

YÜKSEK ENERJİ FİZİĞİNDE KULLANILAN YAZILIMLAR VE UYGULAMALAR

Prof. Dr. İlkey TÜRK ÇAKIR

*Ankara Üniversitesi
Hızlandırıcı Teknolojileri Esntitüsü
iturk@ankara.edu.tr*

NOT: Bu ders notlarında Yüksek Enerji Fiziğinde kullanılan yazılımlar ve uygulamaları hakkında genel bilgiler verilecektir. Bu yazılımlarda yapılabilecek en basit örnekler sunulmuştur. Yazılımlar ile ilgili ayrıntılı ve ileri düzeydeki bilgileri önceden hazırlamış olduğum ders notlarında kaynaklar kısımlarından ulaşabilirsiniz.

İÇERİK

- Bazı kavramlar
 - Simulasyon (benzetim)
 - Monte-Carlo (MC)
 - Olay ve Olay Sayısı
- Yüksek Enerji Fizikinde Kullanılan Programlardan Bazıları
 - Partonik Seviye Yazılım
 - CalcHEP/CompHEP
 - Uygulama 1
 - MadGraph

- Hadronik Seviye Yazılım
 - Pythia8
- Dedektör Benzetimi Yazılımı
 - Delphes
- Analiz Yazılımı
 - ROOT
- Uygulama 2

SİMÜLASYON (BENZETİM) ve MONTE CARLO (MC) NEDİR?

- Simülasyon (benzetim) gerçek bir problemin bilgisayarla yapılan modelidir. Böyle problemler doğada rastgele ve karmaşık olup çok sayıda değişken içerir. Bu tür problemlerin analitik çözümlerini bulmak oldukça zordur. Çözüm için ise yaklaşımlardan biri Monte Carlo (MC) gibi simülasyon teknikleri kullanarak sistemin bilgisayar modelinin oluşturulmasıdır.
- Simülasyonlar oldukça önemlidir, çünkü bunlar pahalı ve dikkat isteyen gerçek deneylerde, sistem parametrelerini değiştirerek sistemin davranışını gözlemlememize imkan sağlar. Simülasyonlar gerçekte meydana gelebilecek durumları oluşturmak ve bu olayların etkisini çalışmamıza imkan sağlar. Sayısal simülasyon yönteminde temel adım ardışık rastgele sayı üretilmesidir.

SİMÜLASYON (BENZETİM) ve (MC) NEDİR?

- MC simülasyonu veya MC yöntemi, adını Monako Prensiği'ndeki ünlü kumarhaneden dolayı almıştır. Oynanan rulet oyunu rastgele sayılar üretme mekanizmasının en basit örneğidir. Bu isim Stanislaw Marcin Ulam, Enrico Fermi, John von Neumann ve Nick Metropolis gibi ilk kullanıcıları tarafından popüler hale getirilmiştir.
- MC yöntemleri başta “istatistiksel örnekleme” gibi daha jenerik isimler altında uygulanmıştır. MC yöntemi, matematik veya fizik problemlerinin, bu problemlerin olasılık modellerine uygulanan rasgele denemeler yoluyla çözülmesi yöntemidir. Bu rasgele yöntemin en önemli kullanımı 1930 yılında Fermi tarafından, o yıllarda yeni keşfedilen nötronların özelliklerini hesaplamak için kullanılmıştır.



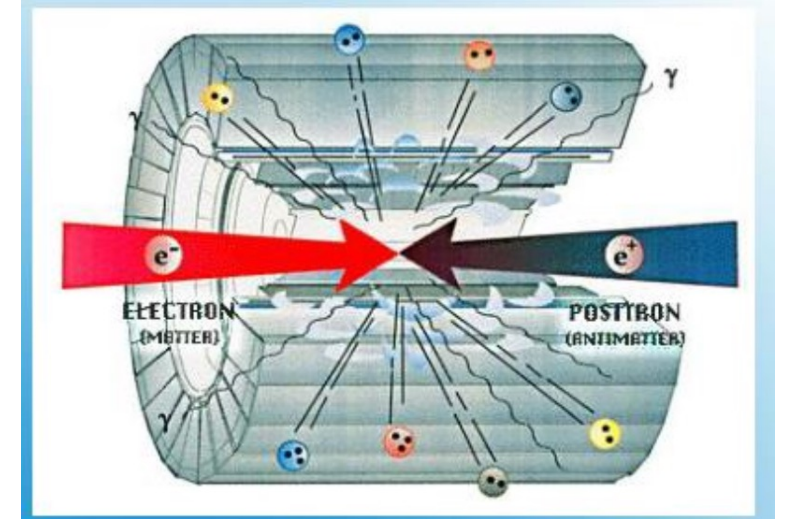
- Belirlemci (deterministic) algoritmaların kurulmasının olanaksız olduğu ya da çok karmaşık algoritmalar gerektiren problemlerin çözümünde başvurulan bir sayısal çözümleme tekniği olan MC yöntemi, sistemli olarak Manhattan Projesi'nin Los Alamos'taki laboratuvarında nötronların çeşitli maddelerden geçebilme özelliklerinin belirlenmesi çalışmalarında kullanılmıştır (1943).

SİMÜLASYON (BENZETİM) ve (MC) NEDİR?

- MC yöntemine çok yalın bir örnek, kapalı bir eğri içinde kalan bir alanın hesaplanmasıdır. Kapalı eğri kenar uzunluğu birim olarak alınan karenin içine yerleştirilir. 0 ile 1 arasında ve birbirinden bağımsız olarak üretilen iki rasgele sayı çifti, kare içinde rasgele bir noktayı belirler. Çok sayıda üretilen böyle noktalardan kapalı eğri içine rastlayanların sayısının üretilen toplam nokta sayısına oranı, eğri içinde kalan alanın yaklaşık değerini verir; üretilen nokta sayısı arttıkça yaklaşıklık daha doğru olur. MC yönteminin bilgisayarlar kullanılması bilgisayarda rasgele sayı üretilmesi yöntemlerinin geliştirilmesini gerektirmiştir.
- MC yönteminin oldukça fazla uygulama alanı mevcuttur: Sayısal analiz, atom ve molekül fiziği, nükleer fizik, yüksek enerji fiziği, dedektör simülasyonu, hücre simülasyonu, borsa modelleri, dağılım fonksiyonları uygulamalarında kullanılır. Çok katlı integrallerin hesaplanmasında, kısmi diferansiyel denklemlerin, doğrusal denklem sistemlerinin çözümünde, nötron yayını, gama ışını, soğurulması problemlerinde, paraçacık fiziğinde bozunum genişliklerinin ve saçılma kesitlerinin hesaplanması gibi birçok yerde kullanılır.
- NOT: Rasgele sayı: Bir kümenin veya dizinin elemanlarından bir kısmının istatistiksel olarak rasgele seçilmesi yoluyla üretilmiş sayıya rasgele sayı denir. Örneğin, zar atmak, tombala çekmek, rulet, kumar makinaları v.b. Bilgisayar belirlenimli bir aygıt olduğundan üretilen sayılar aslında tam olarak nitelikli değildir. Rasgele sayılarda bulunması gereken özellikleri belli ölçüde içeren bu sayılar bu nedenle “sözde rasgele sayılar” olarak adlandırılır.

OLAY NEDİR?

- Doğrusal çarpıştırıcısı (LC) veya Büyük Hadron Çarpıştırıcısı (LHC) yüksek enerji fiziği deneyleri, yüksek performanslı hesap ortamı için en çok istenen alt yapılardır. Parçacıklara yeteri kadar enerji verdikten sonra, bu parçacıklar ya bir hedefle yada birbirleriyle çarpıştırılırlar. Bu çarpışmaların her biri, **OLAY** olarak adlandırılır. Bir yüksek enerji fizikçisinin amacı, her bir olayı ayıklayıp, bu tek olaydaki verileri toplayıp, bu olaydaki parçacık sürecininin test edilen teori ile uyuşup uyuşmadığını incelemektir. Her olay, bir çok parçacık üretildiğinden çok karmaşıktır. Dolayısıyla, başka parçacıklara bozunmadan önce çok kısa mesafeler gittiklerinden gözlenebilir izler bırakmazlar. Dedektörler (algıçlar) bu izleri ve bu sinyalleri algılayan aletlerdir.

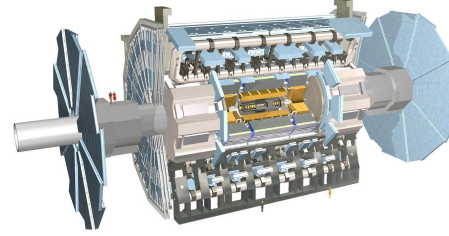


MONTE CARLO BENZETİM ADIMLARI

Veriler

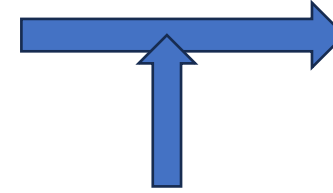


Çarpışma



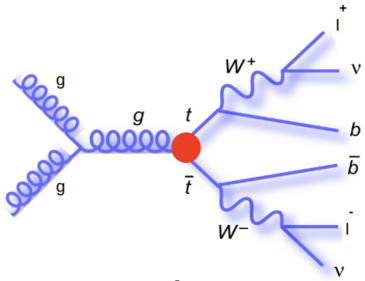
Algılama /
Algıç

Yeniden oluşturma
/ Yapılandırma

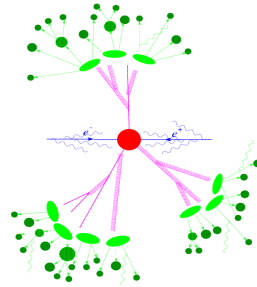


Analiz
girdileri

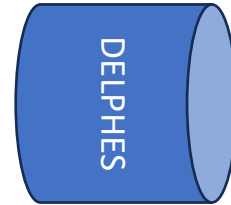
MC



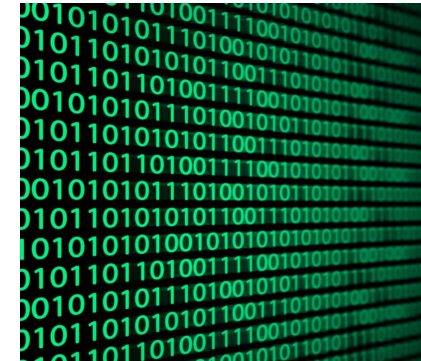
Matris elemanı
hesaplanması



Parton sağanağı /
hadronlaşma



Dedektor
benzetim



Dijitalleştirme

OLAY SAYISI

Toplanmış Işınık

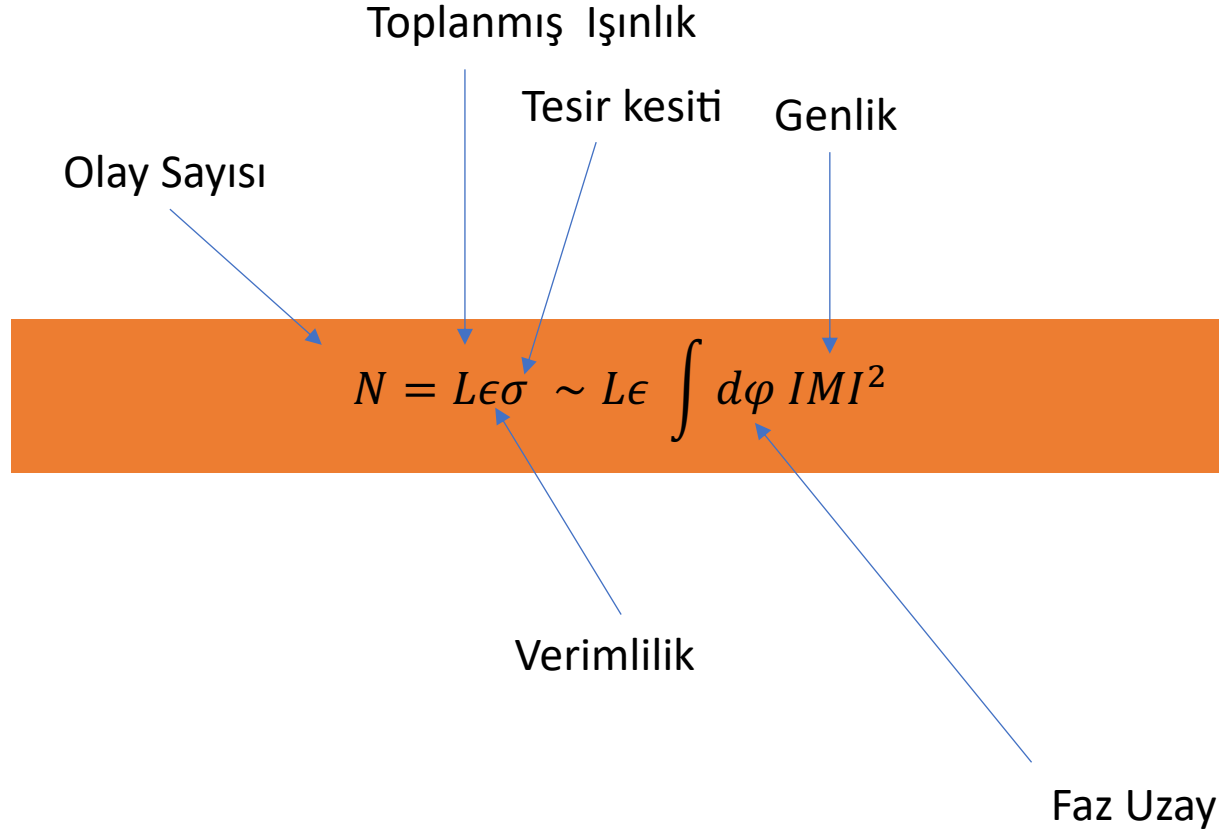
Tesir kesiti Genlik

Olay Sayısı

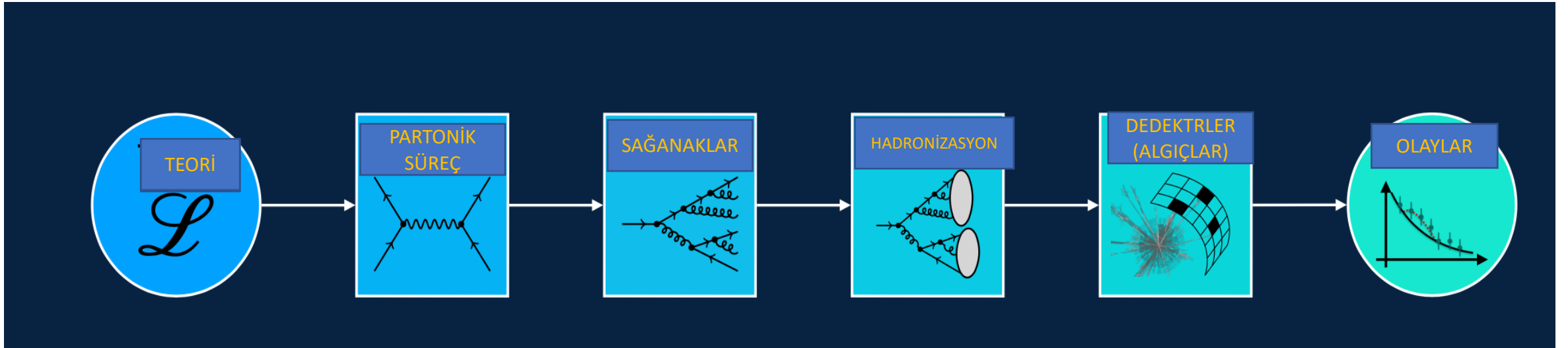
$$N = L\epsilon\sigma \sim L\epsilon \int d\varphi IMI^2$$

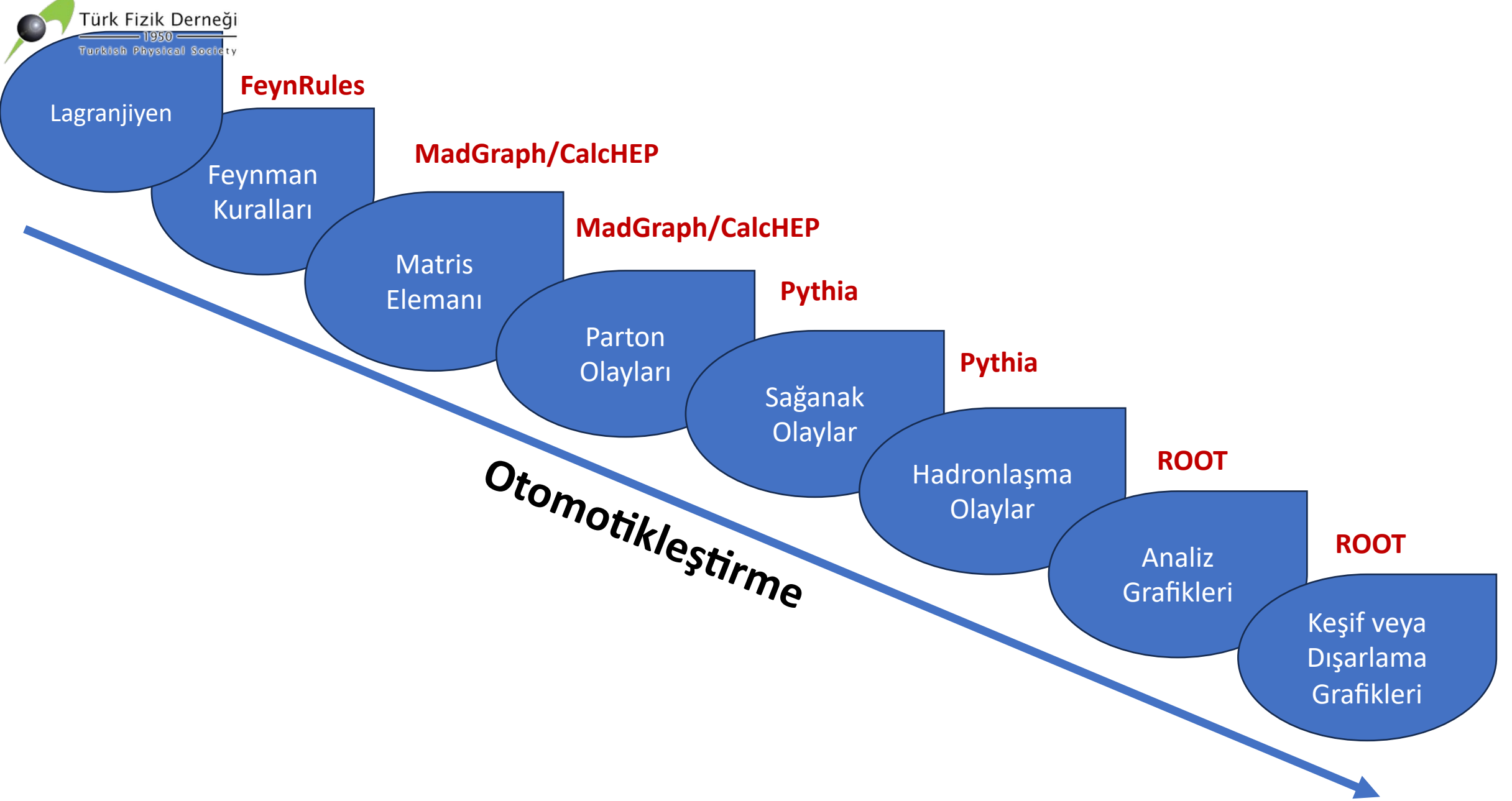
Verimlilik

Faz Uzayı



FİZİK OLAYLARI





YEF'DE KULLANILAN BAZI PROGRAMLAR

CalcHEP, MadGraph ve PYTHIA olay üretimine giden son hesap aşamalarını otomatikleştirmeye çalışan programlardır. Verilen bir saçılma süreci için temel basamaklar aşağıdaki gibidir:

- Süreçte bulunan Feynman diyagramlarının çizilmesi
- Matris elemanı ifadelerinin bulunması
- Faz uzayı üzerinden integralin alınması
- Toplam veya diferensiyel tesir kesitinin bulunması
- Herbiri son durum parçacıklarına karşı gelen dörtlü enerji-momentum vektörleri kümesi olarak toplam diferensiyel tesir kesitine göre olaylar **rasgele** olarak üretilir.

CalcHEP

Olay üretici Monte Carlo programıdır.

- Çoklu parçacıkların çarpışmaları ve bozunmaları sürecinde olayların üretimi ve olay üretim işlemi amacıyla hazırlanan analiz yapma ve hesaplama paketidir.
- CalcHep içerisinde parçacıklar semboller ve rakamlar ile gösterilir. YEF'deki hesaplama metotlarının otomatikleştirilmesi için kullanılmaktadır.

<https://theory.sinp.msu.ru/~pukhov/calchep.html>

CalcHEP - a package for calculation of Feynman diagrams and integration over multi-particle phase space.

Authors - Alexander Pukhov, Alexander Belyaev, Neil Christensen

The main idea of CalcHEP is to enable one to go directly from the Lagrangian to the cross sections and distributions effectively, with a high level of automation. The package can be compiled on any Unix platform.

General information

[Main features](#), [Acknowledgments](#), [Publications&Lectures](#), [Contributions](#)

Manual

[calchep_man_3.3.6.pdf](#) (manual for version 3.3.6, July 19, 2012)

[HEP computer tools](#) (Lecture by Alexander Belyaev)

See also: [Dan Green, High Pt physics at hadron colliders](#) (Cambridge University Press)

Code download.

[License GPL-3](#)

[Installation](#), [Current version 3.8.10](#) (updated 07.06.2021), [New Options and Bugs Fixed](#), [All versions](#)

Models:

[MSSM_10.14\(15.10.2014\)](#), [NMSSM_8.15\(25.08.2015\)](#), [CPVMSSM_10.14\(16.10.2014\)](#), [SUSY models By A.Semenov](#), [LeptoQuarks](#), [SDSM](#), [6DSM](#)

Model database [HEPMDB](#)

Related packages on Web:

Packages for model generation: [LanHEP](#), [FeynRules](#), [SARAH](#)

RGE and spectrum calculation: [SuSpect](#), [Isajet](#), [SoftSUSY](#), [SPheno](#), [CPsuperH](#), [NMSSMTools](#)

Particle widths in MSSM: [SUSY-HIT](#), [HDECAY](#)

Parton showers: [PYTHIA](#)

Contacts

Email: calchep@googlegroups.com

Launchpad service: [Ask a question](#), [File a bug](#)

CalcHEP İLE NELER YAPILABİLİR?

- **Dallanma Oranı Hesaplayabiliriz 😊**
- **Ağaç Seviyesinde Tesir Kesiti Hesaplayabiliriz 😊**
- **Feynman Çizimleri Yapabiliriz 😊**
- **Olay Üretimi 😊**
- **Yeni süreç girebiliriz 😊**

Not: En çok kullanacağınız tuş: esc tuşu

CalcHEP KURULUM

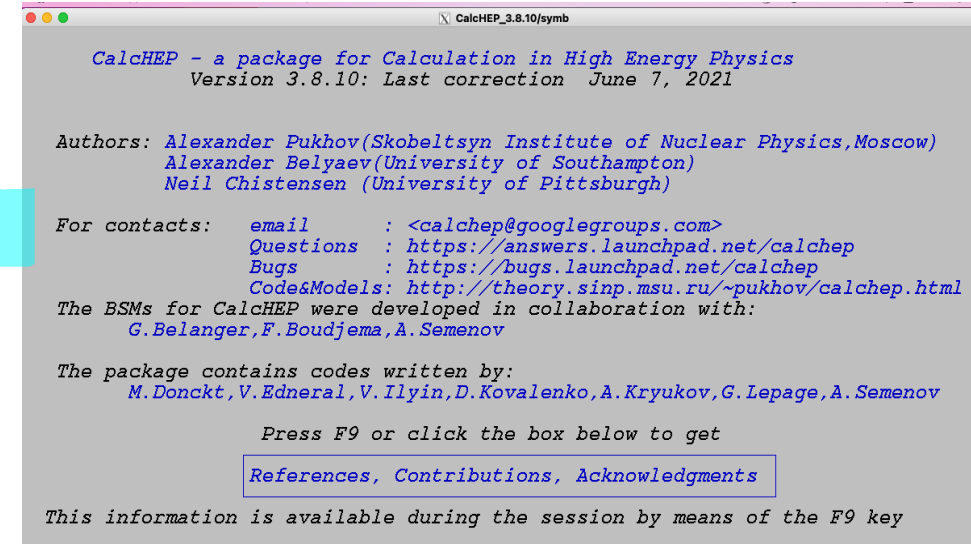
- **tar zxvf calchep_3.8.x.tgz**
- **cd calchep_3.8.x**
- **make**
- **./mkWORKdir ../calcworkilk (istenilen bir isim verilebilir)**
- **cd ..**
- **cd calcworkilk**

CalcHEP

```
#-----  
# CalcHEP has compiled successfully and can be started.  
# The manual can be found on the CalcHEP website:  
#   http://theory.sinp.msu.ru/~pukhov/calchep.html  
# The next step is typically to run  
#   ./mkWORKdir <new_dir>  
# where <new_dir> is the new directory where you will do  
# your calculations. After creating this directory, you  
# should cd into it and run calchep or calchep_batch.  
# Please see the manual for further details.  
#-----
```

```
ilkay@ilkay1 calchep_3.8.10 % ./mkWORKdir ../calcworkilk  
ilkay@ilkay1 calchep_3.8.10 % cd ..  
ilkay@ilkay1 Desktop % cd calcworkilk  
ilkay@ilkay1 calcworkilk %
```

```
ilkay@ilkay1 Desktop % cd calcworkilk  
ilkay@ilkay1 calcworkilk % ./calchep
```



```
CalcHEP_3.8.10/symb  
CalcHEP - a package for Calculation in High Energy Physics  
Version 3.8.10: Last correction June 7, 2021  
  
Authors: Alexander Pukhov(Skobeltsyn Institute of Nuclear Physics, Moscow)  
Alexander Belyaev(University of Southampton)  
Neil Chistensen (University of Pittsburgh)  
  
For contacts:   email       : <calchep@googlegroups.com>  
                Questions  : https://answers.launchpad.net/calchep  
                Bugs       : https://bugs.launchpad.net/calchep  
                Code&Models: http://theory.sinp.msu.ru/~pukhov/calchep.html  
  
The BSMs for CalcHEP were developed in collaboration with:  
G.Belanger, F.Boudjema, A.Semenov  
  
The package contains codes written by:  
M.Donckt, V.Edneral, V.Ilyin, D.Kovalenko, A.Kryukov, G.Lepage, A.Semenov  
  
Press F9 or click the box below to get  
References, Contributions, Acknowledgments  
  
This information is available during the session by means of the F9 key
```

UYGULAMA-1

- 1- Bir model seçelim.
Örneğin Standart Model seçilir.
- 2- Bir süreç seçelim.
İstenmeyen alt süreçler atılabilir.
- 3- Feynman çizimlerinin karesini alıp sonucu "C" dilinde yazdıralım.
- 4- "C" programını derleyip sayısal inceleme yapalım. Sonuçlar burada elde edebiliriz.
- 5- Hızlandırıcı da burada seçelim.
- 6- Tesir kesitlerini ISR ve BS etkileri dahil ederek hesaplayalım.

The screenshot displays the CalcHEP software interface. At the top right, there is a logo for the Türk Fizik Derneği (Turkish Physical Society) established in 1950. The main window is titled "CalcHEP_s.6.10/symb" and shows the "Abstract" screen. The abstract text describes the CalcHEP package's purpose for calculating decay and high energy collision processes. Below the abstract, there are two menu panels. The first panel, titled "SM", lists options: "SM(+hgg)", "SM(+hgg+h4G)", "IDM", "IDM(+hgg)", and "IMPORT MODEL". The second panel, titled "Enter Process", lists options: "Force Unit.Gauge= OFF", "Edit model", "Numerical Evaluation", and "Delete model". At the bottom of the window, there is a menu bar with options: "F1-Help F2-Man F5-Switches F6-Results F9-Ref F10-Quit".

CalcHEP

Abstract

CalcHEP package is created for calculation of decay and high energy collision processes of elementary particles in the lowest order (tree) approximation. The main idea put into the CalcHEP was to make available passing from the lagrangian to the final distributions effectively with the high level of automatization.

Use F2 key to get information about interface facilities and F1 - as online help.

Questions: <https://answers.launchpad.net/calchep>

Bugs: <https://bugs.launchpad.net/calchep>

```
<
SM
SM(+hgg)
SM(+hgg+h4G)
IDM
IDM(+hgg)
IMPORT MODEL
```

A(A) - photon
W+(W-) - W boson
nm(Nm) - mu-neutrino
l(L) - tau-lepton
c(C) - c-quark
b(B) - b-quark

Z(Z) - Z boson
ne(Ne) - neutrino
m(M) - muon
u(U) - u-quark
s(S) - s-quark
h(h) - Higgs

G(G) - gluon
e(E) - electron
nl(Nl) - tau-neutrino
d(D) - d-quark
t(T) - t-quark

e-e⁺ → μ μ

```
inter process: e,E -> m,M
```

CalcHEP

CalcHEP_3.8.10/symb

Model: SM

Process: e,E -> m,M

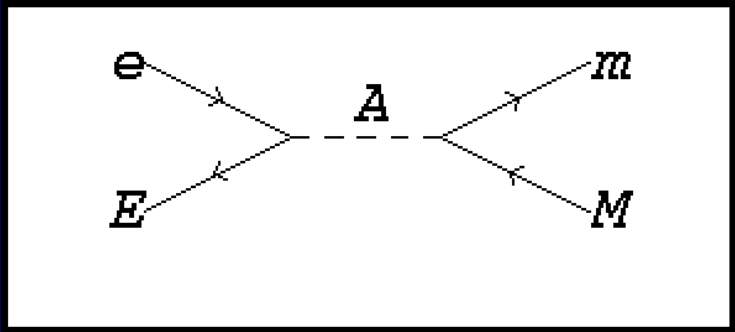
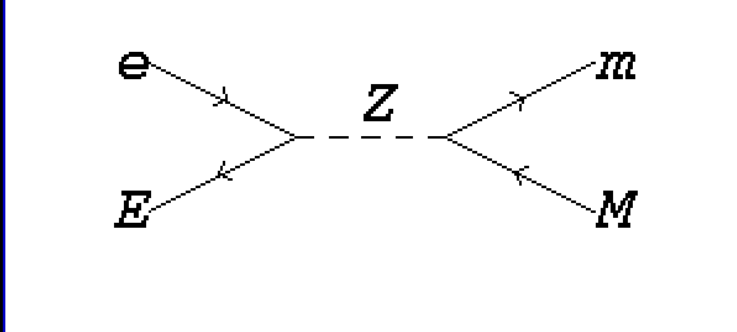
Feynman diagrams

2 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.

View diagrams
Square diagrams
Write down processes

CalcHEP_3.8.10/s

Delete, On/off, Restore, Latex

| | |
|---|--|
|  |  |
|---|--|

Model: SM

Process: $e, E \rightarrow m, M$

Feynman diagrams

2 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.

<
View diagrams
Square diagrams
Write down processes

Model: SM

Process: $e, E \rightarrow m, M$

Feynman diagrams

2 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.

Squared diagrams

3 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.
0 diagrams are calculated.

<
View squared diagrams
Symbolic calculations
Make&Launch n_calchep
Make n_calchep
REDUCE program

CalcHEP

```
CalcHEP_3.8.10/symb
Model: SM
Process: e,E -> m,M

Feynman diagrams
2 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.

Squared diagrams
3 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.
0 diagrams are calculated.
```

View squared diagrams
Symbolic calculations
Make&Launch n_calchep
Make n_calchep
REDUCE program

```
CalcHEP_3.8.10/symb
Model: SM
Process: e,E -> m,M

Feynman diagrams
2 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.

Squared diagrams
3 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.
3 diagrams are calculated.
```

C code
C-compiler
Edit Linker
REDUCE code
MATHEMATICA code
FORM code
Enter new process

```
CalcHEP_3.8.10/symb

Model: SM

Process: e,E -> m,M

Feynman diagrams
2 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.

Squared diagrams
3 diagrams in 1 subprocesses are constructed.
0 diagrams are deleted.
3 diagrams are calculated.
```

<

- C code
- C-compiler**
- Edit Linker
- REDUCE code
- MATHEMATICA code
- FORM code
- Enter new process

```
CalcHEP_3.8.10/num

(sub)Process: e, E -> m, M
Monte Carlo session: 1(begin)

#IT Cross section[pb] Error[%] nCall Eff. chi^2
```

<

- Subprocess
- IN state**
- Model parameters
- Constraints
- QCD alpha & scales
- Breit-Wigner
- Aliases
- Cuts
- Phase space mapping
- Monte Carlo simulation
- 1D integration

```
(sub)Process: e, E -> m, M
Monte Carlo session: 1

#IT Cross section[pb] E <
  S.F.1: OFF
  S.F.2: OFF
  First particle momentum[GeV] = 4000
  Second particle momentum[GeV] = 4000
  First particle unpolarized
  Second particle unpolarized
```

ISR&BS Etkisini (Hızlandırıcımıza Uygun Olarak) Göz Önüne Alalım

```
Model: SM
CalcHEP_3.8.10/num

(sub)Process: e, E -> m, M
Monte Carlo session: 1(begin)

#IT Cross section[pb] E <
  S.F.1: Qis
  Qisr=1.00E0*sqrtS,Beamstr:560.0,0.400,2.
```

```

(sub)Process: e, E -> m, M
Monte Carlo session: 1(begin)

#IT Cross section[pb] E

```

IN state

S.F.1: Q_{ISR}=1.00E0*sqrtS, Beamstr:560.0,0.400,2.0E+10

```

ISR scale      = 1.00E+00*sqrtS
Beamstrahlung  ON
Bunch x+y sizes (nm)= 560.0
Bunch lenght (mm) = 0.400
Number of particles = 2.0e+10
      * N_gamma = 1.53
      * Upsilon = 0.63
Beamstrahlung F(x) plot
Beamstrahlung F(x)*(1-x)^(2/3)

```

ISR: Initial State
Radiation

BS:Bremsstrahlung

```

(sub)Process: e, E -> m, M
Monte Carlo session: 1(begin)

#IT Cross section[pb] E

```

IN state

S.F.2: Q_{ISR}=1.00E0*sqrtS, Beamstr:560.0,0.400,2.0E+10

```

ISR scale      = 1.00E+00*sqrtS
Beamstrahlung  ON
Bunch x+y sizes (nm)= 560.0
Bunch lenght (mm) = 0.400
Number of particles = 2.0e+10
      * N_gamma = 1.53
      * Upsilon = 0.63
Beamstrahlung F(x) plot
Beamstrahlung F(x)*(1-x)^(2/3)

```

ISR&BS Etkisini (Hızlandırıcımıza Uygun Olarak) Göz Önüne Alalım

- ISR: K ışımadan yayımlanan foton radyasyonudur. Bu etki kütle merkezini azaltabilir. Tesir kesitini değiştirebilir.
- BS: Parçacıkların elektrik alanından dolayı enerji kaybedebilir. Bu spektrum doğrusal çarpıştırıcının geometrisine (bunch x+y sizes), demetin boyutlarına (bunch lenght), demetdeki parçacık sayısına (bunch of particles), demet enerjisine göre değişim göstermektedir.

```
CalcHEP_3.8.10/num  
  
(sub)Process: e, E -> m, M  
Monte Carlo session: 1(begin)  
  
#IT Cross section[pb] E  
S.F.1: Qisr=1.00E0*sqrtS,Beamstr:560.0,0.400,2.0E+10  
S.F.2: Qisr=1.00E0*sqrtS,Beamstr:560.0,0.400,2.0E+10  
First particle momentum[GeV] = 4000  
Second particle momentum[GeV] = 4000  
First particle unpolarized  
Second particle unpolarized  
  
CalcHEP_3.8.10/num  
  
(sub)Process: e, E -> m, M  
Monte Carlo session: 1(begin)  
  
#IT Cross section[pb] Error[%] nCall Eff. chi^2  
Subprocess  
IN state  
Model parameters  
Constraints  
QCD alpha & scales  
Breit-Wigner  
Aliases  
Cuts  
Phase space mapping  
Monte Carlo simulation  
1D integration
```

CalcHEP_3.8.10/num

```
(sub)Process: e, E -> m, M
Monte Carlo session: 1(begin)
```

```
#IT Cross section[pb] Error[%] nCall Eff. chi^2
```

Monte Carlo simulation

```
nSess = 5
nCalls = 100000
Set Distributions
*Start integration
Display Distributions
Clear statistic
Freeze grid OFF
Clear grid
```

CalcHEP_3.8.10/num

```
(sub)Process: e, E -> m, M
Monte Carlo session: 1(begin)
```

```
#IT Cross section[pb] Error[%] nCall Eff. chi^2
39 1.3357E-02 1.06E-01 98568
40 1.3344E-02 1.06E-01 98568
< > 1.2663E-02 9.47E-01 3942720 6
41 1.3351E-02 1.07E-01 98568
42 1.3348E-02 1.06E-01 98568
43 1.3357E-02 1.07E-01 98568
44 1.3367E-02 1.07E-01 98568
45 1.3368E-02 1.06E-01 98568
< > 1.2740E-02 8.36E-01 4435560 7
```

Monte Carlo simulation

*Start integration

Integration is over
— Press any key —


```
(sub)Process: e, E -> m, M
Monte Carlo session: 1(begin)
```

| #IT | Cross section[pb] | Error[%] | nCall | Eff. | chi ² |
|-----|------------------------------|----------|---------|------|------------------|
| 39 | 1.3357E-02 | 1.06E-01 | 98568 | | |
| 40 | 1.3344E-02 | 1.06E-01 | 98568 | | |
| < > | 1.2663E-02 | 9.47E-01 | 3942720 | | 6 |
| 41 | 1.3351E-02 | 1.07E-01 | 98568 | | |
| 42 | 1.3348E-02 | 1.06E-01 | 98568 | | |
| 43 | 1.3357E-02 | 1.07E-01 | 98568 | | |
| 44 | 1.3367E-02 | 1.07E-01 | 98568 | | |
| 45 | 1.3368E-02 | 1.06E-01 | 98568 | | |
| < > | 1.2740E-02 | 8.36E-01 | 4435560 | | 7 |

Monte Carlo simulation

```
nSess = 5
nCalls = 100000
Set Distributions
*Start integration
Display Distributions
Clear statistic
Freeze grid OFF
Clear grid
```

```
(sub)Process: e, E -> m, M
```

* **Distributions** 1

| Clr | Del | Size | Read | ErrMes | Parameter_1/ | Min_1 | Max_1 | Parameter_2/ | Min_2 | Max_2 |
|-----|-----|------|------|--------|--------------|-------|-------|--------------|-------|-------|
| | | | | | M(m,M) | 0 | 200 | | | |

```

CalcHEP_3.8.10/num

(sub)Process: e, E -> m, M
Monte Carlo session: 1(continue)

#IT Cross section[pb] Error[%] nCall Eff. chi^2
51 1.3341E-02 1.06E-01 98568
52 1.3375E-02 1.07E-01 98568
53 1.3348E-02 1.07E-01 98568
54 1.3359E-02 1.06E-01 98568
55 1.3349E-02 1.06E-01 98568
< > 1.2853E-02 6.78E-01 5421240 7
56 1.3351E-02 1.06E-01 98568
57 1.3344E-02 1.07E-01 98568
    
```

Monte Carlo simulation

*Start integration

Integration is over
—Press any key—

```

CalcHEP_3.8.10/num

(sub)Process: e, E -> m, M
Monte Carlo session: 1(continue)

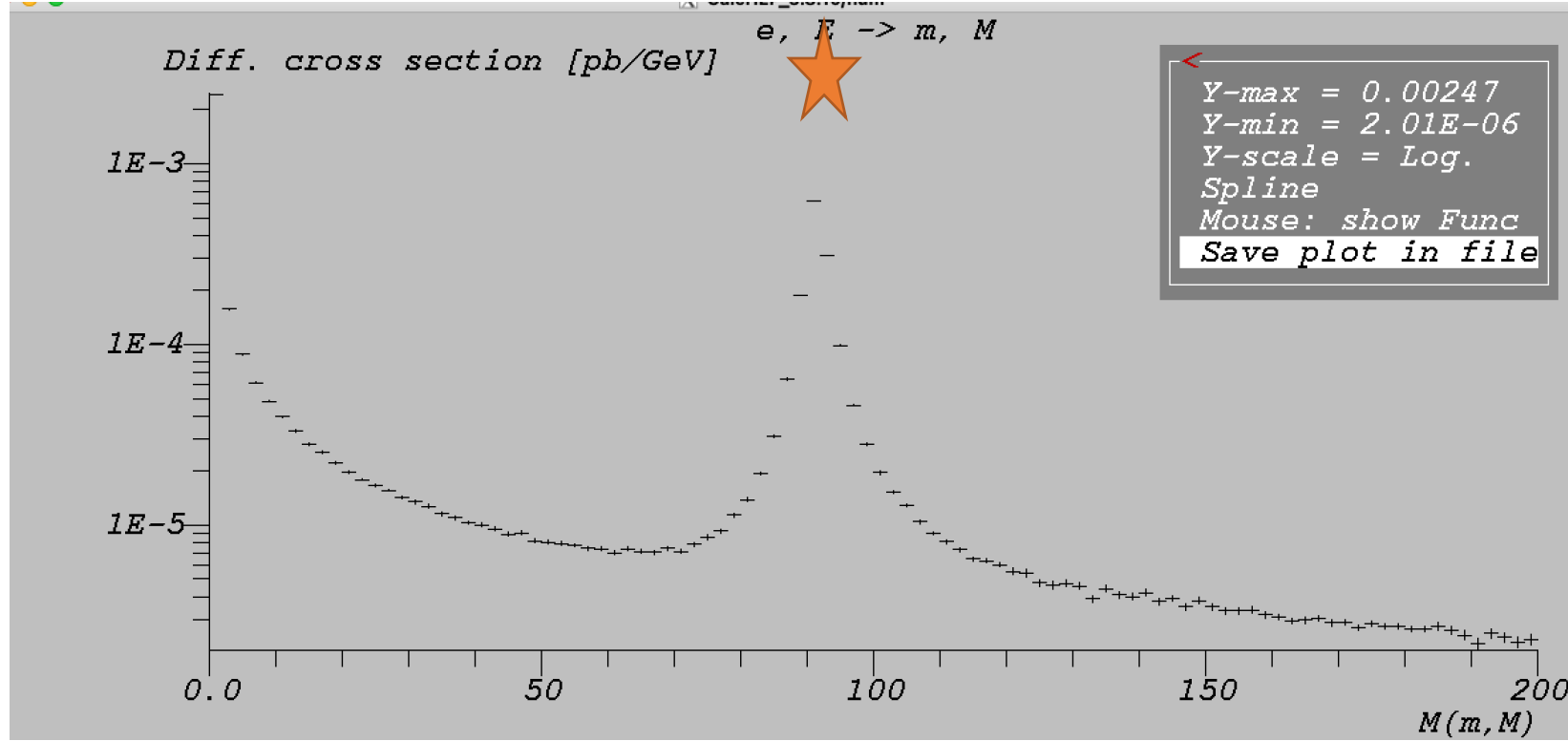
#IT Cross section[pb] Error[%] nCall Eff. chi^2
51 1.3341E-02 1.06E-01 98568
52 1.3375E-02 1.07E-01 98568
53 1.3348E-02 1.07E-01 98568
54 1.3359E-02 1.06E-01 98568
55 1.3349E-02 1.06E-01 98568
< > 1.2853E-02 6.78E-01 5421240 7
56 1.3351E-02 1.06E-01 98568
57 1.3344E-02 1.07E-01 98568
    
```

Monte Carlo simulation

Display Distributions

Distributions

M(m, M)



Değişmez kütle dağılımı grafiği

CalcHEP/CompHEP için kaynaklar

1- Hızlandırıcı ve Parçacık Fiziğinde Bilgisayar Uygulamaları Okulu, Anadolu Üniversitesi,
İlkay TÜRK ÇAKIR, Ders Notları: <https://indico.cern.ch/event/335370/contributions/781762/>

2- Hızlandırıcı ve Parçacık Fiziğinde Bilgisayar Uygulamaları Okulu, Gaziosmanpaşa Üniversitesi
İlkay TÜRK ÇAKIR, Ders Notları: <https://indico.cern.ch/event/264054/contributions/592261/>

3- CalcHEP 3.4 for collider physics within and beyond the Standard Model (Alexander Belyaev^{1,2}, Neil D. Christensen³, Alexander Pukhov) : <https://arxiv.org/pdf/1207.6082.pdf>

4- Practical introduction into CalcHEP

https://indico.cern.ch/event/656211/contributions/2756825/attachments/1547486/2429259/calchep_tools_bootcamp_belyaev.pdf

5-Veri analizi, Ders Notları, İlkay TÜRK ÇAKIR

<https://www.slideserve.com/atira/ver-anal-z-i>

MadGraph5 +Pythia8+ Delphes+ ROOT

- MadGraph5
 - Süreç tanımlama, diyagramlar, parametreler
- Pythia8
 - Hadronlaşma, bozunum
- Delphes
 - Dedektör hızlı simülasyonu, bilinen dedektörler için kartlar, parametreler
- ROOT
 - Analiz

MadGraph Özellikleri

- MadGraph programı, Standart Model (SM) ve SM ötesi fenomenoloji için gerekli tüm bileşenleri sağlayan bir pakettir.
- Ağaç seviyesinde matrix elemanı hesaplayıcısı, tesir kesiti hesabı ve olay üretimi yapar.
- Uzun süre C++/Fortran ile hazırlanan MG4 kullanılmış, 2010 yılından bu yana “Python” dilinde hazırlanan MG5 kullanılmaktadır.
- Diğer MC üreticilerine göre; Grid üzerinde, çok çekirdekli ya da küme bilgisayarlarda (pbs) daha başarılı hesaplama yapabilir.
- Terminal üzerinde arayüz sunar. Sonuçlar html olarak verilebilir. Günümüzde olayları simüle etmek için yaygın olarak kullanılmaktadır.

MadGraph İLE NE YAPILIR?

- Bu takım, parton seviyesindeki basit bir tesir kesiti

$$\sigma = \frac{1}{F} \int dPS^{(n)} \overline{|M_{fi}|^2}$$

- Matris elemanı karesi: Feynman diagramlarının genlikleri yoluyla gelir, diagramlar modelinin tüm etkileşim ve parçacık bilgisini taşıyan Lagrangieni ile elde edilir.
- F başlangıç durumları üzerinden alınan ortalama akı faktörlerdir.
- Tüm faz uzayı (PS) üzerinden integral alınarak hesaplanır.

MadGraph İLE NE YAPILIR?

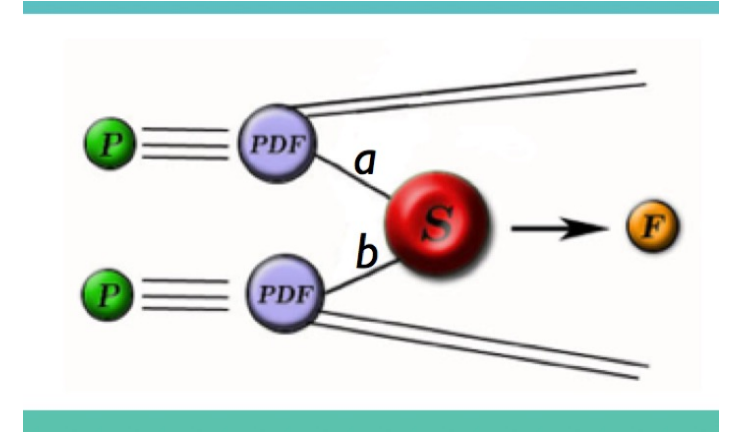
- Hadron Çarpıştırıcısında bir tesir kesiti hesabı;

$$\sigma = \frac{1}{F} \sum_{ab} \int d\text{PS}^{(n)} dx_a dx_b f_{a/p}(x_a) f_{b/p}(x_b) \overline{|M_{fi}|^2}$$

- Tüm partonlar üzerinden toplam alınır (a ve b).

- f: parton dağılım fonksiyonları (PDF)

- a ve b protondan gelen birer parton ve x parton fraksiyonları olmak üzere f(x) partonların protonlardan gelme olasılığını gösterir.

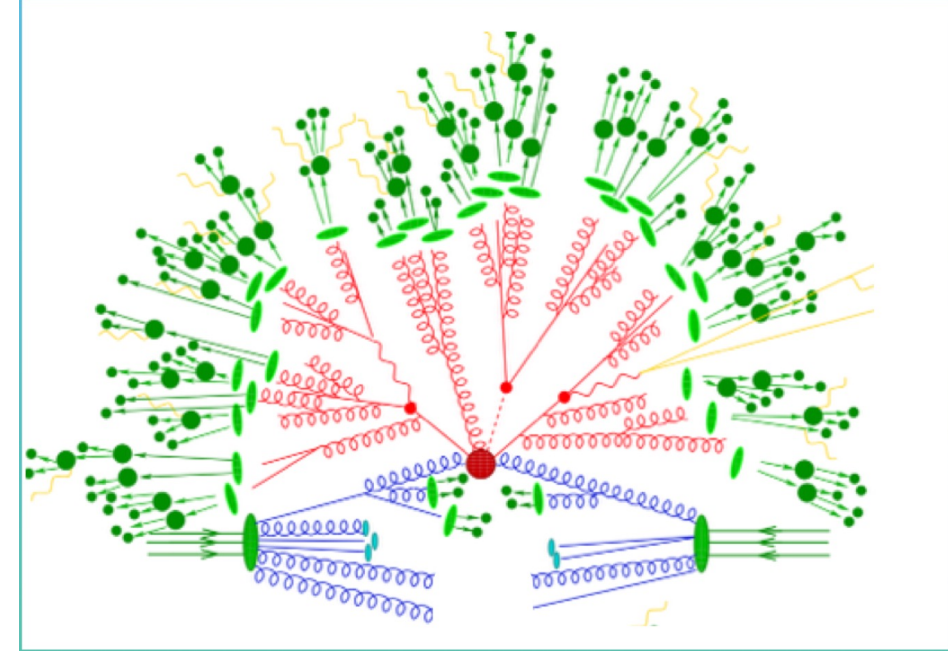


MadGraph NE İÇERİR?

- MadGraph yazılımı, çeşitli programları içerecek şekilde genişletilebilir: bir rastgele olay üretici, parton duşu ve hadronizasyon için kullanılan kod Pythia ve iki dedektör simülatörü (PGS ve Delphes).
- İlgilendiğimiz simülasyon türüne bağlı olarak, bu ek parçalardan bazıları gereksiz olabilir. Örneğin, sadece parton seviyesinde bir tesir kesiti hesaplamak istiyorsak, temel formülü kullanmak yeterlidir. Bununla birlikte, bunun ötesine geçmek ve hadronizasyon veya dedektör simülasyonunu dahil etmek istiyorsak, **Pythia** ve PGS veya **Delphes'i** de kullanmak zorundayız. Bu biraz karmaşık gelebilir, ancak pratikte bu araçların kombinasyonu basittir ve aslında MadGraph çerçevesi bunu otomatik olarak yapmaya hazırdır.

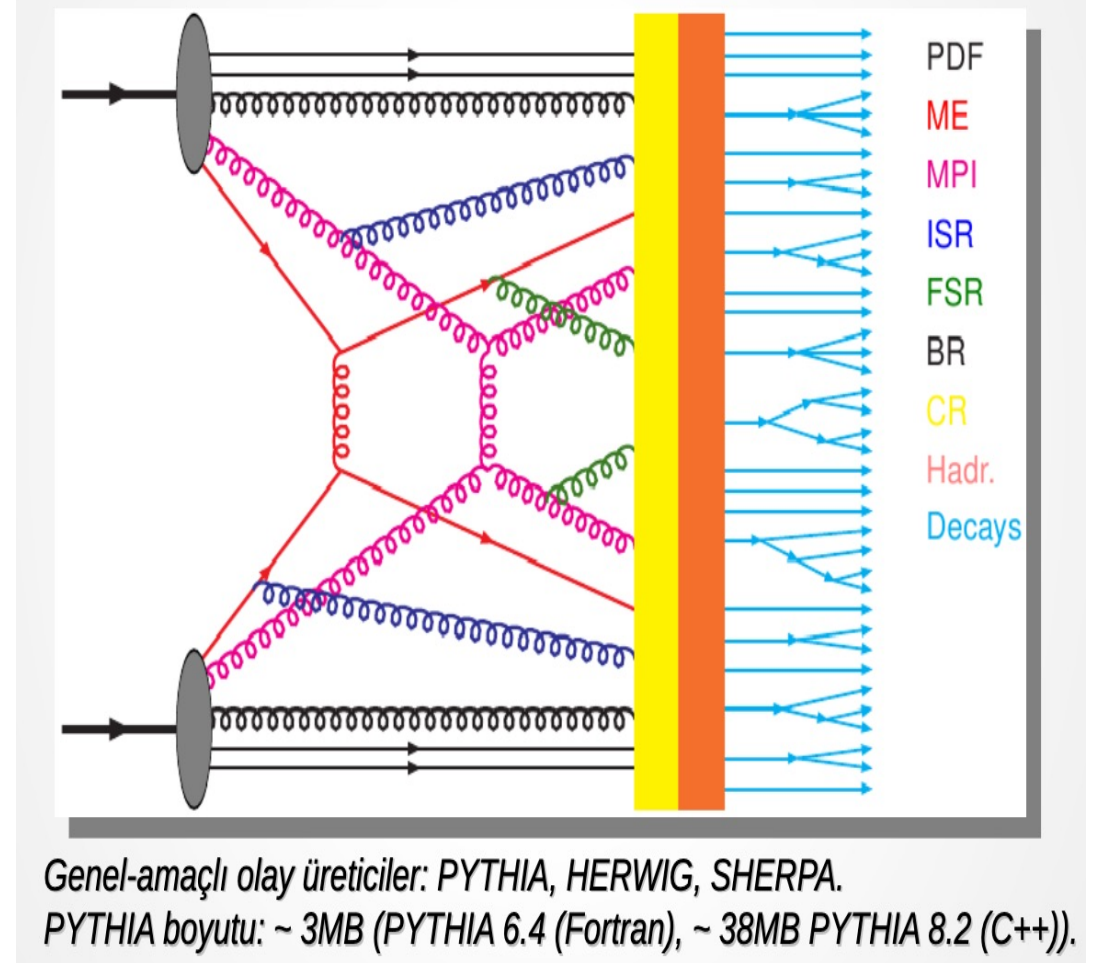
MadGraph NE İÇERİR?

- Yüksek enerjili çarpışmalarda, detektörlerde son aşamada ortaya çıkan hadronlaşma gözlemlenir.
- **Hadronlaşma:** Başlangıçta sürece giren kuark ya da gluonların, son durumda bozunarak asimptotik özgürlüğün izin verdiği ölçüde bir araya gelerek yine kuark ya da gluon ışınımı yapmasıdır.



PYTHIA

- Seçilen fizik modeli çerçevesinde MC teknikleri kullanarak yüksek enerji fiziği **olaylarını üretme** programıdır.
- Parçacık çarpışmalarının ve etkileşmelerinin benzetiminde kullanılır.
- PYTHIA, orta ve yüksek momentum aktarımlı etkileşmeleri, etkileşmede çıkan parçacıkların bozunma ve dallanmalarını, saçılma tesir kesitini, ilk durum ve son durum ışımalarını, çoklu etkileşmeleri, parton dağılım fonksiyonlarını ve partonların hadronlaşması için gerekli alt programları içermektedir.



DELPHEs : Parametrik Hızlı Benzetim



DELPHEs
fast simulation

- Genel bir çarpıştırıcı deneyinin hızlı benzetimi için bir çerçeve programıdır.
- Delphes, hızlı ve çok amaçlı bir algıç yanıt benzetimi gerçekleştiren bir C++ program çerçevesidir. Benzetim manyetik alana gömülü bir izleme sistemi, kalorimetreler ve bir müon sistemi içerir. Bu çerçeve, standart dosya formatlarına (örneğin, Les Hauches Olay Dosyası (LHE) veya High Energy Monte Carlo (HepMC) bağlanır ve ayrı ayrı analizler için kullanılacak izole leptonlar, kayıp enerji ve jet toplama gibi gözlemlenebilir çıktılar üretir. Algıç yanıtının benzetimi manyetik alanın etkisini, kalorimetrelerin tanecikliğini ve alt algıç çözünürlüklerini dikkat alır. Son durum parçacıklarının görselleştirilmesi, aynı zamanda karşılık gelen ROOT kütüphanesi kullanılarak gerçekleştirilebilir.)

DELPHES

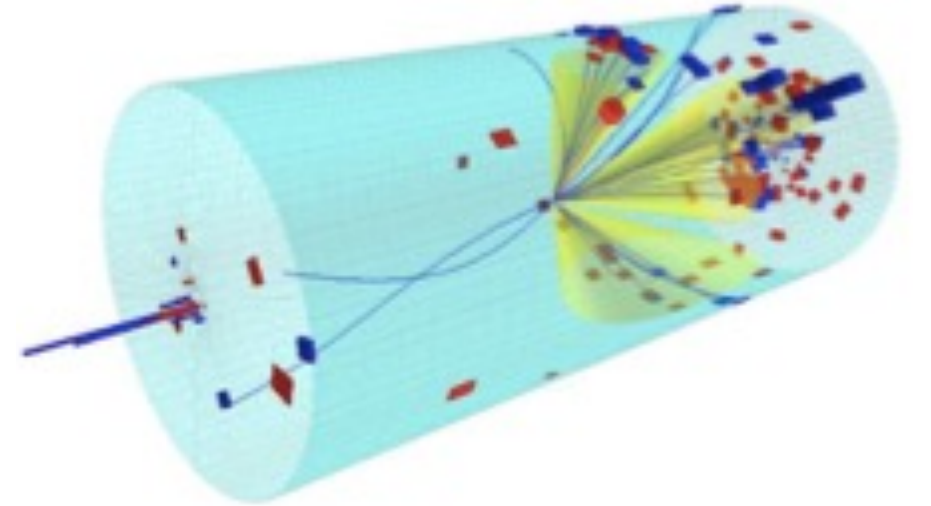
- Delphes, parametrize edilmiş bir yapısalılık içinde çok amaçlı algıçların tepkilerinin benzetimini yapabilen modüler bir çerçevedir.

NE İÇERİR?

- birikme (pile-up)
- manyetik alanda yüklü parçacık iletimi
- elektromanyetik ve hadronik kalorimetreler
- müon sistem

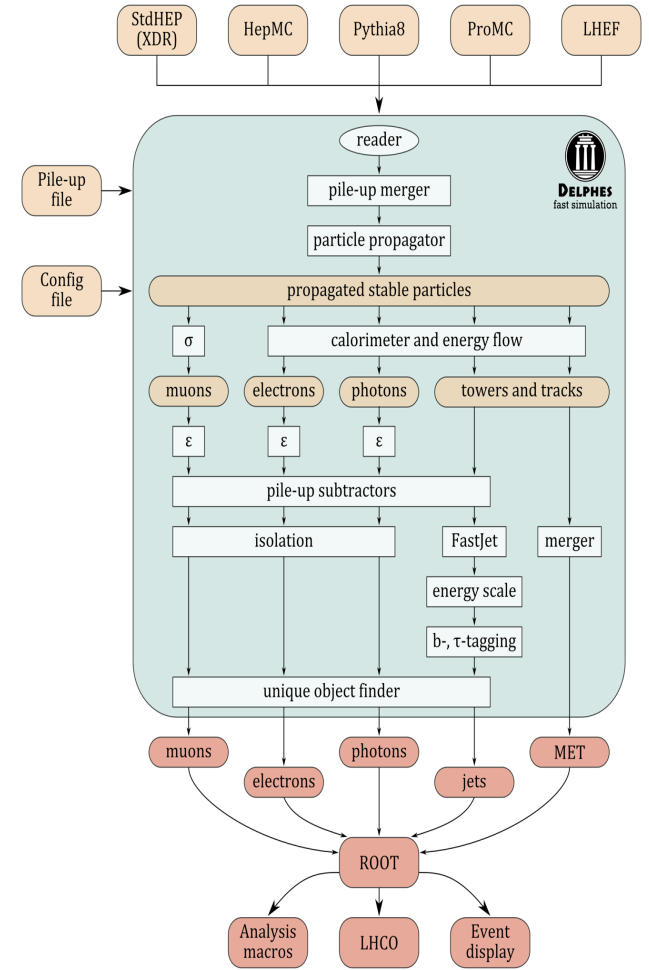
NE SAĞLAR?

- lepton (elektronlar ve müonlar)
- fotonlar
- jetler ve kayıp enine enerji (parçacık- akış)
- tau'lar ve b'ler



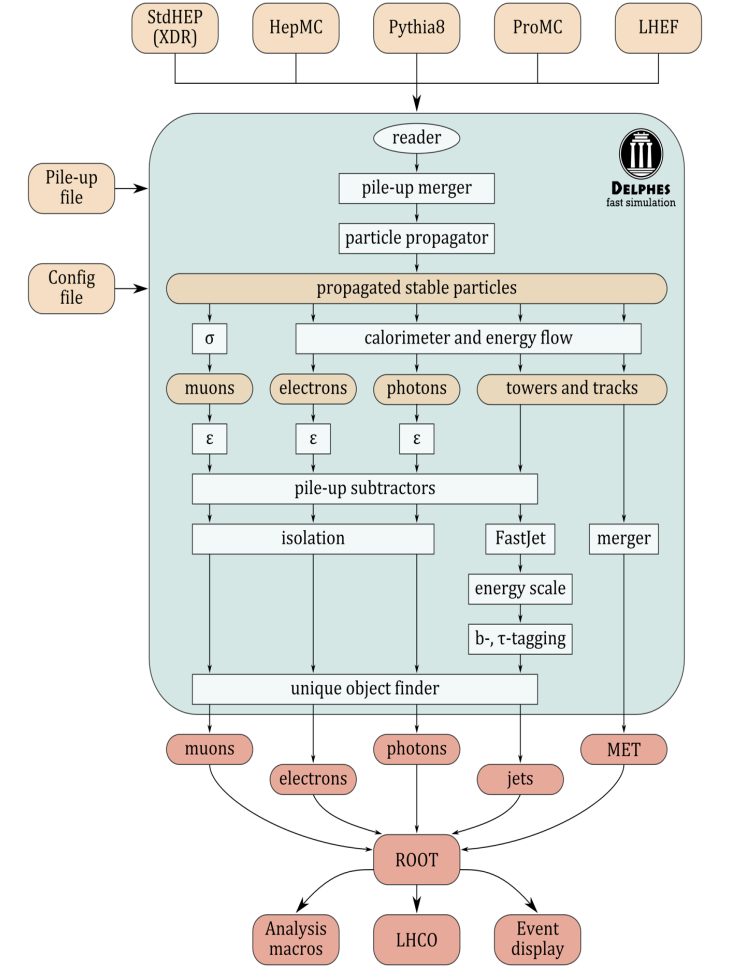
DELPHES: VERİ AKIŞ DİYAGRAMI

- Harici Monte-Carlo jeneratörlerinden gelen olay dosyaları ilk önce bir okuyucu aşama tarafından işlenir.
- Yığılma olayları daha sonra sert saçılma olayına üst üste biner.
- Uzun ömürlü parçacıklar, homojen bir manyetik alan içinde kalorimetrelere yayılır.
- Kalorimetrelere ulaşan parçacıklar enerjilerini kalorimetrelerde biriktirir.



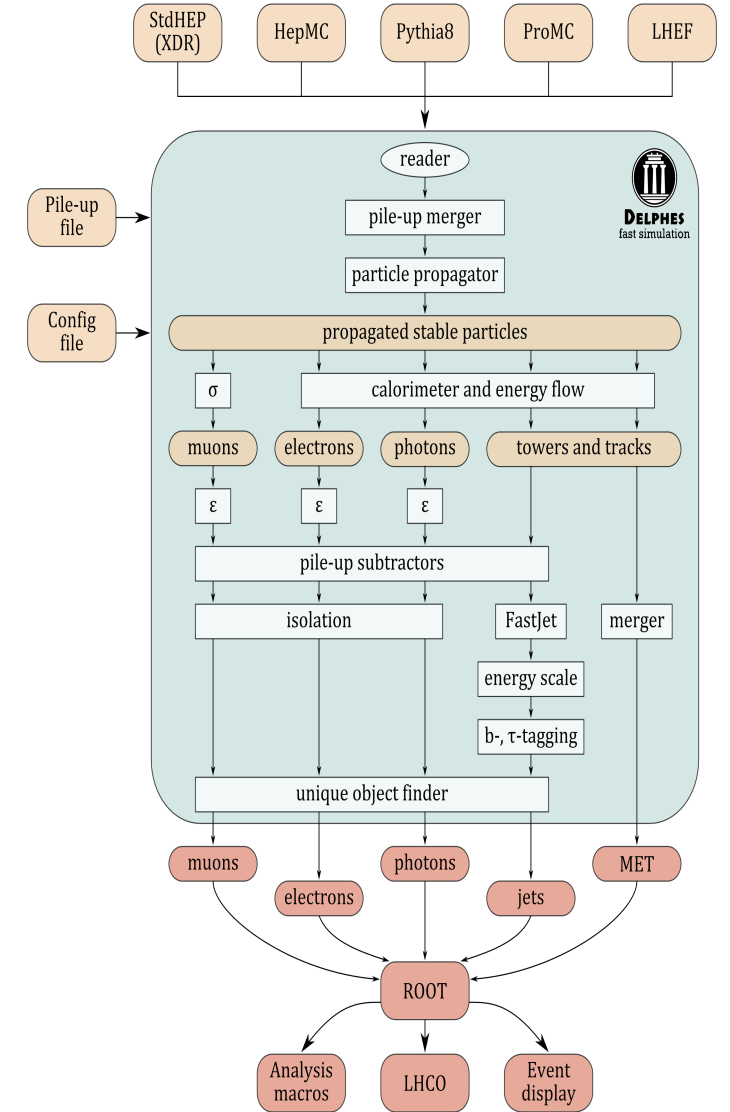
DELPHES: VERİ AKIŞ DİYAGRAMI

- Parçacık akış algoritması 4 vektörden oluşan koleksiyon üretir: elektronlar, fotonlar, parçacık akış izleri ve kuleler
- Muonlar kararlı parçacıklardan seçilir ve 4 vektörleri bulunur.
- Yığılma tepe noktalarından gelen yüklü hadronlar atılır ve kalan olay yığılma yoğunluğu hesaplanır.
- Yığılma yoğunluğu daha sonra, jetler üzerinde ve müonlar, elektronlar ve fotonlar için izolasyon parametresi üzerinde biriktirme çıkarma işlemini gerçekleştirmek için kullanılır. Eksik enerjide birikme çıkarma işlemi gerçekleştirilmez.



DELPHES: VERİ AKIŞ DİYAGRAMI

- Son aşamada, yeniden yapılandırılmış nesnelerin kopyaları kaldırılır.
- Çıktı verileri bir **ROOT** ağacı biçiminde saklanır ve **ROOT** çerçevesi yardımıyla analiz edilebilir. Kök ağaç dosyaları da LHCO dosya formatına dönüştürülebilir.
- Her adımda yapılanma dosyası tarafından kontrol edilir.



ROOT

- Yılda birkaç petabaytlık veri alan, Büyük Hadron Çarpıştırıcısı (BHÇ) deneylerinin verilerini işleme esnasında yüksek işlem verimliliği için tasarlanan ROOT, 2009 yılından itibaren yüksek enerji fiziğindeki veri analizi ve veri toplamada kullanılan en güncel ve açık kaynak kodlu bilimsel bir veri analiz paketidir.
- ROOT'un günümüzde CERN dahil birçok önemli bilimsel merkezde kullanılan temel bir veri analiz paketi olmasının nedeni:
 - ROOT'un verileri kendi formatında (.root) kaydetmesi ve bu verileri yüksek bir işlem verimliliği içinde geri yükleyerek kullanabilmesidir. Bu nedenle bir çok deneyde alınan veriler ROOT formatında alınıp işleme ROOT ile devam edilmektedir.



ROOT

- ROOT, Windows, Linux ve MacOS işletim sistemlerinde desteklenmektedir.
- ROOT kendi içinde bir C++ yorumlayıcı bulundurur.
- Derleyiciye ihtiyaç duyulmadan C/C++ kodları ile çalışılabilir.
- Etkileşimli komut satırı “root” yazarak başlatılır.
- root -l ile de başlatılabilir.

ROOT

- Bir dosyadan veri okuyup çizdirmek, histogram kavramı ve histogram çizdirmek, veri kümesine eğri uydurmak, uydurulan eğrinin parametrelerini görmek (ntuple okumak, ve içindeki bilgileri çizdirir.
- ROOT'un en önemli özelliği, ROOT'un içerisinde Tree adında, içerisinde dalları ve yaprakları olan bir veri taşıyıcı olmasıdır. Bu veri taşıyıcısı içerisine yüklenen verileri saklar ve gerektiğinde ROOT platformu üzerinde çeşitli kullanım imkanları sağlar. Bu özellik verilerin kaybolması, saklanması, bellek gereksinimleri gibi sorunların önüne
- ROOT Windows, Linux ve MacOS işletim sistemlerinde desteklenmektedir.
- ROOT parçacık fiziği veri analizi için tasarlanmış olmasına rağmen aynı zamanda astronomi ve veri madenciliği gibi diğer uygulamalarda da kullanılmaktadır.

ROOT

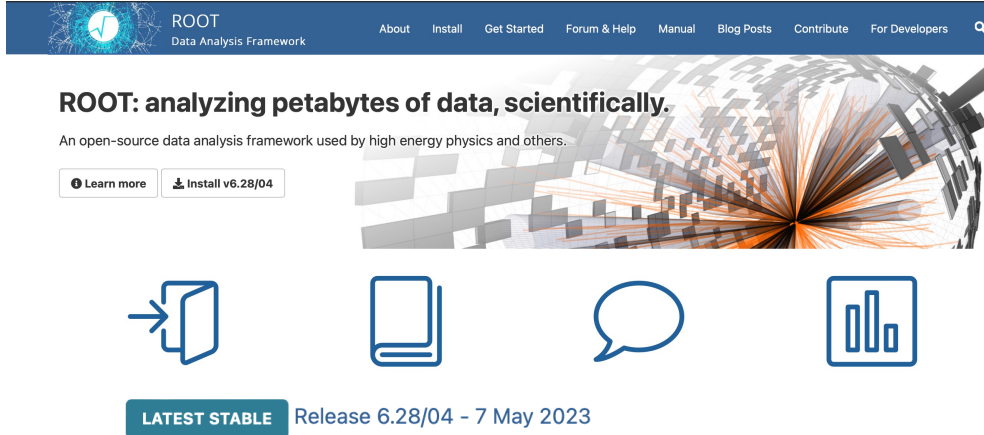
- ROOT içinde olan fakat C++'da olmayan özel komutlar bir . işareti ile birlikte yazılır.

Örnekler:

- ROOT'dan çıkmak için: .q
 - kabuk komutu kullanmak için: .!
 - ROOT'u çalıştırmak için: root veya root -l
 - bir makro yüklemek için: .L dosya adı
 - bir makro çalıştırmak için: .x dosya adı
 - yardım için: .help veya .?
-
- NOT: Makro nedir? Belli bir işi yapmak üzere her istediğinizde çalıştırabileceğiniz bir dosya içerisine yazılmış komutlar ve fonksiyonlar dizisidir.

ROOT KURULUMU

<https://root.cern> sayfasından bilgisayarınıza uygun platformu olan ROOT'u seçip indirebilirsiniz.




ROOT Data Analysis Framework

About Install Get Started Forum & Help Manual Blog Posts Contribute For Developers

ROOT: analyzing petabytes of data, scientifically.

An open-source data analysis framework used by high energy physics and others.

[Learn more](#) [Install v6.28/04](#)



LATEST STABLE Release 6.28/04 - 7 May 2023

Version 6

Release 6.28/04 - 07 May 2023

Release 6.28/02 - 23 Mar 2023

Release 6.28/00 - 04 Feb 2023

Release 6.26/10 - 16 Nov 2022

Release 6.26/06 - 29 Jul 2022

Release 6.26/04 - 07 Jun 2022

Release 6.26/02 - 13 May 2022

Release 6.10/04 - 28 Jul 2017

Release 6.10/02 - 06 Jul 2017

Release 6.10/00 - 13 Jun 2017

Release 6.09/02 - 08 Mar 2017

Release 6.08/06 - 02 Mar 2017

Release 6.08/04 - 13 Jan 2017

get the latest binaries, features and quick user support.

Download a pre-compiled binary distribution

We distribute pre-compiled ROOT for several major Linux distributions as well as MacOS and (as a beta) Windows. The steps to install a pre-compiled binary are simple:

1. Install all [required dependencies](#) with the system package manager
2. [Download the release](#) for the desired platform and ROOT version
3. Unpack the archive
4. Add the ROOT libraries and executables to your environment by sourcing the appropriate `thisroot.*` script. These setup scripts can be found in the ROOT binary release, in the `bin` directory.

For example, on Centos8, a user could execute the following bash commands to install ROOT v6.28/04, after installing all [required dependencies](#):

Binary distributions

| Platform | Files | Size |
|----------------------------|---|------|
| CentOS 8 | root_v6.28.04.Linux-centos8-x86_64-gcc8.5.tar.gz | 225M |
| Fedora 36 | root_v6.28.04.Linux-fedoraVERSION_ID=36-x86_64-gcc12.2.tar.gz | 301M |
| Ubuntu 18.04 | root_v6.28.04.Linux-ubuntu18-x86_64-gcc7.5.tar.gz | 243M |
| Ubuntu 20.04 | root_v6.28.04.Linux-ubuntu20-x86_64-gcc9.4.tar.gz | 244M |
| Ubuntu 22.04 | root_v6.28.04.Linux-ubuntu22-x86_64-gcc11.3.tar.gz | 291M |
| macOS 11.7 arm64 Xcode 12 | root_v6.28.04.macos-11.7-arm64-clang120.pkg | 344M |
| macOS 11.7 arm64 Xcode 12 | root_v6.28.04.macos-11.7-arm64-clang120.tar.gz | 223M |
| macOS 11.7 x86_64 Xcode 12 | root_v6.28.04.macos-11.7-x86_64-clang120.pkg | 359M |
| macOS 11.7 x86_64 Xcode 12 | root_v6.28.04.macos-11.7-x86_64-clang120.tar.gz | 235M |
| macOS 12.6 arm64 Xcode 14 | root_v6.28.04.macos-12.6-arm64-clang140.pkg | 341M |
| macOS 12.6 arm64 Xcode 14 | root_v6.28.04.macos-12.6-arm64-clang140.tar.gz | 223M |
| macOS 12.6 x86_64 Xcode 14 | root_v6.28.04.macos-12.6-x86_64-clang140.pkg | 355M |
| macOS 12.6 x86_64 Xcode 14 | root_v6.28.04.macos-12.6-x86_64-clang140.tar.gz | 234M |
| macOS 13.3 arm64 Xcode 14 | root_v6.28.04.macos-13.3-arm64-clang140.pkg | 354M |

MadGraph (LaunchPad)

- <https://launchpad.net/mg5amcnlo>



MadGraph5_aMC@NLO

[Log in / Regi](#)

Overview

Code

Bugs

Blueprints

Translations

Answers

Registered 2009-09-15 by  Michel Herquet

MadGraph5_aMC@NLO is a framework that aims at providing all the elements necessary for SM and BSM phenomenology, such as the computations of cross sections, the generation of hard events and their matching with event generators, and the use of a variety of tools relevant to event manipulation and analysis. Processes can be simulated to LO accuracy for any user-defined Lagrangian, and the NLO accuracy in the case of models that support this kind of calculations -- prominent among these are QCD and EW corrections to SM processes. Matrix elements at the tree- and one-loop-level can also be obtained.

MadGraph5_aMC@NLO is the new version of both MadGraph5 and aMC@NLO that unifies the LO and NLO lines of development of automated tools within the MadGraph family. It therefore supersedes all the MadGraph5 1.5.x versions and all the beta versions of aMC@NLO. As such, the code allows one to simulate processes in virtually all configurations of interest, in particular for hadronic and e+e- colliders; starting from version 3.2.0, the latter include Initial State Radiation and beamstrahlung effects.

The standard reference for the use of the code is: J. Alwall et al, "The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations", arXiv:1405.0301 [hep-ph]. In addition to that, computations in mixed-coupling expansions and/or of NLO corrections in theories other than QCD (eg NLO EW) require the citation of: R. Frederix et al, "The automation of next-to-leading order electroweak calculations", arXiv:1804.10017 [hep-ph]. A more complete list of references can be found here: http://amcatnlo.web.cern.ch/amcatnlo/list_refs.htm

Download:

Get Involved

[Report a bug](#)

[Ask a question](#)

[Register a blueprint](#)

 [Help translate](#)

Downloads

Latest version is 3.5.x

[MG5aMC_LTS_v2.9.16.tar.gz](#)

[MG5_aMC_v3.5.1.tar.gz](#)

released on 2023-05-12

MG5_aMC_v3.5.0.tar.gz indirilir. Python3 ile uyumludur.

MadGraph KURULUMU

- `tar zxvf MG5_aMC_v3.5.0.tar.gz`
- `./bin/mg5_aMC`
- `install pythia8`
- `install Delphes`

(NOT 1: Eğer burada hata verirse, (örneğin require g77 gibi), bilgisayarınız Mac ise uyumlu olan versiyona uyumlu olan gfortran doyası indirilir. Örneğin, macOS Big Sur ise gfortran 11.2 –BigSur.dmg indirdir ve `install Delphes` komutu tekrarlanır.)

NOT 2: Root Delphes'ten önce mutlaka kurulmalıdır. ROOT'un kurulumu MadGraph'dan bağımsızdır.

NOT 3: Eğer “`install Delphes`” komutu ile yine de başaramaz iseniz bir sonraki slayttaki adımları izlemeniz yeterli olacaktır.

DELPHES KURULUMU

- <https://github.com/delphes/delphes> bağlantısına tıklayıp <https://github.com/delphes/delphes.git> adresi kullanılarak

git clone <https://github.com/delphes/delphes.git>
komutu ile Delphes indirilir.

Sonra sırası ile aşağıdaki komutları çalıştırılarak kurulur.

```
mv delphes Delphes
```

```
cd Delphes
```

```
make -j4
```

```
make display
```

MadGraph KAYNAKLAR

1- <https://cp3.irmp.ucl.ac.be/projects/madgraph>

2- <https://launchpad.net/mg5amcnlo>

3-

https://indico.cern.ch/event/335370/contributions/781780/attachments/655157/900786/HPFBUO4_MadGraph_2.pdf

ROOT KAYNAKLAR

1- <http://root.cern.ch>

2- <https://root.cern/manual/>

3- <https://root-forum.cern.ch/>

4-Parçacık Fiziđi Bilgisayar Uygulamaları Okulu, İstanbul Üniversitesi, İlkay TÜRK ÇAKIR, Ders Notları

https://indico.cern.ch/event/877623/contributions/3707938/attachments/1980072/3297006/s_on_PFHUO.pdf

PYTHIA KAYNAKLAR

- 1- T. Sjostrand et al., An Introduction to PYTHIA 8.2, arXiv:1410.3012 [hep-ph], <http://home.thep.lu.se/~torbjorn/pdfdoc/pythia8200.pdf>
- 2- T. Sjostrand, S. Mrenna and P. Skands, A Brief Introduction to Pythia 8.1, arXiv:0710.3820, web sayfası:
- 3- <http://home.thep.lu.se/~torbjorn/pythia8/pythia8100.pdf>
- 4- <http://home.thep.lu.se/~torbjorn/php8157/Welcome.php>
- 5- T. Sjostrand, S. Mrenna and P. Skands, PYTHIA 6.4 Physics and Manual, hep-ph/0603175, web sayfası:
<http://www.thep.lu.se/~torbjorn/pythia/lutp0613man2.pdf>

DELPHES KAYNAKLAR

- 1- <https://www.tcm.phy.cam.ac.uk/~mjh261/pdfs/delphes.pdf>
- 2- <https://cp3.irmp.ucl.ac.be/projects/delphes>

UYGULAMA 2

```
MG5_aMC>exit
```

```
ilkay@ilkay1 MG5_aMC_v3_5_0 % ./bin/mg5_aMC
```

```
MG5_aMC_v3_5_0 — Python ◀ mg5_aMC — 118x24

lhapdf-config does not seem to correspond to a valid lhapdf-config executable.
Please set the 'lhapdf' variable to the (absolute) /PATH/TO/lhapdf-config (including lhapdf-config).
Note that you can still compile and run aMC@NLO with the built-in PDFs
MG5_aMC> set lhapdf /PATH/TO/lhapdf-config

None does not seem to correspond to a valid lhapdf-config executable.
Please set the 'lhapdf' variable to the (absolute) /PATH/TO/lhapdf-config (including lhapdf-config).
Note that you can still compile and run aMC@NLO with the built-in PDFs
MG5_aMC> set lhapdf /PATH/TO/lhapdf-config

Using default text editor "vi". Set another one in ./input/mg5_configuration.txt
Loading default model: sm
INFO: Restrict model sm with file models/sm/restrict_default.dat .
INFO: Run "set stdout_level DEBUG" before import for more information.
INFO: Change particles name to pass to MG5 convention
Defined multiparticle p = g u c d s u~ c~ d~ s~
Defined multiparticle j = g u c d s u~ c~ d~ s~
Defined multiparticle l+ = e+ mu+
Defined multiparticle l- = e- mu-
Defined multiparticle vl = ve vm vt
Defined multiparticle vl~ = ve~ vm~ vt~
Defined multiparticle all = g u c d s u~ c~ d~ s~ a ve vm vt e- mu- ve~ vm~ vt~ e+ mu+ t b t~ b~ z w+ h w- ta- ta+
MG5_aMC>
```

```

heft                loop_sm-no_widths          sm-zeromass_ckm
heft-ckm            loop_sm-parallel_test      sm/
heft-full           loop_sm-test               SMScalars
heft-no_b_mass      loop_sm-zeromass_ckm      SMScalars-full
heft-no_masses      loop_sm/                   SMScalars/
heft-no_tau_mass    MSSM_SLHA2-full           taudecay_UFO/
heft-zeromass_ckm  MSSM_SLHA2-no_b_mass      TopEffTh
hgg_plugin/         MSSM_SLHA2-no_masses      TopEffTh-
loop_qcd_qed_sm     MSSM_SLHA2-no_tau_mass    triplet_diquarks
loop_qcd_qed_sm-full MSSM_SLHA2/                triplet_diquarks-
loop_qcd_qed_sm-no_widths nmssm                       uutt_sch_4fermion
loop_qcd_qed_sm-with_b_mass nmssm-full                 uutt_sch_4fermion-
loop_qcd_qed_sm-with_b_mass_no_widths RS                          uutt_tch_scalar
loop_qcd_qed_sm_Gmu RS-                          uutt_tch_scalar-
loop_qcd_qed_sm_Gmu-ckm sextet_diquarks            VLQ_v4_4FNS_UFO-3rd
loop_qcd_qed_sm_Gmu-full sextet_diquarks-          VLQ_v4_4FNS_UFO-3rdL
loop_qcd_qed_sm_Gmu-no_widths sm-c_mass                 VLQ_v4_4FNS_UFO-3rdL_noCKM
loop_sm-c_mass      sm-ckm                    VLQ_v4_4FNS_UFO/
loop_sm-ckm         sm-full
MG5_aMC> import model sm-full
INFO: Change particles name to pass to MG5 convention
Kept definitions of multiparticles p / j / l+ / l- / vl / vl~ unchanged
Defined multiparticle all = g u d s u~ d~ s~ a ve vm vt ve~ vm~ vt~ c t b c~ t~ b~ z w+ h w- e- mu- ta- e+ mu+ ta+
MG5_aMC>

```



```
MG5_aMC>generate e+ e- > e+ e-
```

```
INFO: Checking for minimal orders which gives processes.
```

```
INFO: Please specify coupling orders to bypass this step.
```

```
INFO: Trying process: e+ e- > e+ e- WEIGHTED<=4 @1
```

```
INFO: Process has 6 diagrams
```

```
1 processes with 6 diagrams generated in 0.012 s
```

```
Total: 1 processes with 6 diagrams
```

```
MG5_aMC>output ilkay
```

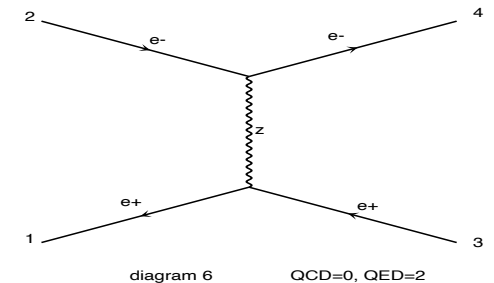
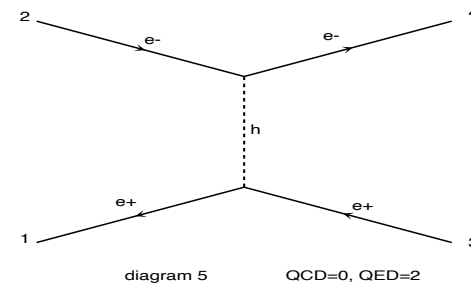
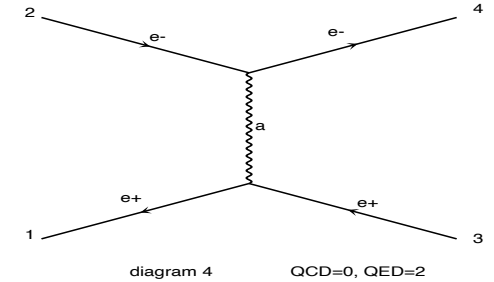
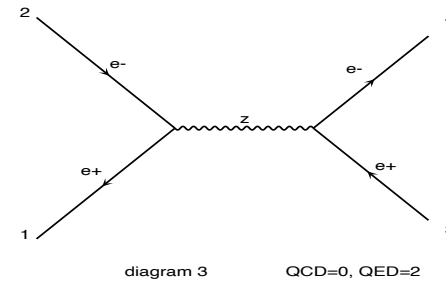
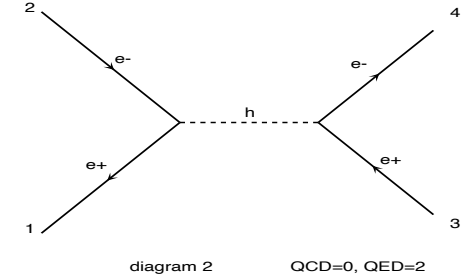
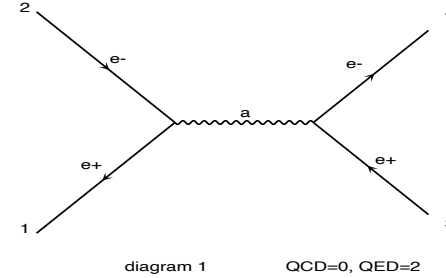
```
>y
INFO: initialize a new directory: ilkay
INFO: remove old information in ilkay
INFO: Organizing processes into subprocess groups
INFO: Generating Helas calls for process: e+ e- > e+ e- WEIGHTED<=4 @1
INFO: Processing color information for process: e+ e- > e+ e- @1
INFO: Creating files in directory P1_epem_epem
INFO: Some T-channel width have been set to zero [new since 2.8.0]
if you want to keep this width please set "zerowidth_tchannel" to False
INFO: Generating Feynman diagrams for Process: e+ e- > e+ e- WEIGHTED<=4 @1
INFO: Finding symmetric diagrams for subprocess group epem_epem
Generated helas calls for 1 subprocesses (6 diagrams) in 0.010 s
Wrote files for 16 helas calls in 0.127 s
ALPHA: aloha starts to compute helicity amplitudes
ALPHA: aloha creates 4 routines in 2.785 s
save configuration file to /Users/ilkay/MG5_aMC_v3_5_0/ilkay/Cards/me5_conf
INFO: Use Fortran compiler gfortran
INFO: Use c++ compiler clang
INFO: Generate web pages
Output to directory /Users/ilkay/MG5_aMC_v3_5_0/ilkay done.
Type "launch" to generate events from this process, or see
/Users/ilkay/MG5_aMC_v3_5_0/ilkay/README
Run "open index.html" to see more information about this process.
MG5_aMC>
```

Total: 1 processes with 6 diagrams

```
MG5_aMC>display diagrams
```

e+ e- > e+ e- WEIGHTED=4

page 1/1



Run "open index.html" to see more information about this process.

MG5_aMC> launch

The following switches determine which programs are run:

```

=====
| 1. Choose the shower/hadronization program      shower = OFF
| 2. Choose the detector simulation program       detector = OFF
| 3. Choose an analysis package (plot/convert)    analysis = Not Avail.
| 4. Decay onshell particles                      madspin = OFF
| 5. Add weights to events for new hypp.         reweight = OFF
=====

```

Either type the switch number (1 to 5) to change its setting,
Set any switch explicitly (e.g. type 'shower=Pythia8' at the prompt)
Type 'help' for the list of all valid option

Type '0', 'auto', 'done' or just press enter when you are done [40s to answer]
ilkay@Ilkays-MacBook-Pro ~ % cd MG5_aMC_v3_5_0/il/Events/run_01

NOT: Bu aşamada hızlı hareket etmek gerekiyor, çünkü beklendiği zaman otomatik olarak seçim yapıyor.

The following switches determine which programs are run:

```

=====
| 1. Choose the shower/hadronization program      shower = Pythia8
| 2. Choose the detector simulation program       detector = Delphes
| 3. Choose an analysis package (plot/convert)    analysis = Not Avail.
| 4. Decay onshell particles                      madspin = OFF
| 5. Add weights to events for new hypp.         reweight = OFF
=====

```

Either type the switch number (1 to 5) to change its setting,
Set any switch explicitly (e.g. type 'shower=OFF' at the prompt)
Type 'help' for the list of all valid option
Type '0', 'auto', 'done' or just press enter when you are done.

- 1
- 2
- 0

Do you want to edit a card (press enter to bypass editing)?

```

-----
| 1. param   : param_card.dat
| 2. run     : run_card.dat
| 3. pythia8 : pythia8_card.dat
| 4. delphes : delphes_card.dat
|-----

```

you can also

- enter the path to a valid card or banner.
- use the 'set' command to modify a parameter directly.
The set option works only for param_card and run_card.
Type 'help set' for more information on this command.
- call an external program (ASperGE/MadWidth/...).
Type 'help' for the list of available command

[0, done, 1, param, 2, run, 3, pythia8, 4, enter path, ...][90s to answer]

>

- 2 (run_card.dat dosyasına girer). Burada i tuşuna basıp gerekli değişiklikler yapılır ve esc tuşuna basılır ve :x kullanılır ise kayıt eder.
- Eğer kayıt etmek istemez isek :q komutu verilir.

```

# Collider type and energy *
# lpp: 0=No PDF, 1=proton, -1=antiproton, *
#           2=elastic photon of proton/ion beam *
#           +/-3=PDF of electron/positron beam *
#           +/-4=PDF of muon/antimuon beam *
#*****
#           0           = lpp1      ! beam 1 type
#           0           = lpp2      ! beam 2 type
#           45.0        = ebeam1    ! beam 1 total energy in GeV
#           45.0        = ebeam2    ! beam 2 total energy in GeV
#*****

```

0 yazılır ise artık sonuçlar alınır:

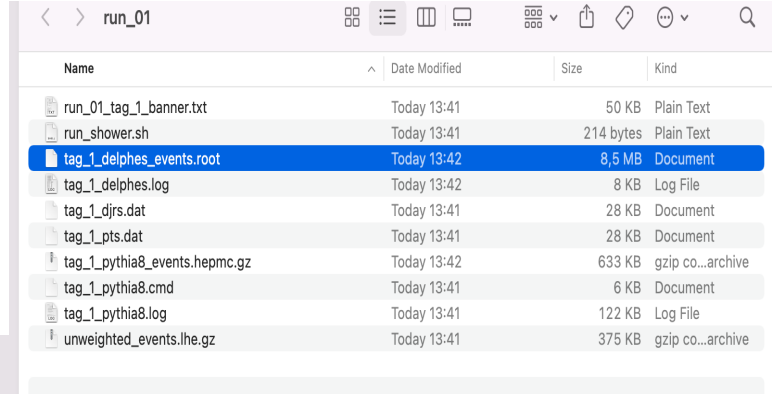


Results in the sm-full for e+ e- > e+ e-

Available Results

| Run | Collider | Banner | Cross section (pb) | Events | Data | Output | Action |
|--------|--------------------------|-----------------------|----------------------------|--------|-----------------|------------------------------|---------------------------------------|
| run_01 | e+ e- 45.0 x 45.0 GeV | tag_1 | 4185 ± 4.1 | 10000 | parton madevent | LHE | remove run launch detector simulation |
| | | | | | pythia8 | LOG HEPMC | remove run launch detector simulation |
| | | | | | delphes | LOG rootfile | remove run |

Main Page



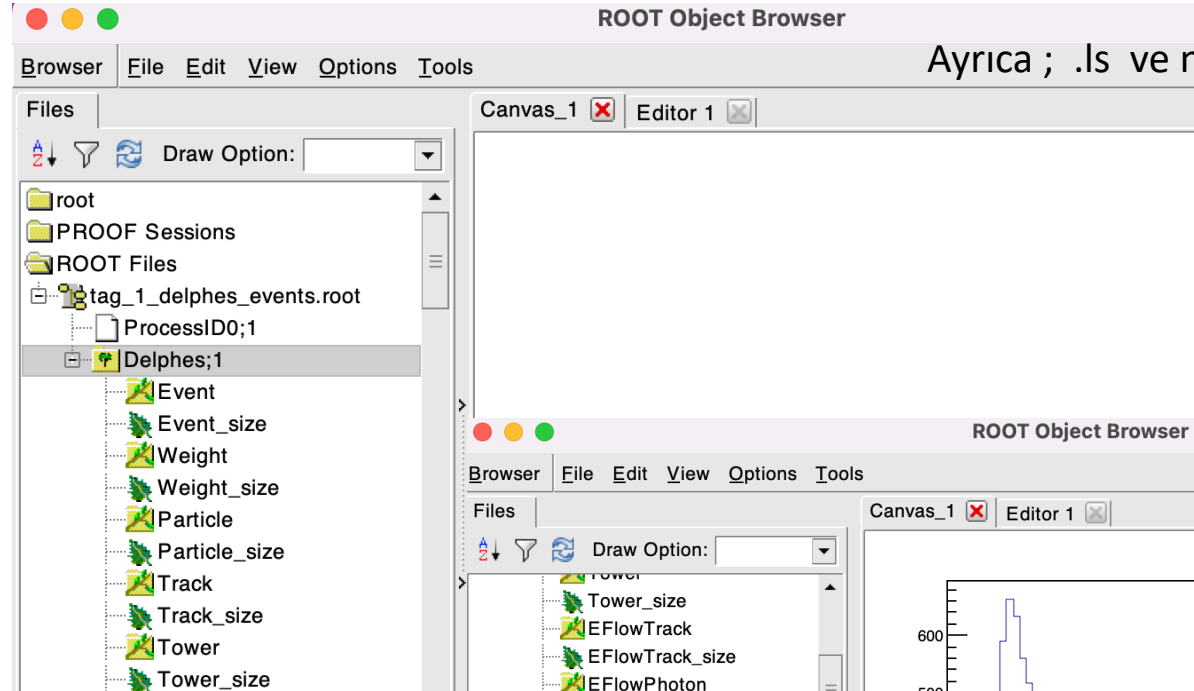
```
ilkay@Ilkays-MacBook-Pro ~ % cd MG5_aMC_v3_5_0/ilkson/Events/run_01
ilkay@Ilkays-MacBook-Pro run_01 % ls
run_01_tag_1_banner.txt      tag_1_pts.dat
run_shower.sh               tag_1_pythia8.cmd
tag_1_delphes.log           tag_1_pythia8.log
tag_1_delphes_events.root   tag_1_pythia8_events.hepmc.gz
tag_1_djrs.dat              unweighted_events.lhe.gz
ilkay@Ilkays-MacBook-Pro run_01 %
```

ROOT İLE ÇALIŞMA

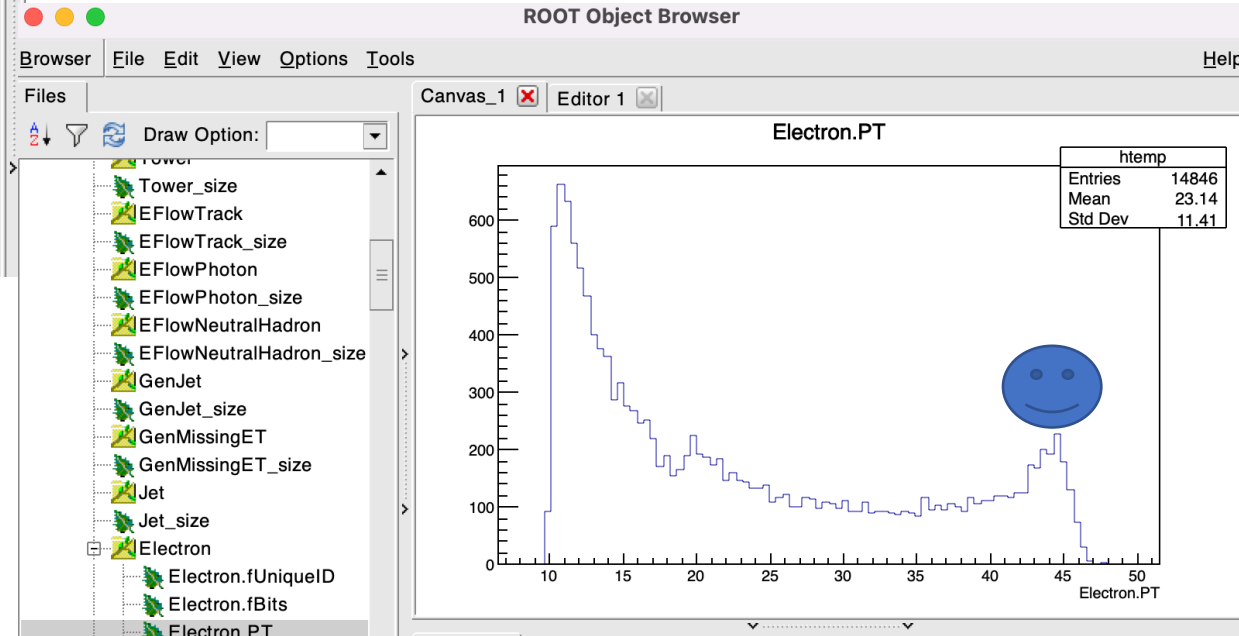
```
ilkay@Ilkays-MacBook-Pro run_01 % root -l tag_1_delphes_events.root
```

- root filenin içinde delphes tree var.
- Delphes dedektör sim. programı Delphes içindeki tree'nin adına Delphes denmiş

```
root [1] lBr  
TBranch  
TBranchCacheInfo  
TBranchClones  
TBranchDescriptor  
TBranchElement  
TBranchObject  
TBranchProxy  
TBranchProxyClassDescriptor  
TBranchProxyDescriptor  
TBranchProxyDirector  
TBranchProxyHelper  
TBranchRef  
TBranchSTL  
TBrowser  
TBrowserImp  
TBrowserPlugin  
root [1] TBrowser i
```



Ayrıca ; .ls ve new Tbrowser komutları da kullanılır.



SON

- Yaz okulu için emeđi geen tüm komitelere ve herkese teŖekkürler.
- “Yüksek Enerji Fiziđinde Kullanılan Yazılımlar ve Uygulamalar” için en az iki ders süre ayrılmalıdır.
- Bu ders notlarında Yüksek Enerji Fiziđinde kullanılan yazılımlar ve uygulamaları hakkında genel bilgiler verilmiŖtir. Bu yazılımlarda yapılabilecek en basit örnekler sunulmuŖtur. Yazılımlar ile ilgili ayrıntılı ve ileri düzeydeki bilgileri önceden hazırlamıŖ olduđum ders notlarında kaynaklar kısımlarından ulaşabilirsiniz.