı tüm uzay zamanı kaplayan, SM’ de parçacıkların nasıl kütle kazandığını açıklamamızı sağlayan bir terimdir. Bir parçacık uzay zamanda hareket ettiği zaman bu Higgs artalanı nı deforme eder. Bu deformasyon Higgs alanı kuantumlarının hareket eden parçacığı tamamen çevrelemesi ya da Higgs alanının parçacık etrafında yoğunlaşması olarak ta düşünülebilir. Parçacık bu şekilde kütle kazanmış olur.

Aynı Higgs alanı W^+ , W^- ve Z^0 bozonlarına da kütle verir.

$$M_z = 1/2(g\delta) \quad M_w = 1/2(g\delta) \quad (3.1)$$

$$g = \frac{\sqrt{4\pi\alpha_{em}}}{\sin\theta_\omega} \quad \bar{g} = \frac{\sqrt{4\pi\alpha_{em}}}{\sin\theta_\omega \cos\theta_\omega} \quad \alpha_{em} = \frac{1}{137} \quad (3.2)$$

Burada $\alpha=1/137$ ince yapı sabiti olarak tanımlanır.

$\sin\theta_\omega$ Standart modelin Weinberg açısı denilen parametre $\eta=245$ GeV

Higgs alanının vakum beklenti değeridir.

Simetri kırılmasından önce aynı kuantum sayılı fermiyonlar ayırt edilemezler. Bu nedenle fermiyon-Higgs etkileşmesinde bu fermiyonlar aynı etkileşme sabitine sahip olmalıdırlar.

Bu nedenle fermiyon-Higgs etkileşmesinde bu fermiyonlar aynı etkileşme sabitine sahip olmalıdırlar. Örnek olarak yüklü leptonların kütle matrislerini incelersek, temel parçacıkların hareket denklemlerini ifade eden Lagrangian'da 9 farklı terim yazabiliriz.

$$L_m^l = \partial_{ee}\eta\bar{e}e + \partial_{\mu\mu}\eta\bar{\mu}\mu + \partial_{\tau\tau}\eta\bar{\tau}\tau + \partial_{e\mu}\eta\bar{e}\mu + \partial_{e\tau}\eta\bar{e}\tau + \partial_{\mu e}\eta\bar{\mu}e + \partial_{\mu\tau}\eta\bar{\mu}\tau + \partial_{e\tau}\eta\bar{e}\tau + \partial_{\tau e}\eta\bar{\tau}e + \partial_{\tau\mu}\eta\bar{\tau}\mu \quad (3.3)$$

Fakat yukarıda belirtilen nedenlerle etkileşme sabitlerinin eşit olması gerekmektedir:

$$\partial_{ee} = \partial_{\mu\mu} = \dots = \partial_{\tau\tau} = \sqrt{4\pi\alpha_{em}} \quad \text{veya} \quad \frac{\sqrt{4\pi\alpha_{em}}}{\sin\theta_\omega} \quad (3.4)$$

Kütle matrisi;

$$\partial\eta(\bar{e}\bar{\mu}\bar{\tau}) \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} e \\ \mu \\ \tau \end{pmatrix} \quad (3.5)$$

formunu alır ve demokratik kütle matrisi (Democratic Mass Matrix - DMM) olarak adlandırılır.

Yukarıdaki matrisin özdeğerlerini elde edecek olursak;

$$\begin{vmatrix} 1 - \lambda & 1 & 1 \\ 1 & 1 - \lambda & 1 \\ 1 & 1 & 1 - \lambda \end{vmatrix} = 0 \quad (3.6)$$

$$(1 - \lambda)^3 + 1 + 1 - (1 - \lambda) - (1 - \lambda) - (1 - \lambda) + 1 + 3\lambda^2 - 3\lambda - \lambda^3 + 2 - 1 + \lambda - 1 + \lambda - 1 + \lambda = 0$$

$$3\lambda^2 - 3\lambda - \lambda^3 + \lambda + \lambda + \lambda = 0$$

$$-\lambda^3 + 3\lambda^2 = 0$$

$$+ \lambda^2(-\lambda - 3) = 0$$

$$\lambda_1, \lambda_2 = 0 \text{ ve } \lambda_3 = 3 \text{ olarak bulunur.} \quad (3.7)$$

Bulmuş olduğumuz $\lambda_1, \lambda_2 = 0$ ve $\lambda_3 = 3$ değerleri kütle değerleri ile orantılıdır.

Bu singüler matrisin üç özdeğerinden ikisi sıfırdır.

$M_1 = M_2 = 0$ üçüncüsü ise $m_3 = 3\partial_\eta$ olur. Etkileşme sabitini $\partial = \sqrt{4\pi\alpha_{em}}$ şeklinde alırsak

$\eta = 245$ GeV (boşluk beklenen değerini) değerini göz önünde tutarak, $m_3 = 240$ GeV bulunur.

Aynı yolla aynı değeri nötrino, üst ve alt kuarklar içinde alabiliriz. Bu değeri üçüncü aile fermiyonlarının deneysel kütleleri ile karşılaştırırsak, üç aile durumunda yanlış sonuçlara getirdiğini görürüz.

Farz edelim ki dördüncü fermiyon ailesinin var olduğu düşünölsün. Bu durumda kütle matrisinin özdeğerlerinden ilk üçü $m_1 = m_2 = m_3$ olacaktır. Dördüncü kütle ise $4m_0$ değerini alacaktır. Eğer $\partial = \sqrt{4\pi\alpha_{em}}$ ise $\lambda_4 = 320$ GeV alınır. Eğer $\partial = \frac{\sqrt{4\pi\alpha_{em}}}{\sin \theta_\omega}$ ise $\lambda_4 = 620$ GeV olur.

İkinci seçim daha doğaldır.

Doğada, en azından ilk üç aileden leptonlar ve kuarklar sıfırdan farklı kütlelere sahiptir ki bu da M^0 kütle matrisinin değişikliğe ihtiyacı olduğu anlamına gelir.

$$M^0 = a\delta \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} \rightarrow M = 4a\delta \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.8)$$

M^0 matrisi :

$$M^0 = a\delta \begin{pmatrix} 1 & 1+\gamma & 1+\beta & 1 \\ 1+\gamma & 1+2\gamma & 1+\beta & 1 \\ 1+\beta & 1+\beta & 1+\alpha & 1-\alpha \\ 1 & 1 & 1-\alpha & 1+\alpha \end{pmatrix} \quad (3.9)$$

γ parametresi, birinci aile fermiyonları için kütleler üretilirken, β parametresi; ikinci aile fermiyonları için kütleler üretir. α parametresi ise üçüncü aile fermiyon kütlelerini üretir.

Yukawa etkileşiminden dolayı örneğin bir d kuarkın kütlesi:

$$L_Y^{(d)} = a_d (\bar{u}_L \bar{d}_L) \begin{pmatrix} \varphi^+ \\ \varphi^- \end{pmatrix} d_R + h.c. \Rightarrow L_m^{(d)} = m_d \bar{d} d \quad \text{Şeklinde bir lagrangian ile verilebilir.}$$

Burada;

$$m_d = a_d \delta$$

$$\delta = 245$$

Nötrino bir Dirac parçacığı ise

$$m_u = a_u \delta \quad m_e = a_e \delta \quad m_{\nu_e} = a_{\nu_e} \delta \quad (3.10)$$

Birinci Varsayım: Simetri kırılmasından önce bütün kuarklar kütesizdir. d_0, s_0, b_0, \dots arasında fark yoktu. Bu yüzden $m_d = a_d \delta$ denklemindeki

$$a_{ij}^d \approx a^d, a_{ij}^u \approx a^u, a_{ij}^l \approx a^l, a_{ij}^v \approx a^v \quad (3.11)$$

SM fermiyonlarının her bir çeşidi için (n-1) tane kütesiz parçacık, sadece bir tane de ağır kütleli parçacık vardır.

$$m = na\delta$$

Burada, δ Higgs alanı için vakumun beklenen değeri, a ise Fermiyon-Higgs etkileşmesinin büyüklüğü olarak tanımlanır.

Fermiyonların (up kuarklar, down kuarklar, yüklü leptonlar ve nötrinolar) dört tipi için Dirac kütlelerini veren sadece bir Higgs çifti olduğundan dolayı ikinci yaklaşımı yapmak doğaldır.

İkinci Varsayım: Sadece bir Higgs çifti içermektedir. Bu sebeple fermiyonların her biri için Yukawa çiftleri neredeyse verilen ifadedeki gibi olmalıdır.

$$a^d \approx a^u \approx a^l \approx a^v \approx a$$

Bu değerleri yukarıdaki gibi alırsak aşağıdaki değer elde edilir.

$$m_4 = 4g\delta = 8m_W = 640\text{GeV} \quad (3.12)$$

Eğer g , elektromanyetik bağlaşım sabiti olan e değeri ile yer değiştirirse yeni m_4 değerini elde ederiz.

$$m_4 = 320\text{GeV} \quad (3.13)$$

Ayrıca burada,

g : SU(2) için ayar çiftlenim sabiti.

e : elektromanyetik çiftlenim sabiti.

Üçüncü Varsayım: α değeri $e = 9w\sin\theta_w$ ile $g_z = 9w/\cos\theta_w$ değerleri arasındadır. Bu yüzden dördüncü SM ailesi fermiyonları hemen hemen dejenere ve $320 < M_4 < 730$ GeV değerleri arasındadır.

DMM'ye göre varlığı kabul edilen dördüncü SM fermiyon ailesi için üst limit $M_4 \approx 1$ TeV şeklindedir ve aynı zamanda bu değer yüksek enerjilerdeki üniterlik değeridir.

4. ATHENA DEDEKTÖR MODELLEME PROGRAMI

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 programını çalıştırabilmek için CERN'den (Nükleer Araştırmalar için Avrupa Organizasyonu) alınmış özel (büyük disk alanı ayrılması ve uzaktan erişim ile orada kurulu olan mevcut bilgisayar sisteminin-bilgi kütüphanelerinin kullanılması açısından) çalışma alanının olması gerekir. Athena, Linux işletim sistemi komutlarıyla çalışabilen bir programdır (Python yazılım dilini kullanır). ATHENA çalıştırılırken iki farklı yolla sonuçlara ulaşılabilir. Bunlar "Kits" ve "Releases" dir. Tez çalışmasında internet erişimi ile yeterli çalışma imkânı veren "Releases" seçilmiştir. Linux altında ayrılan alanın çalışma kabuğuna göre betik dosyaları için dosya uzantısı olarak. sh veya. csh 'tan birisi seçilmelidir. Bu tezde çalışılan kabuk .csh destekli olduğu için .csh uzantısı ile çalışılmıştır.

Athena programını çalıştırabilmek için Secure SHell Linux ara yüzü ile CERN'deki üyeliğimize bağlanmalıyız. Bunun için SSH program dosyaları indirilir.

4.1. SSH (Secure SHell Linux Uzaktan Erişim Programı) 'in Yüklenmesi

CERN üzerinde kurulu olan sunucu sistemi lxplus.cern.ch web adresi üzerinden ssh erişime imkan verir. Çalışma alanı ile CERN-LHC, ATLAS çalışma alanına girilir ve aşağıdaki adımlarla çalışma alanı olay üretici programına uygun hale getirilir.

Çalışma Alanının Düzenlenmesi

```
> cd $HOME
> mkdir cmthome
> mkdir testarea
> mkdir testarea/13.0.30 (çalışılacak klasör)
> cd cmthome
> source ~/cmthome/setup.csh -tag=13.0.30
> source /path_to_kit/CMT/*/mgr/setup.csh
```

adımlarıyla çalışma alanı ve gerekli klasörler oluşturulur. Daha sonra requirements dosyası içeriği oluşturulur:

```
#-----
set CMTSITE CERN

set SITEROOT /afs/cern.ch

macro ATLAS_DIST_AREA /afs/cern.ch/atlas/software/dist

macro ATLAS_TEST_AREA ${HOME}/testarea

apply_tag setup

apply_tag simpleTest

use AtlasLogin AtlasLogin-* $(ATLAS_DIST_AREA)

#-----

> cmt config
```

Her Girişte çağırılması gerekli versiyon ve bunun için yazılması gereken komutlar şunlardır:

```
> source ~/cmthome/setup.csh --tag=13.0.30 (bu çalışılan en son versiyondur, her yeni çıkan versiyon için bu düzenlenmelidir.)
```

Çalışılan program yazıldıktan ve çalıştırdıktan sonra çıktı dosyaları liste ve histogram şeklinde elde edildiğinden histogram çıktı dosyaları için ROOT grafik ve histogramlama programı kullanılmaktadır.

4.2 Root Grafik Analiz Programı

Root programı C ve C++ dillerine dayalı grafiksel analizleme programıdır. Yazılan kodlarla ya da ara yüz ile çalışma imkânı sağlar. Root ile iki, üç boyutta grafik çizilebilir, matematiksel fonksiyonlara uyarlama gibi fiziksel uygulamalar yapılabilir.

ROOT kurulumu için <http://root.cern.ch/root/Availability.html> adresinden program indirilebilir. ROOT programının Windows, Linux işletim sistemlerinde çalışabilen güçlü bir programdır. Solaris sistemi altında da root programını çalıştırabiliriz.

ROOT programının Linux ortamında kurup çalıştırabilmek için kurulum sistemi olan profile'in içine;

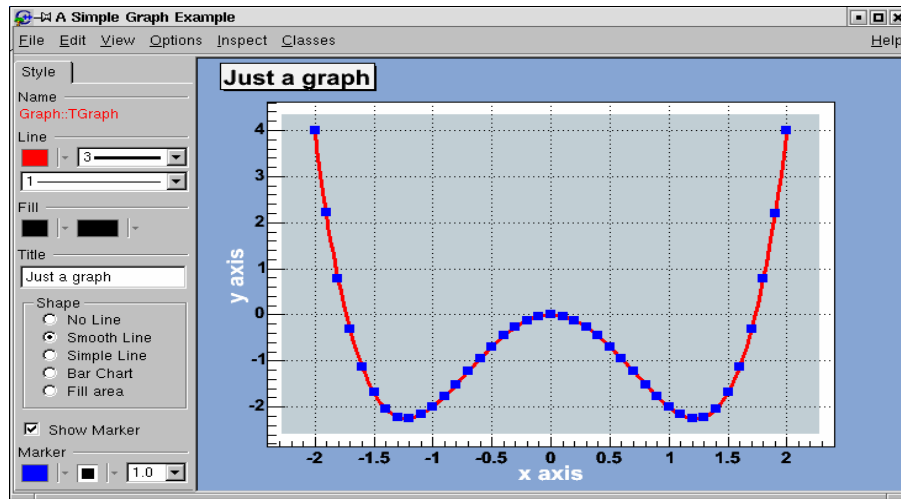
```
export ROOTSYS=/home/kullanici/root/
```

```
export PATH=$ROOTSYS/bin:$PATH
```

```
export LD_LIBRARY_PATH=$ROOTSYS/lib:$LD_LIBRARY_PATH
```

adımlarıyla tanımlanan kütüphane yollarının tanımlanması gerekir.

Aşağıdaki şekil root programıyla çizilebilecek tek boyutta bir grafikdir. Çizelgede görüldüğü gibi araç çubuğu ve grafik ayarlamaları için sütun vardır. Sol sütunla grafikte istenilen ayarlamalar yapılabilir. Örneğin grafik iki ya da üç boyuta taşınabilir.



Şekil 4.1: ROOT Programının çalışma menüsü

4.3 Dördüncü Aile Fermiyonlarının ATHENA ve PYTHON Yazılım Dillerini Kullanarak Elde Edilmesi:

ATHENA ve PYTHON dillerini kullanarak ATLAS dedektörünün tam modellemesine ilişkin bütün bilgileri eksiksiz içeren ve gelişigüzel olayların üretilebileceği ayrılmış alana giriş yapılmalıdır. Bu giriş:

```
> cd testarea/13.0.30
```

```
> cd PhysicsAnalysis/AnalysisCommon/UserAnalysis/run
```

```
> get_files PDGTABLE.MeV
> get_files jobOptions.pythia.py
> athena.py jobOptions.pythia.py > gamze_gen.out
> athena.py jobOptions.pythia.py | tee gamze_gen.out
```

Kodlarıyla liste çıktısını ekrana yazdırırız. Program kodlarını modifiye etmek için önce pico myGenOptions.py ile dosya içeriğini düzenlememiz gerekmektedir. Bu dosyanın içeriğine,

```
#-----
# Private Application Configuration option
# JobOptions 4
# Single Production of Fourth Family Quarks
# CERN, April 2008
#-----
from AthenaCommon.AppMgr import ServiceMgr
ServiceMgr.MessageSvc.OutputLevel = INFO
#-----
# Event related parameters
#-----
# Number of events to be processed (default is 10)
theApp.EvtMax = 100
#-----
# Algorithms Private Options
#-----
from AthenaServices.AthenaServicesConf import AtRndmGenSvc
ServiceMgr += AtRndmGenSvc()
#
from AthenaCommon.AlgSequence import AlgSequence
job=AlgSequence()
#
from Pythia_i.Pythia_iConf import Pythia
# Generate pp->q4+jX events
job.Pythia.PythiaCommand = ["pysubs msel 0", "pysubs msub 83 1",
                             "pypars mstp 1 4",
```

#PMAS

"pydat2 pmas 7 1 320.", "pydat2 pmas 8 1 320.",

VCKM in pydat2

"pydat2 vckm 1 4 0.044", "pydat2 vckm 2 4 0.46",

"pydat2 vckm 3 4 0.47", "pydat2 vckm 4 4 0.57",

"pydat2 vckm 4 1 0.063", "pydat2 vckm 4 2 0.46",

t-quark decay

"pydat3 mdme 41 1 0", "pydat3 mdme 42 1 0",

"pydat3 mdme 43 1 0", "pydat3 mdme 44 1 0",

"pydat3 mdme 45 1 0", "pydat3 mdme 46 1 1",

"pydat3 mdme 47 1 0", "pydat3 mdme 48 1 0",

"pydat3 mdme 49 1 0", "pydat3 mdme 50 1 0",

"pydat3 mdme 51 1 0", "pydat3 mdme 52 1 0",

"pydat3 mdme 53 1 0", "pydat3 mdme 54 1 0",

b' quark decay

"pydat3 mdme 56 1 0", "pydat3 mdme 57 1 0",

"pydat3 mdme 58 1 0", "pydat3 mdme 59 1 0",

"pydat3 mdme 60 1 0", "pydat3 mdme 61 1 1",

"pydat3 mdme 62 1 0", "pydat3 mdme 63 1 0",

"pydat3 mdme 64 1 0", "pydat3 mdme 65 1 0",

t' quark decay

"pydat3 mdme 66 1 0",

"pydat3 mdme 67 1 0",

"pydat3 mdme 68 1 0", "pydat3 mdme 69 1 0",

"pydat3 mdme 70 1 0", "pydat3 mdme 71 1 1",

"pydat3 mdme 72 1 0", "pydat3 mdme 73 1 0",

"pydat3 mdme 74 1 0", "pydat3 mdme 75 1 0",

W boson decay

"pydat3 mdme 190 1 0",

"pydat3 mdme 191 1 0",

"pydat3 mdme 192 1 0", "pydat3 mdme 193 1 0",

"pydat3 mdme 194 1 1", "pydat3 mdme 195 1 0",

"pydat3 mdme 196 1 0", "pydat3 mdme 197 1 0",

"pydat3 mdme 198 1 0", "pydat3 mdme 199 1 0",

```

"pydat3 mdme 206 1 1","pydat3 mdme 207 1 1",
"pydat3 mdme 208 1 0","pydat3 mdme 209 1 0"]
from TruthExamples.TruthExamplesConf import DumpMC
job += DumpMC()
#-----
# Pool Persistency
#-----
from AthenaPoolCnvSvc.WriteAthenaPool import AthenaPoolOutputStream
Stream1.OutputFile = "pythia4fam.pool.root"
Stream1.ItemList += [ 'EventInfo#*', 'McEventCollection#*' ]
#####
yazılarak düzenlemeler yapılır. Çalıştırmak için ise
> athena.py myGenOptions.py > gamze_gen.out

```

yazılır. Programın başarılı bir şekilde çalıştığını ise ekran çıktılarının içinde aşağıdaki satırı görerek doğruladık.

```
-rw-r--r-- 1 lloyd hep 358673 Jan 23 11:49 pythia.pool.root
```

ATHENA ile yapılan çalışmalarda, Standard Model Dördüncü Aile fermiyonlarının protondan üretimi ve bozunumları ($pp \rightarrow u_4 X$, $pp \rightarrow \bar{u}_4 X$, $pp \rightarrow \bar{d}_4 X$, $pp \rightarrow d_4 X$ Standart Modelde, u_4 , W^+ ve b kuark'a bozunur. ($u_4 \rightarrow W^+ b$) ve \bar{u}_4 , W^- ve b kuarka bozunur ($\bar{u}_4 \rightarrow W^- b$). Burada W 'ler leptonik kanalda bir lepton ve bir nötrinoya bozunur, hadronik kanalda iki jet'e bozunur.) için olan PYTHIA programı ATHENA tam dedektör modelleme programı arayüzü kullanılarak yazıldı.

4.4 PYTHIA Programı

PYTHIA programı T.Sjöstrand tarafından yazılmış uzun bir FORTRAN programıdır. Yapılan çalışmada 6.4 lük sürümü kullanıldı. Bu program leptonlar, hadronlar ve bunlar arasındaki yüksek enerjili çarpışmalar için Monte Carlo tekniklerini kullanarak olay üreten bir programdır.

Etkileşmede çıkan parçacıkların bozunma genişlikleri, saçılma tesir kesiti, ilk durum ve son durum ışınasal katkıları ile patronların hadronlaşması için gerekli alt programları içermektedir. 1978'de Lund grubu tarafından yazılan JETSET olay üretici programların başlangıcıdır. Daha sonra PYTHIA programı yazılmıştır. Bu iki programın birleştirilmeden

önceki son sürümleri JETSET 7.4 ve PYTHIA 5.7 olarak bilinmektedir. PYTHIA 6,1'de bu iki program birleştirilmiştir. Program seçilen fizik modeli çerçevesinde Monte Carlo teknikleri kullanarak yüksek enerji fiziği olayları üretmektedir. Üretilen olayların analizi ve histogram şeklinde verilmesi için gerekli alt programlar da bulunmaktadır. Program web adresinden elde edilebilir. Linux işletim sistemi altında bu fortran programını derlemek için **g7pythia62xx.f – opythia62xx.o** yazılır. Kullanıcı hazırladığı örnek programları PYTHIA ile bağlaması (link) için nesne dosyasından arşiv dosyası yapılmalıdır. Bunun için **ar rv libpythia62xx.a pythia62xx.o** yazılır. Daha sonra **ranlib libpythia62xx.a** yazarak bu dosyayı rasgele erişime hazırlamış oluruz. Genellikle bu dosya CERN kütüphane dosyaları (CERNLIB) ile birlikte kullanılmaktadır. Bu nedenle oluşturulan bu dosya **cp libpythia62xx.a /cern/pro/lib** komutu ile normal erişim yoluna kopyalanır.

Yüksek enerji fiziği süreçleri genelde karmaşık adımlardan oluşur. Ancak, ilk yaklaşım olarak bütün süreçler, temel parçacıklar (leptonlar, kuarklar ve ayar bozonları) arasındaki etkileşmeler seviyesinde basit bir yapıya sahiptir. Karmaşık yapılar çarpışmalardan sonra oluşmaktadır. Olay üreticiler böyle karmaşık durumlarda yardıma yetişir. Gerçek bir dedektörde gözlenebilecek olayları üretmek için bilgisayarlar kullanılmaktadır. Bir olay üreticisinin çıktısı gerçek verilerdeki aynı ortalama davranışı ve aynı dalgalanmaları göstermelidir. Verilerdeki dalgalanmalar ilgili teorinin kuantum mekaniksel davranışından kaynaklanmaktadır. Olay üreticilerde istenen olasılık dağılımlarına uygun olarak değişkenleri seçmek ve olayların rasgele olmasını sağlamak için Monte Carlo teknikleri kullanılır. Bir olay üreticisi birkaç değişik şekilde kullanılabilir. Bunlar; olay çeşitleri ve oranları, dedektör performansı, dedektör alım düzeltmeleri, analiz teknikleri, gözlem sonuçlarının model çerçevesinde yorumlanması olarak ayrılabilir.

Bir olay üreticisi, bir deneyin bütün analiz basamaklarında yer alabilir. Gerçekte, bir çarpıştırıcı parçacıklar arasında etkileşmeler oluşturur. Bu olaylar dedektör tarafından algılanır ve ilgilenilen olaylar veri alım sistemi tarafından disklere kaydedilir. Bundan sonra bu olaylar istenirse yeniden oluşturulabilir yani, elektronik sinyaller (tel odaları, kalorimetreler ve diğerlerinden alınan sinyaller) yüklü izler veya yüksüz enerji depolanması şekline dönüştürülebilir. Dolayısıyla parçacıklar ve momentumları belirlenebilir. Bu bilgiden sonra olayların fizik analizine geçilebilir. Dedektörün bu olayları algılamadaki davranışının - olay üreticisi ile üretilen parçacıkların dedektör ile etkileşmesi, magnetik alandaki spiraller, kalorimetrelerdeki sağanaklar ve algılanamayan olaylar vb. – GEANT programı ile simülasyonu yapılmaktadır. Dedektör simülasyonu çıktısı dedektörün kaydettiği gerçek veriler ile aynı

formatta olmalıdır. Simülasyonda da olay yeniden oluşturma ve fizik analizleri zinciri takip edilir. Dedektör simülasyonu ve olay yeniden oluşturma işlemleri zinciri çok zaman alıcı bir süreçtir, bazen daha hızlı ve basitleştirilmiş işlemlerle hızlı simülasyon mümkün olmaktadır. Bunun için özel yazılmış programlar da bulunmaktadır, örneğin ATLFast programı ile ATLAS dedektörü hızlı simülasyonu yapılmaktadır. Bazı çalışmalarda sadece dedektörün önemli parametreleri dikkate alınarak olay üreticinin çıktısı kullanılabilir.

PYTHIA programı CERNLIB program kütüphanesi ile birlikte çalışmaktadır. CERNLIB kütüphanesinde bulunan alt programlar:

Temel matematik fonksiyonlar, denklemler ve özel fonksiyonlar, sayısal integralleme, minimum yapma işlemleri, lineer olmayan fit, interpolasyon, yaklaşımlar, lineer fit, matrisler, vektörler ve lineer denklemler, istatistik analiz ve olasılık veri giriş ve çıkışı, verilerin grafiksel gösterimi, veri eldesi, magnet ve demet dizaynı, elektronik, kuantum mekaniği, parçacık fiziği, rasgele sayı üreticiler, yüksek enerji fiziği simülasyonu, kinematik, faz uzayı, istatistik veri analizi ve temsili, sistem ile ilgili programlar şeklinde sıralanabilir.

CERNLIB programları web sayfasından elde edilebilir. Toplam 9 sıkıştırılmış dosya halinde elde edilebilen /cern dizinindeki CERNLIB klasöründeki dosyalar

cernlib.tar.gz

cernlib.tar.gz

cernbin.tar.gz

mclibs.tar.gz

include.tar.gz

lapack.tar.gz

geant321.tar.gz

gcalor.tar.gz

patchy.tar.gz şeklindedir. Bu dosyalar **plitar** adındaki küçük bir program yardımıyla kolayca uygun bir şekilde açılabilir, alt programlar /cern/2003/lib, /cern/2003/bin ve /cern/2003/include dizinlerine açılmaktadır. Bundan başka bir de patchy dosyaları /cern/patchy dizinine açılacaktır, bu dosyalar her zaman gerekli değildir. Genel olarak bazı

programlarda CERNLIB /**cern/pro/lib** şeklinde tanımlanmaktadır. Bunun için **ln -s /cern/2003 /cern/pro** şeklinde bir sembolik bağ (link) yapılmalıdır.

Fizik analiz iş ortamı (PAW) programı, CERNLIB ile birlikte çalışan bir uygulama programıdır. PAW programı ile bir olaya ait verilerden vektör grafik, histogram, histograma fit, grafiklerin çeşitli formatlara dönüştürülmesi ve basılması gibi birçok çalışma yapılabilir. Bu program PYTHIA veya diğer simülasyon programlarının çıktılarında renkli, görsel grafiklerin oluşturulması, histogramlar dan istatistik analizlerin yapılarak fizik sonuçlarının çıkarılması için kullanılmaktadır. PAW komut satırından çalıştırılmaktadır, PAW++ ise butonlarla ve daha görsel olarak kullanma imkanı sağlamaktadır. Grafikselle kullanıcı arabirimi kullanan PAW++ ile sinyal ve fon olaylarının ayrıntılı analizleri yapılabilir. Örneğin, belli bir aralıkta (bin) toplam sinyal ve fon olaylarının sayısının bulunması, bir bölgede maksimum gösteren sinyal olaylarının Gauss eğrisine fit edilmesi, vb. işlemler kolayca yapılabilmektedir. Pythianın ilk adımı; değişken ve fonksiyonların tanımlanması [IMPLICIT DOUBLE PRECISION, INTEGER,...], common block tanımları [COMMON/.../...], süreç seçimi [MSEL, MSUB(..)], kinematik sınırların seçimi [CKIN(..)], fizik parametrelerinin girilmesi, patron dağılım fonksiyonunun seçimi, gerekli olan simülasyon anahtarlarının seçimi, olay üretiminin başlatılması ve diferensiyel tesir kesitinin maksimum değerinin araştırılması [CALL PYINIT(...)], histogram tanımlamaları pythiada başlangıç aşamasını oluşturmaktadır.

İkinci adımı ise olay üretim aşamasıdır. Olay döngüsü ile istenen sayıda olay üretilir CALL PYEVNT, ilk birkaç olayın üretimi ve listelenmesi [CALL PYLIST(1)], ilgilenilen olayların analizi, olayların kaydedilmesi, olayların dedektör simülasyonu için kullanılması olay üretim adımına girmektedir.

Son aşama ise, bir çizelgede sonuçların gösterilmesi ve sürecin tesir kesitinin yazılması [CALL PYSTAT(1)], histogramların çizilmesi [CALL PYHIST], başka çıktı dosyalarının yazılmasıyla sonlandırmış olunur.

Histogramlar da kullanılan HBOOK1(.....) alt programı histogram hazırlanması için kullanılır. HFILL(...) alt programı histogramı doldurmak için kullanılmaktadır. Oluşturulan bu histogramlar PAW veya PAW++ ile açılarak histogramların istatistiksel analizleri yapılabilir. Histogramların üst üste çizdirilmesi, fit edilmesi, eğri altında kalan alanların hesaplanması, olasılık hesaplarının yapılmasında çok yaygın olarak kullanılmaktadır. Histogram dosyaları ve verileri üzerinde işlemler yapabilmek ve grafiklerin bilimsel nitelikte olması için PAW ve

PAW++ programlarına girdi dosyaları (kumac) hazırlanabilir. Bu dosyalar PAW içinden **exec xyz.kumac** şeklinde bir komutla yüklenebilir.

PYTHIA'nın 6.4 sürümünde tanımlı yaklaşık 496 süreç vardır. Standart Model ve ötesi modellerde (süpersimetri, kompozitlik vb.) tanımlı parçacıklar ve süreçleri incelenebilmektedir. Temel olarak ee, ep ve pp çarpıştırıcıları için mümkün fizik süreçleri tanımlanmıştır. Biraz zor olmasına rağmen yeni fizik için yeni süreçler dışarıdan girilebilir. PYTHIA'da parametreler, anahtarların ve süreçlerin seçiminde çok yoğun olarak kullanılmaktadır. İleri düzeyde araştırma yapabilmek için kullanıcı tarafından fiziksel sürece uygun program yazılarak çalıştırılmalıdır.

Standart Modelin Dördüncü Fermiyon Ailesinin tam dedektör modellemesininide dikkate alarak yapılan programın sonuçları elde edilmiştir. Sonuçlar "gamze_gen.out" ve "pythia.pool.root" isimli dosyalarda mevcuttur. ".out" dosyasından liste çıktısı ve "pool.root" dosyasından da histogram çıktısı elde edilir. Programda, 14 TeV kütle-merkezi-enerjisi kullanılmıştır ve parton dağılım fonksiyonu olarak CTEQ6M seçilmiştir. Liste çıktılarındaki bilgiler Çizelgelerde şu şekilde görülmektedir.

Çizelge 4.1 Diferansiyel tesir kesitinin maximum araştırılmasının sonuçları

ISUB	Subprocess name	Maximum value
181	q + qbar -> Q + Qbar, mass	2.0862E-02
182	g + g -> Q + Qbar, massive	3.5199E+00
183	f + q -> f' + Q, massive	9.8110E-10
196	Semihard QCD 2 -> 2	1.0296E+04

Parçacık/parton veri Çizelgesinde birçok parçacığa ait verilen bilgilerin yanında programını yazarak çalıştırılan dördüncü aile fermiyonlarının üretimleri $t'(u_4)$ and $b'(d_4)$ olan kuarkların isimleri ve detaylı özellikleri (parçacıkların orijinleri, kodları, bozdukları parçacıklar, renk, yük, antü yük, kütle, genişlik, dallanma oranları, yaşam süreleri) çizelgede gösterilmektedir. KF ve KC kodları parçacıkların sıkıştırılmış kodunu ifade eder.

Çizelge 4.2 Parçacık/parton veri Çizelgesi

KF	KC	particle	antiparticle	chg	col	anti	mass	width	w-cut	lifetime	decay
IDC	on/off	ME	Br.rat.	decay	products						
7	7	b'	b'bar	-1	1	1	320.00000	10.92786	109.27858	0.00000E+00	1
56	0	102	0.000000	g	b'						
57	0	102	0.000000	gamma	b'						
58	0	102	0.000000	Z0	b'						
59	0	0	0.042779	W-	u						
60	1	0	0.447206	W-	c						
61	0	0	0.150403	W-	t						
62	0	0	0.000000	W-	t'						
63	0	102	0.000000	h0	b'						
64	0	0	0.359612	H-	c						
65	0	0	0.000000	H-	t						
8	8	t'	t'bar	2	1	1	320.00000	10.55492	105.54923	0.00000E+00	1
66	0	102	0.000000	g	t'						
67	0	102	0.000000	gamma	t'						
68	0	102	0.000000	Z0	t'						
69	0	0	0.063416	W+	d						
70	0	0	0.463034	W+	s						
71	1	0	0.472780	W+	b						
72	0	0	0.000000	W+	b'						
73	0	102	0.000000	h0	t'						
74	0	0	0.000770	H+	b						
75	0	0	0.000000	H+	b'						

Çıktı için son Çizelge da dördüncü aile çift üretimleri için tesir kesitleri, olay sayıları görülmektedir. Bu Çizelge enerji 14 TeV ve luminositi 10^5 pb^{-1} düşük değerindeyken elde edilmiştir.

Çizelge 4.3 Tesir Kesiti ve Olay sayılarının istatistiği

I	Subprocess	I	Number of	I		I	
I		I	points	I	Sigma	I	
I	-----I	I	-----I	I	(mb)	I	
I	N:o Type	I	Generated	I	Tried	I	
I	0 All included subproc	I	2999	I	25904	I	4.103E-01
I	181 q+qbar ->Q+Qbar,mass	I	48	I	184	I	5.466E-03
I	182 g+g ->Q+Qbar,massive	I	2951	I	25720	I	4.049E-01
I	183 f+q ->f+Q, massive	I	0	I	0	I	0.000E+00

Sonuçta proton-proton çarpışmasından 140 sayfalık toplam liste çıktısı elde edilmiştir. Özet liste çıktısı aşağıda verilmektedir.

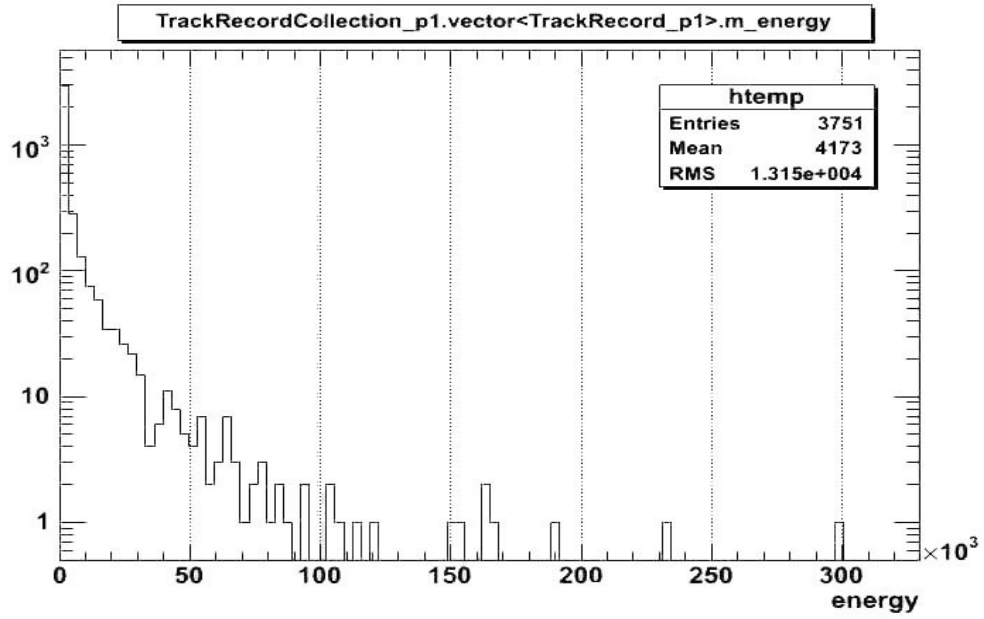
daha sonra tarafımızdan yazılan programı aşağıdaki komutlar ile

```
> athena.py myGenOptions.py > athena_gen.out
```

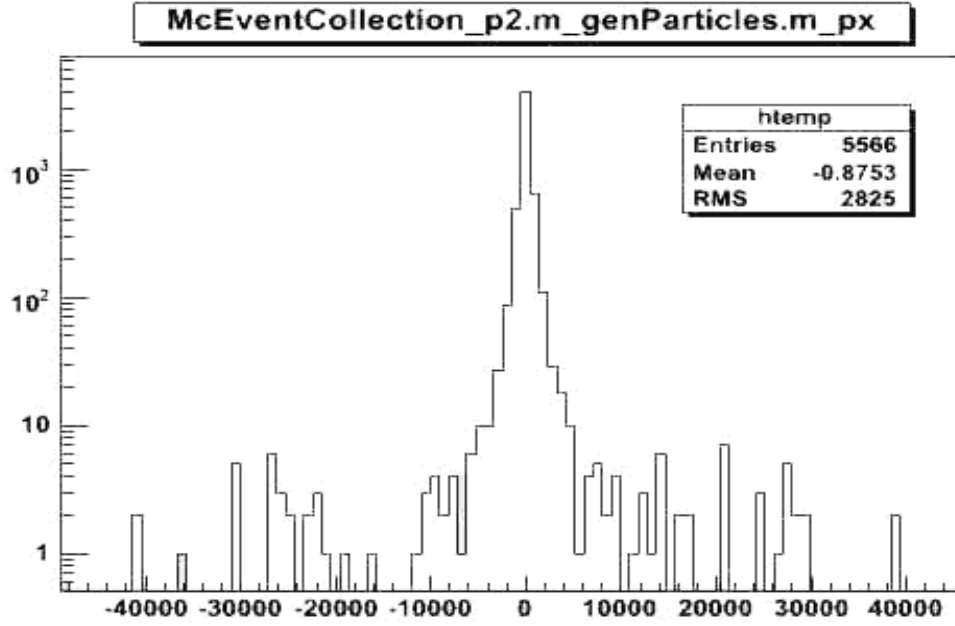
ile çalıştırırız ve ekran çıktısında aşağıdaki satırı görürüz:

-rw-r--r-- 1 lloyd hep 358673 July 23 11:49 pythia.pool.root

Bu pythia.pool.root dosyası grafik çıktıların elde edildiğini belirtmektedir. Grafik dosyasını ROOT grafik programı ile açtığımızda dedektörün en ince ayrıntısına kadar bütün grafiklerin elde edildiği görülmektedir. Bu grafiklerden bazıları:



Şekil 4.2: Atlas dedektörünün $pp \rightarrow Z, g$ veya Z, γ bozunum kanalları için kalorimetredeki enerji dağılımı.



Şekil 4.3: Üretilen olaylar için parçacığın x eksenindeki momentum dağılımı.

Şekildeki hitogramda x yöünde üretilen parçacıkların sahip olabileceği momentum değerlerinin çok küçük olduğu ve o değeri civarında dağılım gösterdiği görülmektedir. Bu beklenen bir durumdur. Çünkü momentumu yüksek olan parçacıklar geliş doğrultusunda geri dönme eğilimde bulunacaklar ve bu yönde z-ekseni olacağından, momentumun küçük kısmı parçacığın x ve y eksenini arasında dağılım gösterir. Burada sıfır civarında dağılım gösteren parçacıklara yumuşak parçacıklar jetler veya leptonlar denir.

4.5. gamze_gen.out Özet Dosya içeriği

Bu dosya üretilen olay sayısına göre büyüklüğü değişmektedir. Burada özet halinde verilmiştir.

```

Py:Athena          INFO including file "AthenaCommon/Compat.py"
Py:PropertyProxy  INFO Read module info for 2995 configurables from 10 genConfDb
files
Py:PropertyProxy  INFO No duplicates have been found: that's good !
Py:Athena          INFO including file "AthenaCommon/Bootstrap.py"
ApplicationMgr     SUCCESS
=====
                        Welcome to ApplicationMgr $Revision: 1.66 $
                        running on lxplus205.cern.ch on Sun May 18 02:56:31 2008
=====
ApplicationMgr     INFO Successfully loaded modules : AthenaServices
ApplicationMgr     INFO Application Manager Configured successfully
Py:Athena          INFO including file "AthenaCommon/Services.py"
Py:Athena          INFO including file "AthenaCommon/Atlas.UnixStandardJob.py"
ApplicationMgr     INFO Successfully loaded modules : IdDictDetDescrCnv
Py:Athena          INFO including file "my4famOptions.py"
ApplicationMgr     INFO Successfully loaded modules : DBDataModelAthenaPoolPoolCnv
ActiveStoreSvc    INFO Initializing ActiveStoreSvc - package version StoreGate-02-20-04
Py:Athena          INFO including file "AthenaCommon/runbatch.py"
ApplicationMgr     INFO Successfully loaded modules : GaudiSvc
ApplicationMgr     INFO Successfully loaded modules : StoreGate
ApplicationMgr     INFO Successfully loaded modules : AthenaPoolCnvSvc
ApplicationMgr     INFO Successfully loaded modules : AthenaSealSvc
ApplicationMgr     INFO Successfully loaded modules : AthenaRootStreamerSvc
ApplicationMgr     INFO Successfully loaded modules : DetDescrCnvSvc
ApplicationMgr     INFO Successfully loaded modules : PoolSvc
ApplicationMgr     INFO Successfully loaded modules : Pythia_i
StatusCodeSvc     INFO initialize
DetectorStore      INFO Initializing DetectorStore - package version StoreGate-02-20-04
EventPersistenc... INFO 'CnvServices':[ 'DetDescrCnvSvc' , 'AthenaPoolCnvSvc' ]
ProxyProviderSvc  INFO Initializing ProxyProviderSvc - package version SGComps-00-00-04
ClassIDSvc        INFO Initializing ClassIDSvc - package version CLIDComps-00-00-02
ClassIDSvc        INFO getRegistryEntries: read 61 CLIDRegistry entries for module ALL
ClassIDSvc        INFO ClassIDSvc Initialized successfully
StoreGateSvc      INFO Initializing StoreGateSvc - package version StoreGate-02-20-04
DetDescrCnvSvc    INFO initializing
DetDescrCnvSvc    INFO Found DetectorStore service
DetDescrCnvSvc    INFO filling proxies for detector managers
DetDescrCnvSvc    INFO filling address for InDetMgr with CLID 2512 and storage type 68
to detector store
DetDescrCnvSvc    INFO filling address for MuonMgr with CLID 4060 and storage type 68
to detector store
DetDescrCnvSvc    INFO filling address for CaloTTMgr with CLID 117659265 and storage
type 68 to detector store
DetDescrCnvSvc    INFO filling address for CaloMgr with CLID 4548337 and storage type
68 to detector store
DetDescrCnvSvc    INFO filling address for CaloIdManager with CLID 125856940 and
storage type 68 to detector store
DetDescrCnvSvc    INFO filling address for PixelID with CLID 2516 and storage type 68
to detector store
DetDescrCnvSvc    INFO filling address for SCT_ID with CLID 2517 and storage type 68
to detector store
DetDescrCnvSvc    INFO filling address for TRT_ID with CLID 2518 and storage type 68
to detector store
DetDescrCnvSvc    INFO filling address for SiliconID with CLID 129452393 and storage
type 68 to detector store
.
.
.
DetDescrCnvSvc    INFO filling address for CaloCell_ID with CLID 123500438 and storage
type 68 to detector store
DetDescrCnvSvc    INFO filling address for CaloDM_ID with CLID 167756483 and storage
type 68 to detector store

```



```

AthenaEventLoopMgr INFO Initializing AthenaEventLoopMgr - package version
AthenaServices-01-18-02-01
PartPropSvc        INFO PDT file(s):
PartPropSvc        INFO PDGTABLE.MeV
PartPropSvc        INFO Type: PDG
----- HepPDT Version 2.02.02 -----
found 258 particles
Pythia             INFO Standard Initialization: Single Interaction Mode
Pythia             INFO PYTHIA INITIALISING.
AtRndmGenSvc      INFO Initializing AtRndmGenSvc-package version AthenaServices-01-18-02-01
INITIALISING RANDOM NUMBER STREAMS.
Pythia             INFO Command is: pysubs msel 0
Pythia             INFO pysubs block msel item 0 value -1.1
Pythia             INFO Command is: pysubs msub 83 1
Pythia             INFO pysubs block msub item 83 value 1
Pythia             INFO Command is: pypars mstp 1 4
Pythia             INFO pypars block mstp item 1 value 4
Pythia             INFO Command is: pydat2 pmas 7 1 320.
Pythia             INFO pydat2 block pmas item 7 value 1
Pythia             INFO Command is: pydat2 pmas 8 1 320.
Pythia             INFO pydat2 block pmas item 8 value 1
Pythia             INFO Command is: pydat2 vckm 1 4 0.044
Pythia             INFO pydat2 block vckm item 1 value 4
Pythia             INFO Command is: pydat2 vckm 2 4 0.46
Pythia             INFO pydat2 block vckm item 2 value 4
Pythia             INFO Command is: pydat2 vckm 3 4 0.47
.
.
Pythia             INFO Command is: pydat3 mdme 207 1 1
Pythia             INFO pydat3 block mdme item 207 value 1
Pythia             INFO Command is: pydat3 mdme 208 1 0
Pythia             INFO pydat3 block mdme item 208 value 1
Pythia             INFO Command is: pydat3 mdme 209 1 0
Pythia             INFO pydat3 block mdme item 209 value 1
*****
**                  *.....*                               Welcome to the Lund Monte Carlo! **
**          *:::!!:::~::~::~::~*                           **
**          *:::~::~::~::~::~*                               PPP Y Y TTTT H H III A **
**          *:::~::~::~::~::~*                               P P Y Y T H H I A A **
**          *:::~::~::~::~::~*                               PPP Y T HHHH I AAAAA **
**          *:::~::~::~::~::~*                               P Y T H H I A A **
**          *:::~::~::~::~::~*                               P Y T H H III A A **
**          *:::~::~::~::~::~*                               **
**          !! *:::!!~::~::~::~*                               !! This is PYTHIA version 6.412 **
**          !! *! -><- * !!                               !! Last date of change: 23 Jul 2007 **
**          !! !! !! !!                               !! Now is 18 May 2008 at 2:56:41 **
**          !! !! !! !!                               !! Disclaimer: this program comes **
**          !! lh !! !!                               !! without any guarantees. Beware **
**          !! hh !! !!                               !! of errors and use common sense **
**          !! ll !! !!                               !! when interpreting results. **
**          !! !! !! !!                               !! Copyright T. Sjostrand (2007) **
**          !! !! !! !!                               **
**          !! !! !! !!                               **
** An archive of program versions and documentation is found on the web: **
** http://www.thep.lu.se/~torbjorn/Pythia.html **
** **
** Main author: Torbjorn Sjostrand; CERN/PH, CH-1211 Geneva, Switzerland, **
** and Department of Theoretical Physics, Lund University, Lund, Sweden; **
** phone: + 41 - 22 - 767 82 27; e-mail: torbjorn@thep.lu.se **
** Author: Stephen Mrenna; Computing Division, GDS Group, **
** Fermi National Accelerator Laboratory, MS 234, Batavia, IL 60510, USA; **
** phone: + 1 - 630 - 840 - 2556; e-mail: mrenna@fnal.gov **
** Author: Peter Skands; Theoretical Physics Department, **
** Fermi National Accelerator Laboratory, MS 106, Batavia, IL 60510, USA; **
** phone: + 1 - 630 - 840 - 2270; e-mail: skands@fnal.gov **
*****
*****

```

```

***** PYINIT: initialization of PYTHIA routines *****
==== PYTHIA WILL USE LHAPDF ====
*****
*      LHAPDF Version 5.3.0      *
*****
>>>>> PDF description: <<<<<<
CTEQ6L1 - LO with LO alpha_s
Reference:
J. Pumplin, D.R. Stump, J. Huston, H.L. Lai, P. Nadolsky,
W.K. Tung
hep-ph/0201195
>>>>>          <<<<<<
Parametrization: CTEQ6
=====
PDFset name
/afs/cern.ch/atlas/software/releases/13.0.30/sw/lcg/external/MCGenerators/lhapdf
with          1 members
==== initialized. =====
Strong coupling at Mz for PDF is: 0.12978

=====
I                                                    I
I          PYTHIA will be initialized for a p on p collider          I
I          at 14000.000 GeV center-of-mass energy                    I
I                                                    I
=====

***** PYMAXI: summary of differential cross-section maximum search *****

=====
I          I          I
I  ISUB  Subprocess name          I  Maximum value  I
I          I          I
=====
I          I          I
I   83   f + q -> f' + Q, massive  I   1.2057E-07  I
I   96   Semihard QCD 2 -> 2      I   1.0296E+04  I
I          I          I
=====

***** PYMULT: initialization of multiple interactions for MSTP(82) = 4 *****
      pT0 = 2.98 GeV gives sigma(parton-parton) = 5.38E+02 mb: accepted
***** PYMIGN: initialization of multiple interactions for MSTP(82) = 4 *****
      pT0 = 2.98 GeV gives sigma(parton-parton) = 2.32E+02 mb: accepted
***** PYINIT: initialization completed *****

          List of KF codes in program
          1      d          -1      dbar
          2      u          -2      ubar
          3      s          -3      sbar
          4      c          -4      cbar
          5      b          -5      bbar
          6      t          -6      tbar
          7      b'        -7      b'bar
          8      t'        -8      t'bar
          11     e-        -11     e+
          12     nu_e      -12     nu_ebar
          13     mu-       -13     mu+
          14     nu_mu     -14     nu_mubar
          15     tau-      -15     tau+
          16     nu_tau    -16     nu_taubar
          17     tau'-     -17     tau'+
          18     nu'_tau   -18     nu'_taubar
          .
          .
          4000012  nu*_e0    -4000012  nu*_ebar0
          5000039  Graviton*
          9900012  nu_Re
          9900014  nu_Rmu
          9900016  nu_Rtau

```

```

9900023    Z_R0
9900024    W_R+                -9900024    W_R-
9900041    H_L++                -9900041    H_L--
9900042    H_R++                -9900042    H_R--
9900110    rho_diff0
9900210    pi_diff+                -9900210    pi_diff-
9900220    omega_di
9900330    phi_diff
9900440    J/psi_di
9902110    n_diff0                -9902110    n_diffbar0
9902210    p_diff+                -9902210    p_diffbar-
9900441    cc~[1S08]
9900553    bb~[3S18]
9900551    bb~[1S08]
9910551    bb~[3P08]
DumpMC          INFO >>> DumpMC from Initialize
----- DumpMC World From initialize
ToolSvc.Streaml... INFO AthenaPoolOutputStreamTool initialize
ToolSvc.Streaml... INFO streamProperty ProcessingTag = Streaml
AthenaPoolCnvSvc INFO Initializing AthenaPoolCnvSvc-package version AthenaPoolCnvSvc-
00-11-44
AthenaSealSvc   INFO begin initialize() - loading dictionary fillers: size 4
PoolSvc         INFO No POOL WriteCatalog was specified--using POOL default.
XMLFileCatalog: level[Info] Connecting to the catalog
PoolXMLFileCatalog: level[Info] Xerces-c initialization Number 0
PoolSvc         INFO Set connectionsvc retry/timeout/IDLE timeout to
'ConnectionRetrialPeriod':30/ 'ConnectionRetrialTimeOut':300/ 'ConnectionTimeOut':5
seconds
PoolSvc         INFO Frontier compression level set to 5
  DbSession      Info      Open      DbSession
AthenaRootStrea... INFO POOL/ROOT class loader initialized
.
.
.
Streaml         INFO Found HelperTools = PublicToolHandleArray([])
Streaml         INFO Data output: pythia4fam.pool.root
HistogramPersis... INFO 'CnvServices':[ 'HbookHistSvc' , 'RootHistSvc' ]
HistogramPersis...WARNING Histograms saving not required.
AthenaEventLoopMgrWARNING Histograms saving not required.
ApplicationMgr  INFO Application Manager Initialized successfully
AthenaEventLoopMgr INFO ===>>> start of run 0 <<<===
AthenaEventLoopMgr INFO ===>>> start of event 1 <<<===
HistorySvc     INFO Registered 3 Algorithms
HistorySvc     INFO Registered 2 AlgTools
HistorySvc     INFO Registered 25 Services
DumpMC        INFO >>> DumpMC from execute
----- DumpMC World From execute
Next event in the bag
PYTHIA        + 83

```

```

GenEvent: #1 ID=1000083 SignalProcessGenVertex Barcode: 0
Current Memory Usage: 1 events, 56 vertices, 140 particles.
Entries this event: 56 vertices, 140 particles.
RndmState(2)=4789899 989240512
Wgts(3)=1 1 1
EventScale -1 [energy]      alphaQCD=-1      alphaQED=-1

```

		GenParticle Legend					
Barcode	PDG ID	(Px,	Py,	Pz,	E)	Stat	DecayVtx
GenVertex: -1 ID: 0 (X,cT):0							
I: 1	1	2212	+0.00e+00,	+0.00e+00,	+7.00e+06,	+7.00e+06	3 -1
O: 3	3	21	-1.88e+03,	+5.77e+02,	-2.54e+03,	+3.21e+03	3 -3
	10	2	+8.17e+02,	-1.60e+02,	+5.44e+04,	+5.44e+04	2 -8
	14	2101	+1.07e+03,	-4.18e+02,	+6.95e+06,	+6.95e+06	2 -12
GenVertex: -2 ID: 0 (X,cT):0							
I: 1	2	2212	+0.00e+00,	+0.00e+00,	-7.00e+06,	+7.00e+06	3 -2
O: 4	4	-1	-1.58e+03,	+6.81e+02,	-1.34e+06,	+1.34e+06	3 -4
	12	1	+1.58e+03,	-6.81e+02,	-1.49e+06,	+1.49e+06	2 -10
	15	1	-2.48e+02,	+6.37e+02,	-1.80e+05,	+1.80e+05	2 -14
	16	2203	+2.48e+02,	-6.37e+02,	-3.99e+06,	+3.99e+06	2 -14

```

.
.
Vertex:  -213 ID:    0 (X,cT)=-2.13e+00,-3.36e+00,+9.16e+01,+9.17e+01
I: 1      668      111 -1.01e+03,-1.01e+03,+3.36e+04,+3.36e+04  2      -213
O: 2      672      22  -3.59e+02,-3.18e+02,+1.26e+04,+1.26e+04  1      (nil)
        673      22  -6.50e+02,-6.93e+02,+2.09e+04,+2.10e+04  1      (nil)
Vertex:  -214 ID:    0 (X,cT)=-2.13e+00,-3.36e+00,+9.16e+01,+9.16e+01
I: 1      669      221 -2.28e+03,-2.46e+03,+7.93e+04,+7.94e+04  2      -214
O: 2      675      22  -3.35e+02,-1.75e+02,+1.20e+04,+1.20e+04  1      (nil)
        674      22  -1.94e+03,-2.28e+03,+6.73e+04,+6.74e+04  1      (nil)
-----
AtRndmGenSvc      INFO  Stream = PYTHIA, Seed1 = 1187802338, Seed2 = 1748997362
AtRndmGenSvc      INFO  Stream = PYTHIA_INIT, Seed1 = 820021, Seed2 = 2347532
AthenaEventLoopMgr  INFO  ===>>> end of event 744 <<<===
AthenaEventLoopMgr  INFO  ===>>> start of event 745 <<<===

```

```

(PYPTIS:) Weight 1.2835E+00 above unity
DumpMC          INFO >>> DumpMC from execute
----- DumpMC World From execute
Next event in the bag
PYTHIA          + 83

```

```

-----
GenEvent: #745 ID=1000083 SignalProcessGenVertex Barcode: 0
Current Memory Usage: 1 events, 95 vertices, 248 particles.
Entries this event: 95 vertices, 248 particles.
RndmState(2)=1187802338 1748997362
Wgts(3)=1 1 1
EventScale -1 [energy]          alphaQCD=-1  alphaQED=-1

```

		GenParticle Legend					
Barcode	PDG ID	(Px,	Py,	Pz,	E)	Stat	DecayVtx

GenVertex:	-1	ID:	0 (X,cT):0				
I: 1	1	2212	+0.00e+00,	+0.00e+00,	+7.00e+06,	+7.00e+06	3 -1
O: 9	12	21	-2.18e+03,	+1.17e+03,	-9.24e+03,	+9.57e+03	2 -10
	3	21	+3.93e+02,	-4.92e+02,	+1.51e+05,	+1.51e+05	3 -3
	13	21	+3.15e+02,	+9.28e+02,	-4.40e+03,	+4.51e+03	2 -10
	14	21	-3.26e+02,	-4.64e+02,	-7.99e+02,	+9.80e+02	2 -10
	15	21	+2.21e+01,	-2.93e+03,	-1.47e+04,	+1.50e+04	2 -10

```

.
.

```

```

Vertex:  -108 ID:    0 (X,cT)=+2.17e+00,+1.74e+00,-1.31e-01,+2.79e+00
I: 1      304      313 +2.02e+04,+1.69e+04,-7.35e+02,+2.64e+04  2      -108
O: 2      309      321 +1.06e+04,+9.17e+03,-4.19e+02,+1.40e+04  1      (nil)
        310      -211 +9.67e+03,+7.71e+03,-3.17e+02,+1.24e+04  1      (nil)

```

```

-----
AtRndmGenSvc      INFO  Stream = PYTHIA, Seed1 = 907575434, Seed2 = 14296221
AtRndmGenSvc      INFO  Stream = PYTHIA_INIT, Seed1 = 820021, Seed2 = 2347532
AthenaEventLoopMgr  INFO  ===>>> end of event 999 <<<===
AthenaEventLoopMgr  INFO  ===>>> start of event 1000 <<<===
DumpMC          INFO >>> DumpMC from execute
----- DumpMC World From execute
Next event in the bag
PYTHIA          + 83

```

```

-----
GenEvent: #1000 ID=1000083 SignalProcessGenVertex Barcode: 0
Current Memory Usage: 1 events, 369 vertices, 1136 particles.
Entries this event: 369 vertices, 1136 particles.
RndmState(2)=907575434 14296221
Wgts(3)=1 1 1
EventScale -1 [energy]          alphaQCD=-1  alphaQED=-1

```

		GenParticle Legend					
Barcode	PDG ID	(Px,	Py,	Pz,	E)	Stat	DecayVtx

GenVertex:	-1	ID:	0 (X,cT):0				
I: 1	1	2212	+0.00e+00,	+0.00e+00,	+7.00e+06,	+7.00e+06	3 -1
O:81	156	21	-5.48e+02,	+9.21e+02,	-4.11e+03,	+4.24e+03	2 -34
	103	21	+1.95e+02,	-5.86e+01,	+2.30e+04,	+2.30e+04	2 -22
	3	21	+1.90e+03,	-9.21e+02,	+1.51e+05,	+1.51e+05	3 -3
	101	2	+8.50e+02,	-3.41e+02,	+6.21e+05,	+6.21e+05	2 -22
	104	21	+2.61e+02,	+3.22e+02,	+7.38e+03,	+7.39e+03	2 -22

```

.
.
Vertex:   -368 ID:    0 (X,CT)=-3.92e-02,-7.91e-02,-1.46e-01,+1.71e-01
I: 1     1117      111 -5.06e+02,-1.53e+03,-2.23e+03,+2.76e+03  2   -368
O: 2     1133      22 -3.68e+02,-1.06e+03,-1.65e+03,+1.99e+03  1   (nil)
        1134      22 -1.38e+02,-4.74e+02,-5.84e+02,+7.64e+02  1   (nil)
Vertex:   -369 ID:    0 (X,CT)=+2.74e+00,+5.87e+00,-1.64e+01,+1.77e+01
I: 1     1126      111 +2.48e+03,+5.46e+03,-1.38e+04,+1.51e+04  2   -369
O: 2     1135      22 +1.15e+03,+2.61e+03,-6.71e+03,+7.29e+03  1   (nil)
        1136      22 +1.33e+03,+2.85e+03,-7.10e+03,+7.77e+03  1   (nil)
-----
AtRndmGenSvc      INFO Stream = PYTHIA, Seed1 = 1422670727, Seed2 = 145930269
AtRndmGenSvc      INFO Stream = PYTHIA_INIT, Seed1 = 820021, Seed2 = 2347532
AthenaEventLoopMgr  INFO ==>>> end of event 1000 <<====
HistorySvc        INFO Service finalised successfully
Pythia            INFO PYTHIA Ending.
Pythia            INFO Call PYSTAT at endRun with level 1
1***** PYSTAT:  Statistics on Number of Events and Cross-sections *****

=====
I          Subprocess          I          Number of points          I          Sigma          I
I-----I-----I-----I-----I-----I-----I-----I-----I-----I
I N:o Type          I          Generated          Tried I          I
=====
I  0 All included subprocesses  I          7000          69789 I  3.304E-08 I
I  83 f + q -> f' + Q, massive  I          7000          69789 I  3.304E-08 I
=====
***** Total number of errors, excluding junctions =          0 *****
***** Total number of errors, including junctions =          0 *****
***** Total number of warnings =          3 *****
***** Fraction of events that fail fragmentation cuts = 0.00000 *****

MetaData: cross-section (nb)= 0.00330384
DumpMC            INFO >>> DumpMC from finalize
----- DumpMC World From finalize
Stream1           INFO Records written: 1000
PoolSvc           INFO finalize() in PoolSvc
pythia4fam.pool.root  Info Database being retired...
Domain[ROOT_All]Info->Deaccess DbDatabase CREATE[ROOT_All]64762243-380D-DD11-9829-
000423D2B9E8
Domain[ROOT_All]Info > Deaccess DbDomain UPDATE [ROOT_All]
AthenaSealSvc    INFO finalize() in AthenaSealSvc
AtRndmGenSvc     INFO FINALISING
PartPropSvc      INFO Service finalised successfully
StoreGateSvc     INFO Finalizing StoreGateSvc - package version StoreGate-02-20-04
DetectorStore    INFO Finalizing DetectorStore - package version StoreGate-02-20-04
ToolSvc.finalize() INFO Removing all tools created by ToolSvc
ToolSvc.Stream1... INFO AthenaPoolOutputStreamTool finalize
*****Chrono*****
*****Chrono*****
*****Chrono***** INFO The Final CPU consumption ( Chrono ) Table (ordered)
*****
AthenaSealSvc:... INFO Time User: Tot=168 [ms] Ave/Min/Max=56(+/- 27)/ 22/ 88 [ms]
#= 3
AthenaSealSvc:... INFO Time User: Tot=339 [ms] Ave/Min/Max=67.8(+/- 84.9)/ 1/ 225 [ms]
#= 5
AthenaSealSvc:... INFO Time User: Tot=437 [ms]
#= 1
ChronoStatSvc    INFO Time User: Tot=60.4 [s]
*****
ChronoStatSvc.f... INFO Service finalized successfully
ApplicationMgr    INFO Application Manager Finalized successfully
ApplicationMgr    INFO Application Manager Terminated successfully
Py:Athena        INFO leaving with code 0: "successful run"

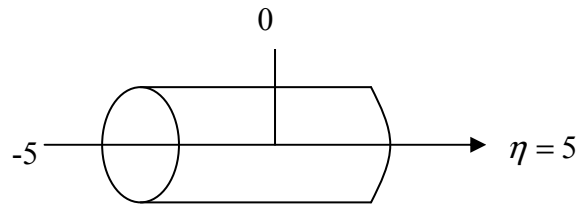
```

4.6. Listenin Yorumu:

Yukarıdaki çok uzun listede (7000 olay için 44000 sayfalık bilgi içerilmektedir.) üretilen parçacıklara, onların geometrisine, kinematiğine, fiziğine, çarpışma ortamında davranışlarına ait bütün bilgiler içerilmektedir. Bu bilgiler, yazılan program doğrultusunda hangi parçacıkların üretileceği ve dedektör katmanları içerisinde bu parçacıkların nasıl davranacağı bilgilerinin verilmesiyle başlar. Üretilen parçacıklar örneğin elektronlar elektromagnetik kalorimetrede iyonlaştırılarak gözlenebilir, bu duruma göre kalorimetre bilgileri içerilmektedir, yine kuarklar hadronik kalorimetrenin sürüklenme tüpleri içerisinde çarpışmadan sonra spark'ın oluşması için geçen zaman ve nerede oluştuğunun x, y bileşenlerinin ölçülmesiyle konum ve momentum bilgileri elde edilebileceğini ATHENA programı çalışmasıyla bilgileri vermiştir. Listenin devamında, yazılan programda hangi parçacıkların çarpışmasıyla hangi kanallara bakılacağı ve bu kanalların açılması ile PYTHIA komutlarının alınmasıyla devam eder. İstenen çarpışma hangi kütle merkezi enerjisinde olacağı ve başlangıç tesir kesitleri tablo halinde verilmektedir. Daha sonra üretilen parçacıkların bu bunlara ait bilgilerin içerildiği kod bilgileri, listenin devamında parçacıkların çarpışmalarının ifade edilebildiği ve fiziğinin içerildiği köşe faktörü bilgileri verilmektedir. Parçacıkların çarpışmaları harekete sahip olduğundan lagranjiyen ile ifade edilir. Lagranjiyen yazılırken iki parçacığın çarpıştığı düşünülen köşeye ait özellik geçiş bilgileri (renk, çeşni, momentum aktarımı, enerji aktarımı) gibi bilgiler verilmektedir. Ayrıca köşe faktörlerinde, dedektörün yapısına ait geometrik bilgiler de yer almaktadır. Momentum bileşenleri, jetlerin gözlenebilmesi için gereken konilerin büyüklüğü, kayıp enerji bilgileri, izolasyon bilgileri, kriter ve verim gibi dedektörde istenilen parçacığı en iyi şekilde elde edilmesini sağlayan optimizasyondur bu bilgiler. Bu bilgilerden sonra tekrar fiziksel sonuç olan çalışılan ve sonuçta üretilen parçacığa ait üretim tesir kesitinin çıktı değeri liste olarak verilmektedir. ATHENA 13.X versiyonu ile dördüncü aile fermiyonlarının üretim tesir kesitleri $\sigma=3.304 \times 10^{-8}$ mb olarak bulunmuştur. LHC deneyi çalışmaya başladığında ilk üç yıl düşük luminositi $L= 10 \text{ fb}^{-1}$ ve sonraki bir yıl yüksek luminositi olan $L=100 \text{ fb}^{-1}$ değeri ile çalışacaktır. Yılda dördüncü standart model fermiyon ailesi için olay üretilme durumları modelleme sonunda $R=L \times \sigma$ denkleminde düşük aydınlık için $R= 10^{-8} \times 10^{12} \times 10=10^5$ yıldaki olay sayısı ve yüksek luminositi için $R= 10^{-8} \times 10^{12} \times 100=10^6$ yılda elde edilecek olay sayısı olarak elde edilir. Yılda düşük luminositi ile yüzbin yüksek luminositi ile milyon miktarda dördüncü fermiyon ailesinin LHC-ATLAS Deneyinde elde edilebilecek olması, bu parçacığın varlığından öte yüksek bir miktarda elde edilebileceğini gösterir. Bu durum projemizin amacına uygunluğunu ve dördüncü ailenin varlığını ispatlar.

4.7. Programa Etki Eden Kinematik ve Geometrik Kriterler

Programda kullanılan kuarkların dedektörde gözlenebilen izlerine jet denir. Dedektörün merkezinde izsürücü dedektöründe ve bunu takip eden kalorimetrede bu jetler gözlenebilir. Jetlerin enerjilerinin büyüklüğüne göre silindir şeklindeki dedektörde bulunuş yerleride değişir. Yüksek enerjili jetler gelen demet doğrultusunda büyük bir enerjiyle geri teperler. Jetlerin silindirik dedektör içerisindeki dağılımları $\eta(\eta)$ fonksiyonu ile verilir. η 'nın (η) dedektör içindeki sınırları şekil 4.4'deki gibidir.



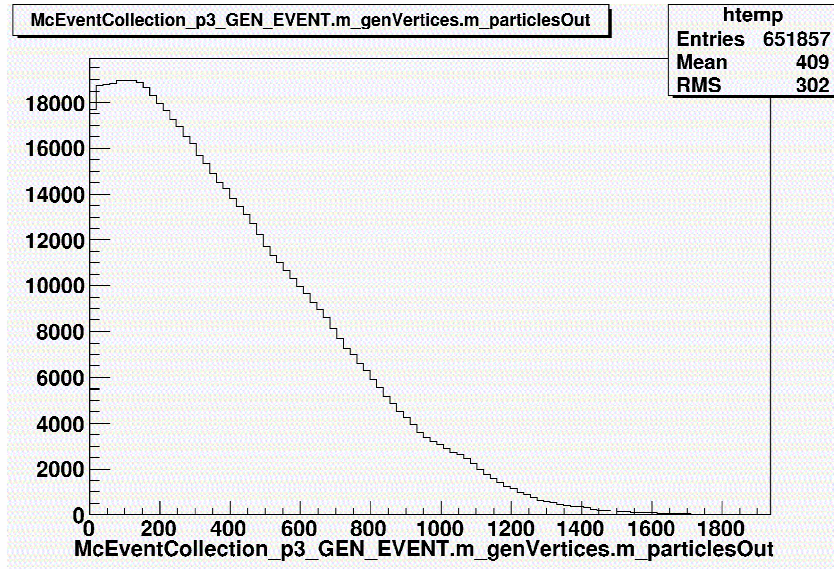
Şekil 4.4. Dedektör içindeki eta sınırları ($-5 \leq \eta \leq 5$)

Detektörün z-ekseni üzerinde kapladığı genişlik değerini η ile gösteririz. η 'nın sayısal değeri -5 ile 5 arasındadır ve aşağıdaki bağıntı ile verilir:

$$\eta = -\log\left(\tan\frac{\theta}{2}\right) \quad (4.1)$$

Enine momentum, p_T , LHC çarpışma eksenine dik olan eksenlerin oluşturduğu momentumdur.

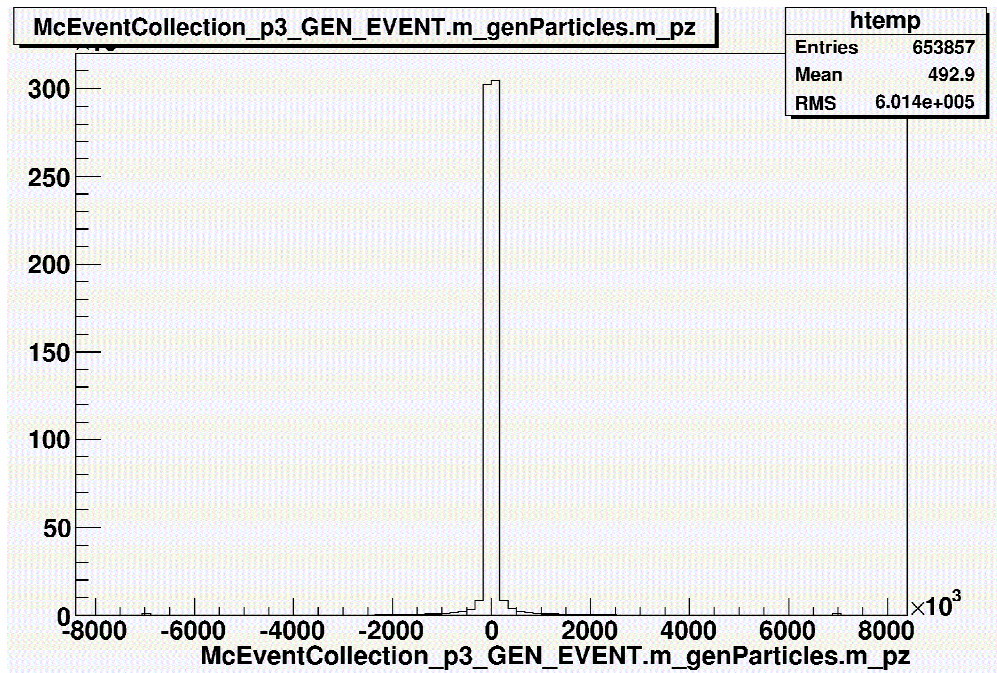
İç dedektörde parçacıkların momentumu ölçülür ve izleri gözlenir. İç dedektörde temel parçacıkolan elektronlarla fotonlar birbirinden enerji farkıyla ayrılırlar. Var olan birçok verteks içinde ilk verteksin tanımlanması için detektörün silindiriksel kap kısmında $|\eta| \leq 1$ 'dir ve detekte edici elemanların hepsi silindirik bir yapı içinde sıralanır. Şekil 4.5'de vertexlere göre parçacıkların dağılımı görülmektedir. Parçacıkların momentum dağılımı, iz uzunlukları boyunca izin yeniden yapılması ile elde edilir. Bu, hem tek dedeksiyon elemanlarının kararlılığını sağlamayı hem de tek ölçümleri doğru olarak izlerler ve birleştirmeyi gerektirir. Jetlerin yeniden oluşturulması karmaşık kuarkların araştırılması için önemlidir. Bu kuark dizilimleri yüksek enerjilerde hadronik kalorimetre cevabının lineerliğini sınırlar. Parçacıkların özelliklerinin en iyi şekilde elde edilmesi için eta ve momentum kriterlerinden başka dik enerji ölçümü, kayıp enerjinin bulunması ve ϕ ölçümleride önemlidir.



Şekil 4.5. Monte Carlo ile Üretilen Parçacıkların Etkileşme Vertekslerinde davranışları

Kayıp dik enerjinin ölçümü, etkileşmeye girmeden detektörden kaçan parçacıkları saptamak için bir yol teşkil eder. Bunlar ya nötrinolar ya da kararlı süpersimetrik parçacıklar olabilir. Kayıp dik enerjiyi tanımlamak için kalorimetre iyi bir hermitikliğe ihtiyaç duyar. Bu da z-yönündeki yüksek enerjili parçacıklar için rapidity kapsamı $|\eta| = 5$ e kadar varacağı anlamına gelir. Şekil 4.6'de enine momentumu oluşturan z-yönündeki parçacıkların dağılımı görülmektedir. Nötrinolar ve müonlar hariç diğer bozunmamış parçacıkların dik enerjileri (E_T) bütün kalorimetreyi kapsayan ($|\eta| < 5$) $\eta \times \Phi$ koordinatlarında izole edilmiş bir son durum parçacığının kalorimetre hücrelerinde toplanır. Eşik değerinin üstünde ($p_t < 0.5 \text{ GeV}$) p_t ve yüklü parçacıkların Φ yönündeki solenoidal 2 Teslalık manyetik alanı programda parametrize edilir. Eşik değerden $E_T^{\text{init}} > 1.5 \text{ GeV}$ daha büyük dik enerji değeri ile bütün kalorimetre hücreleri E_T 'yi azaltmak için $\Delta R = \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}$, ($\Delta R = 0.4$) olan bir konide toplanır. Jetler içinse yine 0.4 yarıçaplı silindirik kap için 0.7 yarıçaplı ileri kısım için koniler seçilir. $0.7 \geq \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}$ olmalıdır. Böyle bir koninin içindeki atom grupları kabul edebilir. Jetler için eta alanı 5 (yani detektör boyunca tüm alan) alıyoruz. $|\eta| = 5$ ise 0-5 arası tüm jetler alınır. $pp \rightarrow \bar{u}_4 u_4 x \rightarrow bW^+ bW^- x$, b-jetleri ve W'lerin hadronik ($W \rightarrow jj$) ve leptonik ($W \rightarrow \tau \nu$) proseslerinde ki leptonik ve hadronik W'leri yeniden oluşturabilmek için leptonları, jetleri istenilen kinematikte (açısal olarak merkezi bölgeye sınırlanmış p_t gözönüne alınabilir) değerde maksimum enerjili jetleri ve leptonları sıralamak için uygulanan kriterler:

- $\eta \leq 2.5$ merkezi jet ve leptonlar için
- $\eta \geq 2.5$ ileri, geri jetler dedektörün bu kısmında bulunur, yüksek enerjiye sahiptirler.
- $E_T^{l,jets} \geq 10 \text{ GeV}$
- $P_T^{l,jets} \geq 6 \text{ GeV}$
- $0.4 \leq \Delta R \leq 0.7$



Şekil 4.6. Üretilen parçacıkların dedektöre göre z-yönündeki momentum dağılımı

Bu uygulanan kriterlerle optimum jet ve leptonlar elde edilir. Bunların yeniden oluşturulmasıyla leptonik ve hadronik W' ler oluşturulur ve b-jetlerin etiketlenmesiyle dördüncü aile kuarklarının varlığına ulaşılır.

5. Bitwise Operatörler

Bu operatörler tamsayı, karakter gibi değişkenlerin sayısal değerlerinin üzerinde değil de doğrudan 0 ve 1 lerden oluşan bitlerini değiştirmek, sınamak veya ötelemek için kullanılır.

En yaygın kullanılan bitwise operatörleri;

& And Operatörü

| Or Operatörü

^ XOR Operatörü

~ The Ones Complement(Bir'e Tümlleme)

<< The Left Shift Operator(Sola Öteleme)

>> The Right Shift Operator(Sağa Öteleme)

5.1. Bitsel Değil Operatörü:

Diğer tüm tek operand alan operatörler gibi operatör öncelik tablosunun ikinci seviyesindedir. Yan etkisi yoktur. (side effect) Operandı bir nesne ise, bu nesnenin bellekteki değerini değiştirmez. Ürettiği değer, operandı olan tamsayının bitleri üzerinde 1'e tümlleme işlemi yapılarak elde edilen değerdir. Yani operandı olan ifade değerinin 1 olan bitleri 0, 0 olan bitleri 1 yapılarak değer üretilir.

5.2. Bitsel Kaydırma Operatörleri (bitwise shift operators)

İki adet bitsel kaydırma operatörü vardır:

Bitsel sağa kaydırma operatörü >> (bitwise right shift)

Bitsel sola kaydırma operatörü << (bitwise left shift)

Her iki operatör de operatör öncelik tablosunun 5. seviyesinde bulunmaktadır. Dolayısıyla bu operatörlerin önceliği tüm aritmetik operatörlerden daha aşağıda fakat karşılaştırma operatörlerinden daha yukarıdadır. Bitsel kaydırma operatörleri binary infix operatörlerdir.

Bitsel sola kaydırma operatörünün ürettiği değer soldaki operandı olan tamsayının sağdaki operandı olan tamsayı kadar sola kaydırılmış biçimindedir. Menzil dışına çıkan bitler için sayının sağından 0 biti ile besleme yapılır.

5.3. Bitsel Ve Operatörü (bitwise and)

Operatör öncelik tablosunun 8. seviyesindedir. Öncelik yönü soldan sağdır. Yan etkisi yoktur, operandları nesne gösteren bir ifade ise bellekteki değerlerini değiştirmez. Değer üretmek için operandı olan tamsayıların karşılıklı bitlerini Ve işlemine tabi tutar. Ve işlemine ait doğruluk tablosu aşağıda tekrar verilmektedir:

x
y
x & y

1
1
1
1
0
0
0
1
0
0
0
0

Bitsel ve operatörünün ürettiği değer operandlarının karşılıklı bitlerinin ve işlemine sokulmasıyla elde edilen değerdir.

5.4. Bitsel Özel Veya Operatörü (bitwise exor)

Operatör öncelik tablosunun 9. seviyesindedir. Öncelik yönü soldan sağdır. Yan etkisi yoktur, operandları nesne gösteren bir ifadeyse bellekteki değerlerini değiştirmez. Değer üretmek için operandı olan tamsayıların karşılıklı bitlerini özel veya (exclusive or) işlemine tabi

tutar. Bitset özel veya işlemine ilişkin doğruluk tablosu aşağıda verilmektedir:

x
y
$x \wedge y$
1
1
0
1
0
1
0
1
0
1
1
0
0
0

Yukarıdaki tablo şu şekilde açıklanabilir. Operandlardan ikisi de aynı değere sahip ise üretilen değer 0 olacak, operandlardan biri diğerinden farklı ise üretilen değer 1 olacaktır.

5.5. Bitset Veya Operatörü (bitwise or operator)

Operatör öncelik tablosunun 10. seviyesindedir. Öncelik yönü soldan sağdır. Yan etkisi yoktur, operandları nesne gösteren bir ifade ise bellekteki değerlerini değiştirmez. Değer üretmek için operandı olan tamsayıların karşılıklı bitlerini veya işlemine tabi tutar. Veya işlemine ait doğruluk tablosu aşağıda verilmektedir:

x
y
$x y$
1
1
1
1
1
0

1
0
1
1
0
0
0

5.6. Bitset Operatörlere İlişkin İşlemler Atama Operatörleri

Bitset değil operatörünün dışında tüm bitset operatörlere ilişkin işlemler atama operatörleri vardır. Daha önce de bahsedildiği gibi bitset operatörlerin yan etkileri (side effect) yoktur. Bitset operatörler operandları olan nesnelerin bellekteki değerlerini değiştirmezler. Eğer operandları olan nesnelerin değerlerinin değiştirilmesi isteniyorsa bu durumda işlemler atama operatörleri kullanılmaktadır.

$x = x \ll y$ yerine $x \lll y$

$x = x \gg y$ yerine $x \ggg y$

$x = x \& y$ yerine $x \&= y$

$x = x \wedge y$ yerine $x \wedge= y$

$x = x | y$ yerine $x |= y$ kullanılabilir.

5.7. Bir Sayının Belirli Bir Bitinin 1 Yapılması (birleşmesi) (set a bit / turn a bit on)

Buna sayının belirli bir bitinin set edilmesi de denebilir. Bir sayının belirli bir bitini set etmek için, söz konusu sayı, ilgili biti 1 olan ve diğer bitleri 0 olan bir sayıyla veya işlemine tabi tutulmalıdır. Çünkü bitset veya işleminde 1 yutan eleman 0 ise etkisiz elemandır.

5.8. Bir Sayının Belirli Bir Bitinin 0 Yapılması (sıfırlanması) (clear a bit / turn the bit off)

Buna sayının belirli bir bitinin temizlenmesi de denebilir. Bir sayının belirli bir bitini sıfırlamak için, söz konusu sayı, ilgili biti 0 olan ve diğer bitleri 1 olan bir sayıyla bitset ve

işlemine tabi tutulmalıdır. Çünkü bitisel ve işleminde 0 yutan eleman 1 ise etkisiz elemandır. Bir bitin sıfırlanması için kullanılacak bu sayıya maske (mask) denir.

5.9. Bir Sayının Belirli Bir Bitinin Değerinin Test Edilmesi (0 mı 1 mi)

Bir sayının belirli bir bitinin 0 mı 1 mi olduğunun öğrenilmesi için, söz konusu sayı, **if** deyiminin koşul ifadesi olarak (**if** parantezi içinde) ilgili biti 1 olan ve diğer bitleri 0 olan bir sayıyla bitisel ve işlemine tabi tutulmalıdır. Çünkü bitisel ve işleminde 0 yutan eleman 1 ise etkisiz elemandır. programın akışı **if** deyiminin doğru kısmına giderse, ilgili bitin 1, yanlış kısmına gider ise ilgili bitin 0 olduğu sonucu çıkarılacaktır.

x bir tamsayı olmak üzere, bir sayının herhangi bir bitinin 1 ya da 0 olduğunu anlamak için aşağıdaki ifadeyi yazabiliriz.

```
if (x & 1 << k)
```

```
/* k. bit 1 */
```

```
else
```

```
/* k. bit 0 */
```

5.10. Bir Sayının Belirli Bir Bitini Ters Çevirmek (toggle)

Bazı uygulamalarda bir sayının belirli bir bitinin değerinin değiştirilmesi istenebilir. Yani söz konusu bir 1 ise 0 yapılacak, söz konusu bit 0 ise 1 yapılacaktır. Bu amaçla bitisel özel veya operatörü kullanılır. Bitisel özel veya operatöründe 0 biti etkisiz elemandır. Bir sayının k.bitinin değerini değiştirmek için, sayı, k.bit 1 diğer bitleri 0 olan bir sayı ile bitisel özel veya işlemine tabi tutulur.

x bir tamsayı olmak üzere, bir sayının herhangi bir bitinin değerini değiştirmek için aşağıdaki ifadeyi yazabiliriz.

```
x ^= 1 << k;
```

Bitwise operatörünün kullanılmasıyla ilgili olarak yapılan ilk çalışma fizik visual C programı aşağıdaki şekildedir:

```
#include <iostream>
#include <math.h>

using namespace std;

void integer_to_binary (int integer, int *rresult, int *rlength)
{
    int result;
    int multiplier;
    int length;
    result = 0;
    multiplier = 1;
    length = 1;

    while(pow(2, length) < integer)
    {
        length++;
    }

    while((integer) > 1)
    {
        result += (multiplier*(integer%2));
        multiplier *= 10;
        integer /= 2;
    }

    if(integer == 1)
    {
        result += multiplier*1;
    }

    *rresult = result;
    *rlength = length;
}

void stuff_to_short_int
{
    int temp_what = *what;
    int temp_target = *target;
    int temp_multiplier = 10;
    int temp_multiplier_2 = pow(10, *what_length - 1);

    for(int i=0; i<*what_length; i++)
    {
        short int temp_temp = temp_what / temp_multiplier_2;
        temp_target = temp_target << 1;
        temp_target = (temp_target | temp_temp);
        if(temp_temp > 0)
        {
            temp_what -= temp_multiplier_2;
        }
    }
}
```

```

    }
    temp_multiplier_2 /= 10;

}

temp_target = temp_target << (pad_length - *what_length);

*target = temp_target;

}

int main()
{
    int day;
    int month;
    int year;

    cout << "Lütfen bir gün giriniz [01 - 31]:";
    cin >> day;

    cout << "Lütfen bir ay giriniz [01 - 12]:";
    cin >> month;

    cout << "Lütfen bir yıl giriniz [00-99]:";
    cin >> year;

    cout << "Girdiğiniz tarih: " << day << " " << month << " " <<
year << endl;

    int binday, bindaylength = 0;
    int binmon, binmonlength = 0;
    int binyea, binyealength = 0;

    integer_to_binary(day, &binday, &bindaylength);
    integer_to_binary(month, &binmon, &binmonlength);
    integer_to_binary(year, &binyea, &binyealength);

    short int bincode;

    stuff_to_short_int(&binday, &bindaylength, 5, &bincode);
    stuff_to_short_int(&binmon, &binmonlength, 4, &bincode);
    stuff_to_short_int(&binyea, &binyealength, 7, &bincode);

    cout << bincode << endl;

    return 0;
}

```

Yapılan bu C++ çalışması lxxplus üzerinde derlenip, çalıştırılmıştır. Sonuç olarak bitler üzerinde sıkıştırma işlemi yapılabilmektedir.

Yapacağımız çalışmanın kaba algoritması aşağıdaki şekildedir:

1. Datayı oluşturmak için 10 000 sayı için 0 ile 1 arasında random sayı seçerek input data oluşturulmalıdır.
2. Bu random sayıları fprintf ile dosyaya kaydedilemelidir.
3. 10.000 sayının grafiği çizdirilmelidir.
4. Algoritma belirlenmelidir.
5. Bu data 32bit ten 10 bite (yani 32 den küçük yarisi veya dahaazina) paketlenmelidir.

Belirlenen işlem adımlarında 10.000 tane 0 ile 1 arasında random sayı üreten ve bu elde edilen sayıları bir dosya içerisine kaydeden program kodları:

```
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
    FILE *f;
    f= fopen ("deneme.txt", "w");
    int seed;
    double r;
    long int M;
    int i;
    seed = 100000;
    for(i=1; i<=10000; ++i)
    {
        r = ( (double)rand() / ((double) (RAND_MAX)+(double) (1)) )
        fprintf(f, "%3d %5f\n", i, r);
    }
    fclose(f);
    system("PAUSE");
    return 0;
}
```

Kodlardan anlaşılacağı üzere elde edilen veriler deneme.txt adlı dosyaya kaydedirilmişdir. Bu elde edilen deneme.txt dosyasının özet çıktı listesi ise,

1 0.966919

2 0.768738

```

3 0.772949
4 0.480042
5 0.853302
6 0.788574
7 0.701935
8 0.651276
9 0.693573
10 0.520477

```

...

...

...

```

9993 0.395538
9994 0.201965
9995 0.772858
9996 0.210419
9997 0.585419
9998 0.482788
9999 0.967072

```

10000 0.753906 şeklindedir.

Bu yapılan çalışmanın lxplus üzerinden ROOT grafiklerini almak için, hesabımız altında çalışma dizini açtıktan sonra derleme işlemleri yapılır ve program çalıştırılır. Fakat bu adımlardan önce lxplus üzerinde yapılması gereken adımlar:

```

>bash
>cd cmthome
>source setup.sh -tag=14.2.0
>cd ../gmz
>root
>int n=100;Double_t x[n].y[n]: char str[20]:
>fstream file_op("deneme.txt".ios::in) : int cnt = 0 :
>while ( file_op>>str) {
>count<<str<<cnt<<endl:x[cnt] = cnt* 0.1: y[cnt] =
atof(str):cnt++ }
>file_op.close() : TGraph*gr = new TGraph (n,x,y):
>TCanvas*c1 = new TCanvas ("c1"."Graph Draw
Options".200.10.600.400):

```

ROOT kodlarını lxplus üzerine yazarken bu adımda cygwin shell ortamına grafik sonuçlarını almak için ihtiyaç duyuldu.

5.11. Cygwin Yapısı

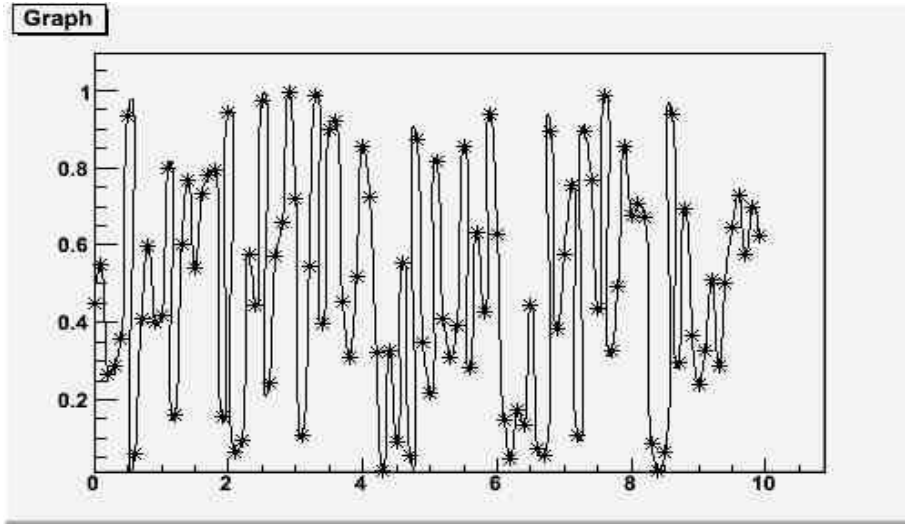
Cygwin windows için hazırlanmış Linux' a benzeyen bir yapıdır. Linux Apilerinin görevlerini yerine getiren bir Dll(cygwin1.dll) dir şeklinde de tanımlayabiliriz. Windows CE haricindeki bütün beta olmayan windows sürümleri ile çalışır. İşlemci yapısı olarak ix86 ve 32 bit olmalıdır. Ancak, linux programlarının windows altında çalışması için kesin çözüm olmayabilir. Çünkü Windows altındaki programları tekrar derlemek gerekebilir.

Cygwin Windows programlarınızın Unix ® sinyal ve fonksiyonlarından haberdar olması onlarla iletişim kurması için yapılmamıştır. Bunun için Source Code den derleme yapılmalıdır.

Setup ı açıp ileri ileri dediğinizde size paket seçimini soracaktır. Burada çalıştırmak istenilen programın durumuna göre paket seçimi yapılabilir. Direkt ileri seçilirse standart paketler yüklenmeye başlanacaktır.

<http://sourceforge.net/projects/xming> adresi ile kurulum işlemi gerçekleştirildi.

Lxplus üzerinde tekrar yukarıdaki işlemler yapıldıktan sonra cygwin programı çalıştırılır ve >startx komutu yardımıyla yeni bir pencere açılır. ROOT grafik sonuçları bu şekilde elde edilmiş olur. Yazılan ROOT histogram kodu veriler üzerinde çalışıp, deneme.txt. dosyasını açar ve histogram için gerekli diziye atılmış olur. Kodların içinde 0.1 ile çarpılmasının nedeni histogramın x ekseninde 100 tane sayı uzayıp gidebilir.



Şekil 5.1. yazılan program ve cygwin arayüzü ile çizdirilen grafik

Yapılmış olan bu çalışmada üretilmiş olan sayıların hassasiyeti 6' dır. En düşük yerde bulunan sayının hassasiyetinin önemi en azdır. Dolayısıyla 0.000000 – 0.000050 arasındaki tüm sayıların değerinin aynı olduğunu ve 0.000000 olduğunu kabul edelim. Bu yaklaşıma göre aynı şekilde 0.000050 – 0.000100 arasındakilerin hepsinin değeri de 0.000050 olacaktır.

$$0.000000 - 0.000050 = 0$$

$$0.000050 - 0.000100 = 1$$

$$0.000100 - 0.000150 = 2$$

Bu karşılık değerler (0,1,2) en fazla 19.999' a kadar gidebilir. Bu değerlerin bit karşılığını alarak bu sayıyı 16 bite paketleyebiliriz.

Yukarıda yazmış olduğumuz programı aşağıdaki şekilde geliştirebiliriz. Bu adıma geçmeden önce bir packing ve unpacking adımlarını katarak algoritma belirlenmelidir.

Packing:

- 1) Gelen sayıyı 0.000050' ya bölüp, bölüm alınır.
- 2) Bu sayı 16 bitlik yapıya çevirilir.
- 3) Bu sayıyı tanımlanmış olan short int içinde paketlenir.

Bi – paketleme, Unpacking:

- 1) short int olarak tanımlı binary bilgisi alınır.

- 2) Bu değerin onluk karşılığı bulunur.
- 3) Bulunan sayı 0.000050 ile çarpılır.

Bu yazılmış olan algoritmanın programa dönüştürülmüş hali ise,

```
#include <stdio.h>

void putvalue(int *ptr, int startbit, int endbit, int value);
void clearbits(int *ptr, int startbit, int endbit);
int getvalue(int x, int startbit, int endbit);

void putvalue(int *ptr, int startbit, int endbit, int value)
{
    int temp = value << startbit;
    clearbits(ptr, startbit, endbit);
}

void clearbits(int *ptr, int startbit, int endbit)
{
    int k;
    for (k = startbit; k <= endbit; ++k)
        *ptr &= ~(1 << k);
}

int getvalue(int number, int startbit, int endbit)
{
    int temp = number >>= startbit;
    clearbits(&temp, endbit + 1, sizeof(int) * 8 - 1);
}

void showbits(short int x)
{
    int i = sizeof(short int) * 8 - 1;

    for (; i >= 0; --i)
        if (x >> i & 1 == 1)
            putchar('1');
```

```
        else
            putchar('0');
    }

int main()
{
    int ch = 0x0041; /* ch = 65 0000 0000 0100 0001 */
    int mask = 0x0020; /* mask = 32 0000 0000 0010 0000 */

    ch |= mask; /* ch = 0000 0000 0110 0001 */
    printf("ch = %d\n", ch); /* ch = 97 */

    showbits(65);
    printf ("\n");

    int val=0;
    putvalue(&val, 4, 8, 6);

    printf("Yeni Deger: %d\n", val);
    showbits(val);
    printf ("\n");

    return 0;
}
```

Yapılan çalışmada bu adıma kadar gelinen noktada sayıların tamamı 16 bite kadar paketlenmiştir. Kaybımız hassasiyet en fazla 0.000050 oranındadır ve bu oranda ihmal edilebilir. Ayrıca bir dizi tanımlanarak tüm sayılar alınabilir. Bu diziler karşılığında paketlenmiş verileri tutabilir.

SONUÇ VE DEĞERLENDİRME

Yapmış olduğumuz araştırma ve çalışmalarda, temel parçacık fizikçileri tarafından araştırılan ve bulunmaya çalışılan birçok önemli sorunun (maddenin orijini nedir? , temel fermiyonların sayıları kaç tanedir? , maddeye kütle kazandıran nedir? , kuvvetler tek bir kuvvet altında toplanabilir mi?) modelleme çalışmaları yaparak cevaplarını bulmaya çalışıldı. Bu çalışmamızda dördüncü fermiyon ailesinin varlığının tüm dedektör modellemesi yaparak düşük luminisitede yıllık 10^5 olay sayısı hemde yüksek luminisitede yıllık 10^6 olay sayısı bulunarak LHC-ATLAS Deneyinde yüksek miktarda elde edilebileceğini böylece standart model ailesinin sayısının üç ile sınırlı olmadığını ve maddenin orijinini oluşturan temel parçacıkların sayılarının dört çiftten oluştuğu ispatlanmış oldu.

LHC Deneyi çalışmaya başladıktan sonra elde edilecek olan gerçek verilerle bu ailenin varlığının teyit edilmesi gerekir. Teyit edildiği takdirde bu konuda NOBEL ödülü kazanılacağı düşünülmektedir.

Bu tezde dedektör bilgileri, modelleme çalışmaları, programlama dillerinde birçok araştırma yapılmış ve çok değerli öğrenimlere sahip olunmuştur. Deneyin yapılacağı CERN ziyaret edilmiştir. Ayrıca CERN araştırma ziyareti boyunca üzerine ATLAS Deneyinin data alımı ile ilgili hazırlıkların yapıldığı bölümde gereksinim duyulan Bitwise operatörleri üzerine çalışılmış ve bu konuda program yazılmış ve CERN-ATLAS'taki bilim insanlarına teslim edilmiştir.

KAYNAKLAR

- [1] Engin Arık, Saleh Sultansoy, hep-ph/0302012 (2003)
- [2] S. Atağ, A. Çelikel, and A.K. Çiftçi, S.Sultansoy, Ü. O. Yılmaz, Physical Review D (1996)
- [3] Orhan Çakır, hep-ph/0301116v1 (2003)
- [4] S. Sultansoy, hep-ph/0004271 (2000)
- [5] E. Arik et al., ATLPHYS- 99-005, Observability of SM 4th Family Quarks at CERN-LHC, 1999.
- [6] David Griffiths, Introduction to Elementary Particles, 1987 John Wiley & Sons, Inc
- [7] <http://root.ch/> Erişim Tarihi: 17.03.2008
- [8] <http://atlasexperiment.org/> Erişim Tarihi: 23.04.2008
- [9] <http://www.biltek.tubitak.gov.tr/bilgipaket/madde/standart.html> Erişim Tarihi: 02.02.2008
- [10] http://www.ankarayefgrubu.web.tr/Orhan_Higgs.pd Erişim Tarihi: 12.04.2008
- [11] <http://www.athena-ip.org/> Erişim Tarihi: 21.05.2008
- [12] James William Rohlf, Modern Physics From α to Z^0 , 1994
- [13] Krane, S. K., (Çeviri Editörü: Şarer, B.), 2001, “Nükleer Fizik 1. Cilt”
- [14] <http://cmsinfo.cern.ch/outreach> Erişim Tarihi: 24.04.2008
- [15] <http://sourceforge.net/projects/xming> Erişim Tarihi: 05.05.2008
- [16] Anomalous single top quark production at the CERN LHC (O. Çakır ve S. A. Çetin)
- [17] Single Anomalous Production of the Fourth SM Family Quarks at Future e^+ , e^- , ep , and pp Colliders (A. K. Çiftçi, R. Çiftçi, S. Sultansoy, H. Duran Yıldız)
- [18] CKM Quarks-Mixing Matrix (A. Ceccucci, Z. Ligeti, Y. Sakai)

KAYNAKLAR (Devamı)

- [19] Anomalous single production of the fourth family quarks at the Fermilab Tevatron (E. Arık, O. Çakır, S. Sultansoy)
- [20] <http://public.web.cern.ch> Erişim Tarihi: 12.11.2007
- [21] <http://www.thep.lu.se/~torbjorn/Pythia.html> Erişim Tarihi: 18.05.2008
- [22] <https://atlas.gantep.edu.tr/twiki/bin/view/Main/WebHome> Erişim Tarihi: 30.05.2008
- [23] <https://atlas.gantep.edu.tr/GridWorkShop2008/> Erişim Tarihi: 19.06.2008
- [24] https://twiki.cern.ch/twiki/bin/view/Atlas/RegularComputingTutorial#Setting_up_your_account Erişim Tarihi: 16.07.2008
- [25] <https://twiki.cern.ch/twiki/bin/view/Atlas/WorkBook> Erişim Tarihi: 03.02.2008
- [26] <http://indico.cern.ch/conferenceDisplay.py?confId=36885#2008-07-16> Erişim Tarihi: 17.07.2008
- [27] <https://twiki.cern.ch/twiki/bin/view/Atlas/GangaTutorial44> Erişim Tarihi: 23.05.2008
- [28] <https://twiki.cern.ch/twiki/bin/view/Atlas/EventView> Erişim Tarihi: 23.05.2008
- [29] <https://tagservices.web.cern.ch/tagservices/tutorial/index.htm> Erişim Tarihi: 18.07.2008
- [30] <https://atldbdev01.cern.ch/tagservices/index.htm> Erişim Tarihi: 18.07.2008
- [31] <http://publib.boulder.ibm.com/infocenter/comphelp/v8v101/index.jsp?topic=/com.ibm.xlcp8a.doc/language/ref/bitande.htm> Erişim Tarihi: 13.07.2008
- [32] <http://www.vipan.com/htdocs/bitwisehelp.html> Erişim Tarihi: 13.07.008
- [33] <http://www.cs.cf.ac.uk/Dave/C/node13.html> Erişim Tarihi: 14.07.2008
- [34] <http://ch2.php.net/language.operators.bitwise> Erişim Tarihi: 14.07.2008
- [35] A.K. Çiftçi et al, 2005, Phys. Rev. D 72, 053006
- [36] Z. Kunszt, S. Moretti and W. Stirling (hep-ex/9611397)

KAYNAKLAR (Devamı)

[37] <http://www.grid.org.tr/> Erişim Tarihi:27.04.2008

[38] John Ellis, CERN- TH/98-403

[39] <http://www.atlasexperiment.org/detector.html> Erişim Tarihi: 21.10.2007

[40] PYTHIA 6.4, Physics and Manual, Torbjörn Sjöstrand