



CALYPSO

Orhan akır
Ankara niversitesi

Hızlandırıcı ve Paracık Fiziğinde Bilgisayar Uygulamaları, 26-30 Ocak 2009, .., Adana

İÇERİK

1

CALYPSO tanıtım

2

CALYPSO altprogramlar

3

CALYPSO kurulum

4

CALYPSO spektrumlar

5

CALYPSO örnekler

CALYPSO* TANITIM

- ❖ CALYPSO, ışıklık spektrumunu olay üreticilerine aktarmaya yarayan bir program kütüphanesidir. Yüksek enerjili lineer çarpıştırıcılarda **demet enerji yayılması** ve **demet ışınması** ışıklık spektrumu üzerinde önemli bir etkiye sahiptir. Bu etki gelecekte kurulacak bu tür çarpıştırıcılarda fizik deneyleri için önemli olacaktır. CALYPSO beş altprogramdan oluşur,
- ❖ CALYPO ve CALYP1: spektrumu başlatmak için kullanılır, CALYPO formatlı biçimde dosyayı okur, CALYP1 ise formatsız okuma yapmaktadır.
- ❖ CALYPI: formatlı spektrum dosyasını, formatsız-ikili (binary) dosya şeklinde kopyalar.
- ❖ CALYPL: ışıklık değerini info dosyasından okumaktadır.
- ❖ CALYP: verilerden bir çarpışmayı çıkarmak için kullanılır, çarpışan iki parçacığın enerjilerini, kabul edilen olay oranını ve boyuna konumu verecektir.

*D. Schulte, CALYPSO,

<http://dschulte.web.cern.ch/dschulte/physics/calypso/calypso.html>

CALYPSO ALTPROGRAMLARI

CALYP0

CALYP1

CALYPL

CALYP

CALYPI

CALYP0

NAME: ışınlik dosyası adı
T1, T2: e- veya e+ veya gamma
Ecm: min. kütle merkezi enerjisi (GeV)
E1, E2: çarpışan parçacık enerjileri
R: kabul edilen olayların kesri

CALYP0 (NAME,T1,T2,Ecm,E1,E2,R)
CHARACTER*(*) NAME,T1,T2
REAL Ecm,E1,E2,R

CALYP1

CALYP0 ile aynı değişken ve tanımlara sahip

CALYPL

NAME: ışınlik dosyası adı
T1, T2: e- veya e+ veya gamma
RES: ışınlik değeri ($\text{cm}^{-2}\text{s}^{-1}$)

CALYPL (NAME,T1,T2,RES)
CHARACTER*(*) NAME,T1,T2
DOUBLE PRECISION RES

CALYP

E1, E2: çarpışan parçacık enerjileri
Z: boyuna konum (mm)
RN: $0 \leq R < 1$ rastgele sayı

CALYP (E1,E2,Z,RN)
REAL E1,E2,Z
DOUBLE PRECISION RN

CALYPI

NAME1: formatlı dosya adı
NAME2: formatsız dosya adı

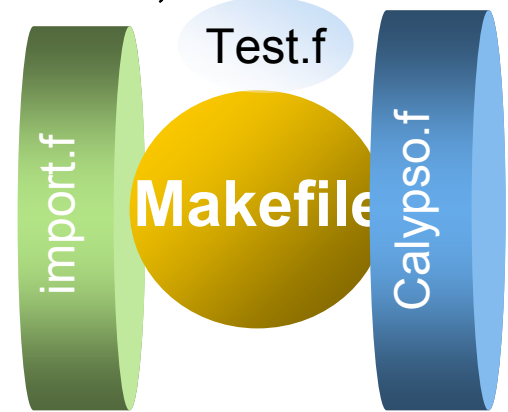
CALYPI (NAME1,NAME2)
CHARACTER*(*) NAME1,NAME2

❖ Kurulum

CALYPSO'yu kurmak ve kullanmak için calypso.tar.gz programını, <http://dschulte.web.cern.ch/dschulte/physics/calypso/> adresinden alabilir, veya Okul'un program deposundan alabiliriz, "/home/kullanici/calypso" dizininde program, **tar zxvf calypso.tar.gz** komutlarıyla açılır.

❖ Kullanıcı kitapçığını

<http://dschulte.web.cern.ch/dschulte/physics/calypso/calypso.ps.gz> adresinden alabiliriz.



❖ Ayarlar

- Makefile içinde,
 - `PYTHIA_PATH = /home/user/pythia/lib`
 - `PYTHIA_LIB = pythia6409`
 - `-lpdflib804`
- test.f dosyası içinde,
 - `CALL CALYP1("3MOM",...)`

❖ Girdi spektrum dosyaları

- "lumi.ep"
- "lumi.info"

Örnek spektrum dosyaları :

LC 500 GeV

- S_500.ep
- S_500.info

LC 3000 GeV

- S_3000.ep
- S_3000.info

❖ **calypso.f**

- CALYPSO altprogramlarını içeren dosya

❖ **import.f**

- Formatlı veri dosyasını, formatsız dosyaya dönüştürür; ikinci dosya daha küçük boyutludur ve bunu okuma hızı daha yüksektir.

❖ **test.f**

- Bir örnek programdır.

❖ **makefile**

- calypso.o nesne dosyasını, calypso import ve calypso test çalıştırılabilir program dosyaları oluşturur. Burada calypso_import formatlı dosyaların ikili (binary) kopyalarını oluşturur, calypso_test ise örnek calypso programıdır.

Hepsini aynı anda derlemek için **make all** yazılmalıdır.

❖ **PYTHIA Derlenmesi**

- `g77 -c pythia6409.f -c pythia6409.o`
- `ar rv libpythia6409.a pythia6409.o`
- `ranlib libpythia6409.a`

❖ **CALYPSO Derlenmesi (cp test1.f test.f)**

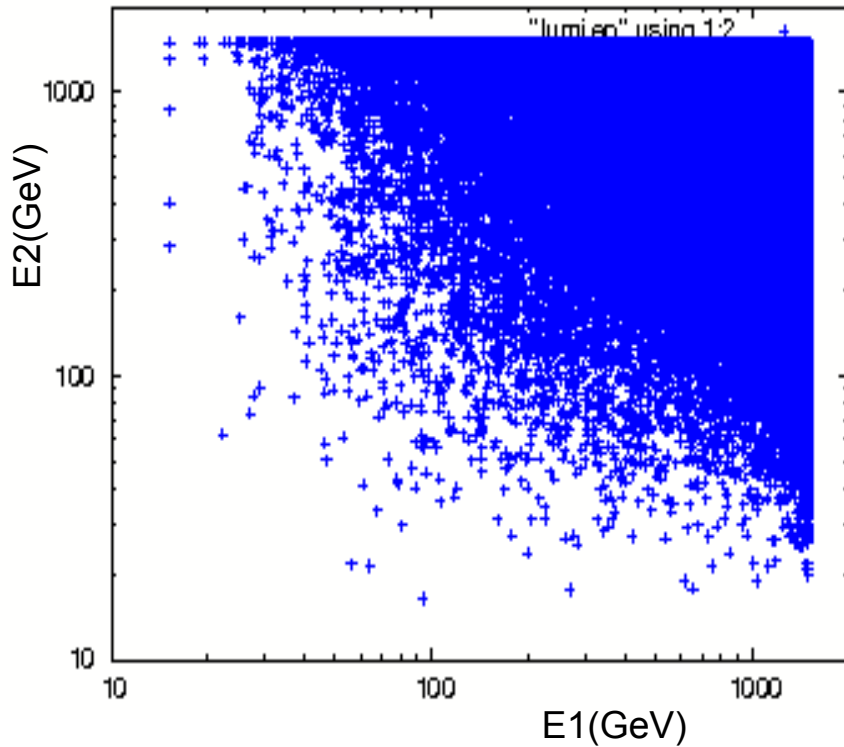
- `./make all`

❖ **Sonuçta çalıştırılabilir dosyalar**

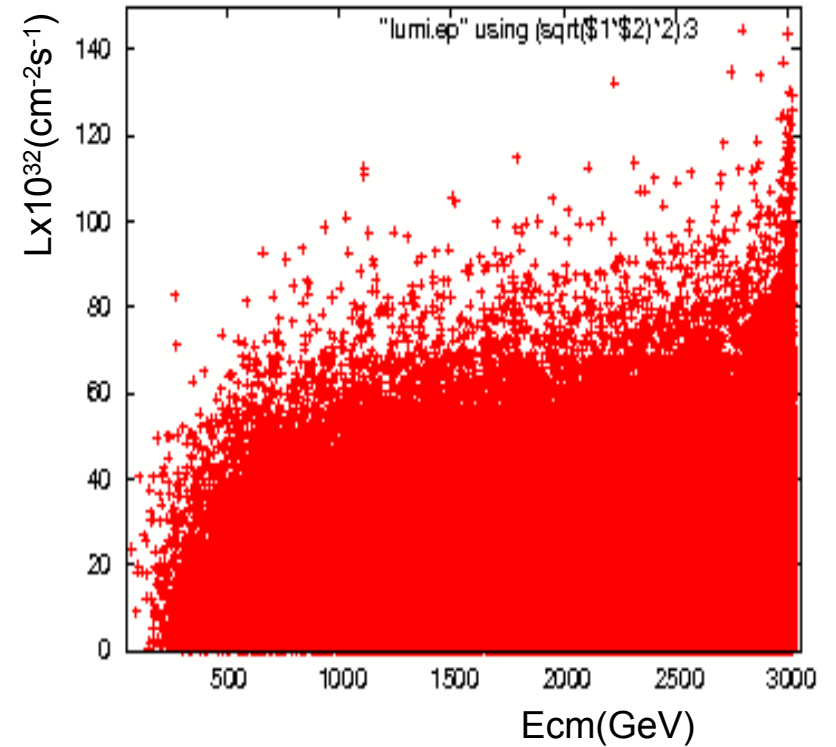
- `calypso_test`
- `calypso_import`

CALYPSO SPEKTRUMLAR

❖ Enerji Spektrumu(E=3 TeV)



❖ Işınlık Spektrumu(E=3 TeV)



❖ Örnek-1

- e⁺e⁻ çarpışmasında oluşturulan örnek ışınlık dosyası (lumi.ep) okunur, ilk 100 çarpışma için elektron ve pozitronun enerjileri ekrana yazılır.

test1.f

```
Program test1
Double precision r,rand
Call calyp0('lumi','e-','e+',0.0,e1,e2,r)
Do i=1,100
Rand=Rndm(i)
Call calyp(e1,e2,z,rand)
Write(*,*)e1,e2
Enddo
end
```

```
function rndm(i)
Double precision pyr,rndm
Rndm=pyr(0)
End
```

```
cp test.f test_orig.f
cp test1.f test.f
make all
./calypso_test
```



<u>E1</u>	<u>E2</u>
...	...
...	...
...	...

❖ Örnek-2

- Formatlı spektrum dosyasından (lumi.ep), ikili (binary) dosya (lumi.ep.bin) oluşturulması

```
>make all
>./calypso_import
*****
CALYPSO Version 0.1
CALYPSO import utility
*****
Enter the file name to be imported
lumi
Enter the file name for the export
"lumi"
Importing file
lumi.ep
Found 9226 event records
...
```

lumi.ep
(19k)



lumi.ep.bin
(9k)

CALYPSO ÖRNEKLER-3

Örnek-3

- Işınlık spektrumu verisi ve bilgisi içeren dosyalar (lumi.ep ve lumi.info) kullanılarak değişken enerjili e+e- çarpışması yaptırılır, seçilen süreçte Z-bozonu benzeri bir Z' bozonu üretilir. Demetlerin çarpışmaya hazırlanmasından sonra CALYP altprogramının her çağrılışında çarpışan iki parçacığın enerjileri elde edilir ve bunlara göre olay üretimi yapılır.

test2.f

```
Program test2
IMPLICIT DOUBLE PRECISION(A-H, O-Z)
REAL E1,E2,Z,R
COMMON/PYJETS/N,NPAD,K(4000,5),P(4000,5),V(4000,5)
COMMON/PYSUBS/MSEL,MSELPD,MSUB(500),KFIN(2,-40:40),CKIN(200)
COMMON/PYPARS/MSTP(200),PARP(200),MSTI(200),PARI(200)
COMMON/PYDAT2/KCHG(500,4),PMAS(500,4),PARF(2000),VCKM(4,4)
EXTERNAL PYDATA,PYDAT1,PYDAT3
CALL CALYPL('lumi','e-','e+',RES)
CALL CALYP1('lumi','e-','e+',2400.0,E1,E2,R)
ECM=2.0*MAX(E1,E2)
MSEL=21
PMAS(32,1)=1488.42*2
ECM=PMAS(32,1)
MSTP(44)=3
MSTP(121)=1
PARP(121)=2.5D0
MSTP(11)=1
MSTP(171)=1
DO 100 I=1,2
DO 100 J=1,5
100 P(I,J)=0D0
P(1,3)=E1
P(2,3)=-E2
CALL PYINIT('3MOM','e-','e+',ECM)
open(11,file='en.data')
NGEN=0
NCALL=0
```

```
110 DO 130 I=1,2
DO 120 J=1,5
P(I,J)=0D0
V(I,J)=0D0
120 CONTINUE
130 CONTINUE
140 RAND=RNDM(E1)
CALL CALYP(E1,E2,Z,RAND)
NCALL=NCALL+1
P(1,3)=E1
P(2,3)=-E2
CALL PYEVNT
IF(MSTI(61).EQ.1) GOTO 110
CALL PYEDIT(2)
SUM=0.0
DO 150,I=1,N
pt=sqrt(p(i,1)**2+p(i,2)**2)
theta=atan2(pt,abs(p(i,3)))
if (theta.gt.0.12) then
sum=sum+p(i,4)
endif
150 continue
write(11,*) sum
NGEN=NGEN+1
IF(NGEN.LT.10000) GOTO 110
CALL PYSTAT(1)
END
```

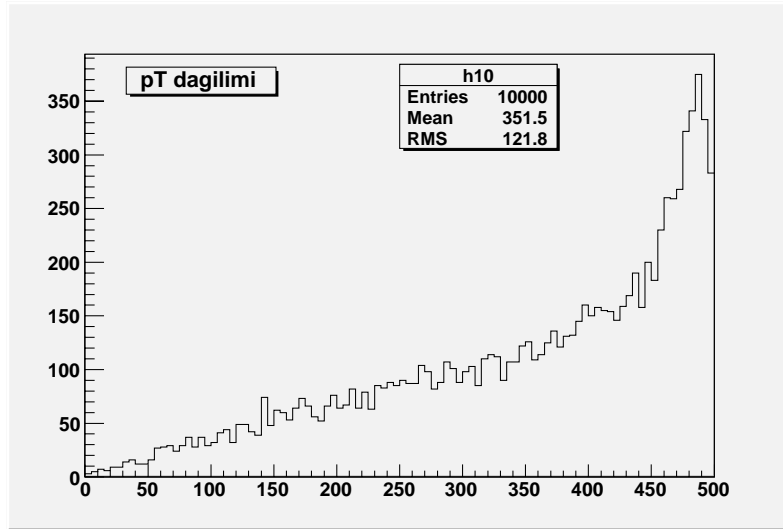
```
function rndm(i)
Double precision pyr,rndm
Rndm=pyr(0)
End
```

```
cp test2.f test.f
make all
./calypso_test
```

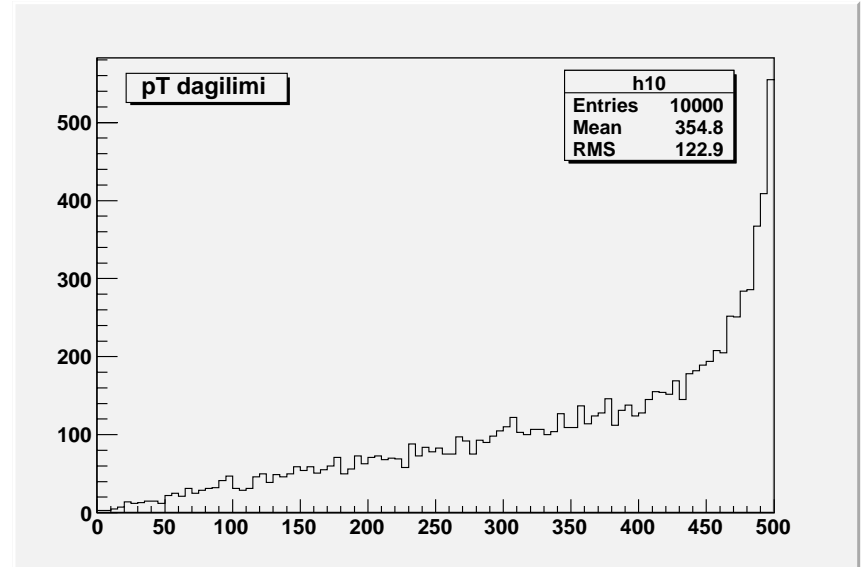
CALYPSO ÖRNEKLER-4

❖ Muon pT dağılımı(spektrumlu)

$Z' \rightarrow \mu^+ \mu^-$ @ $\sqrt{s}=1000$ GeV,
 $m_{Z'}=980$ GeV



❖ Spektrumsuz grafik



CALYPSO DAHASI VAR !

- ❖ CALYPSO için gerekli ışınlık spektrumu dosyalarını üretmek için,
 - **GUINEA-PIG** kurulmalı ve ek olarak aşağıdaki betikler kullanılmalıdır,
 - **run.tcl**: önce düşük çözünürlükte GUINEA-PIG çalıştırarak makinanın ışınlığını tahmin edilir, sonraki adımda parametreler ayarlanarak yüksek çözünürlükte 10 kez GUINEA-PIG çalıştırılır. Sonuçta her bir çalıştırma için lepton-lepton, lepton-foton, foton-lepton ve foton-foton çarpışmalarını içeren dosyalar üretilir.
 - çalıştırmak için “tclsh run.tcl”
 - **lumi_extr.tcl**: 4x10 dosya; elektron-elektron, elektron-pozitron, pozitron-elektron, pozitron-pozitron, foton-elektron, elektron-foton, foton-pozitron, pozitron-foton, foton-foton çarpışmaları için 9 dosyada birleştirilir. Bu dosyalar CALYPSO için formatlı girdi dosyalarıdır.

CALYPSO ÖDEVLER

1. Verilen lumi.ep dosyasından ışınlık değerini, boyuna uzunluğu (z) veren test.f programı yazınız ve çalıştırınız, (zorunlu).
2. Verilen lumi.ep.bin dosyasından ışınlığı ve E1,E2 enerjilerini okuyarak kütle merkezi enerjisini hesaplayıp çıktıya $E_{cm}(\text{GeV})$ ve $L(\text{cm}^{-2}\text{s}^{-1})$ verilerini (iki sütunlu) aktaran bir programı yazınız, (zorunlu).
3. calypso.f programını derleyerek nesne dosyası (.o) oluşturunuz, bunu arşiv dosyası (.a) haline getirin ve PYTHIA C++ test programı içinden çağırabilecek şekilde örnek bir testc.cc programı yazınız, (seçmeli).



Sorular ?