

GUINEA-PIG++

Orhan ÇAKIR
Ankara Üniversitesi

Guinea-Pig++

Guinea-Pig, yüksek enerjili e^+e^- çarpıştırıcılarında demet-demet etkileşmelerinin benzetimi için kullanılan, D. Schulte tarafından C dilinde yazılmış (ilk sürüm) bir programdır. Programın gelişimi son zamanlarda **Guinea-Pig++** projesi olarak devam etmektedir. Program, elektromagnetik kıştırma (pinch) ve (disruption) etkileri, demet ışıması ve düşük enerjili parçacıkların (parçacık çiftleri, hadronlar) arkaplan oluşturmalarına yol açan ikincil etkileşmeleri içerir.

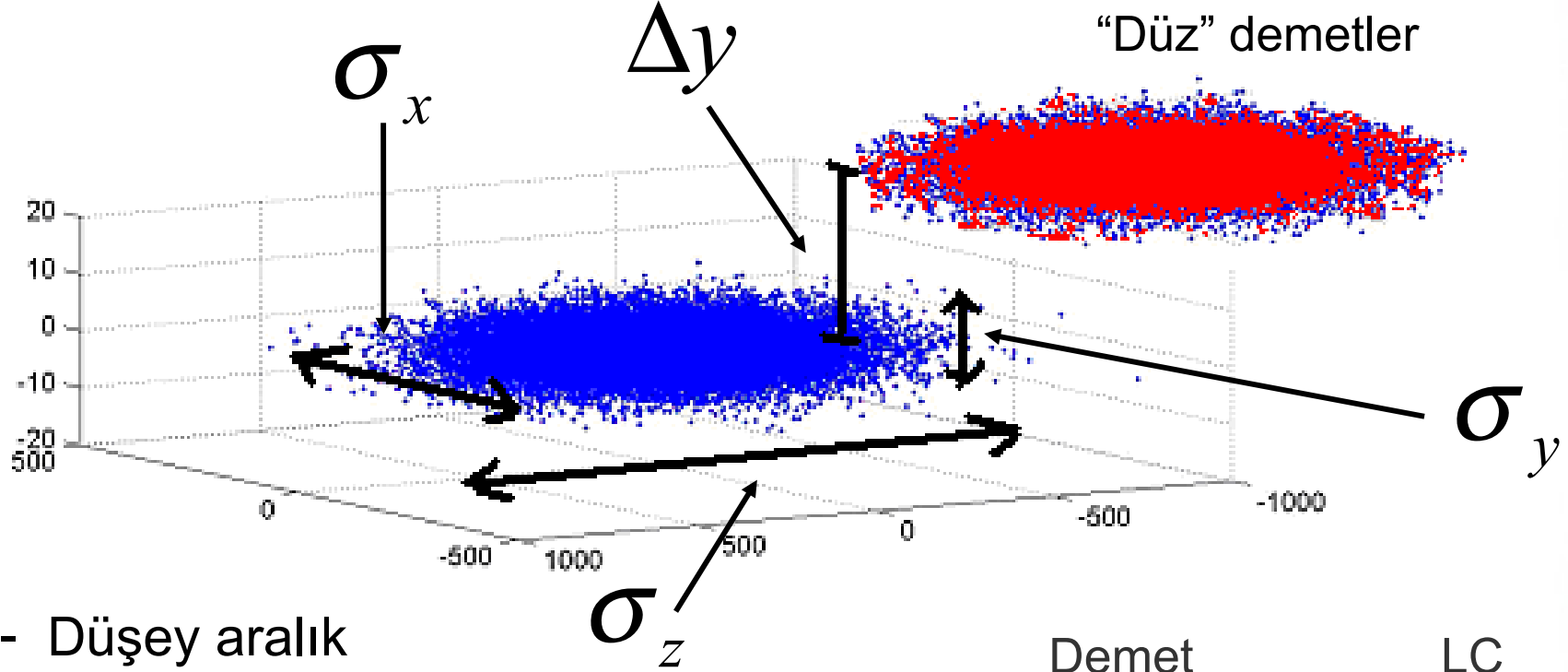
Bu program gelecek doğrusal çarpıştırıcılarda Ar-Ge çalışmaları için standart/temel bir araçtır.

Amaç

- Çarpışma bölgesinde demet-demet etkileşmesi benzetimini yapmak
- Işınlık / enerji spektrumları / demet profili grafiklerini çizmek
- Olay üretimi için spektrum dosyaları oluşturmak
- Fizik gereksinimi için en iyi demet parametrelerini bulmak

Demet Parametreleri

* Demetlerin Etkileşme Bölgesi



Δy - Düşey aralık

σ_x - Demet boyutu x kesiti

σ_y - Demet boyutu y kesiti

σ_z - Paket boyu

O.Cakir

GP++@HPFBU14

Demet Parametreleri	LC
Enerji	500 GeV
Parçacık sayısı	7.5×10^9
σ_x	243 nm
σ_y	3 nm
σ_z	110 μm

Etkileşme Bölgesinde Önemli Parametreler

* **Anlık Işınlık (cm^2s^{-1}):**

$$L = \frac{f_{rep} n_b N^2}{A} H_D$$

$f_{rep} n_b N^2$ | Mevcut güce ve hızlandırıcı teknolojisine bağlı

n_b = paket sayısı / katar

N = paketteki parçacık sayısı

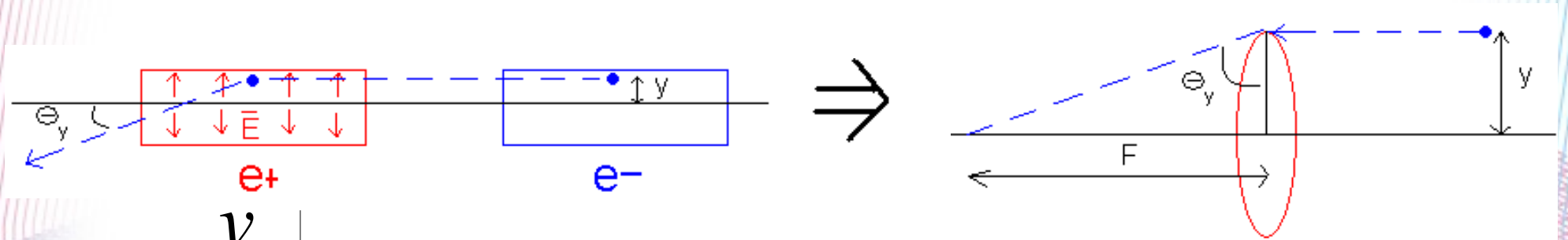
f_{rep} = tekrarlama frekansı

A = IP de demet enine kesiti

H_D = Işınlık kazanç çarpanı

$\frac{H_D}{A}$ | Demet odaklama ve demet-demet etkilerine bağlı

* **Sapma Açısı (urad):**



$$\theta_y = \frac{y}{F}$$

y = düşey konum
 F = odak uzunluğu

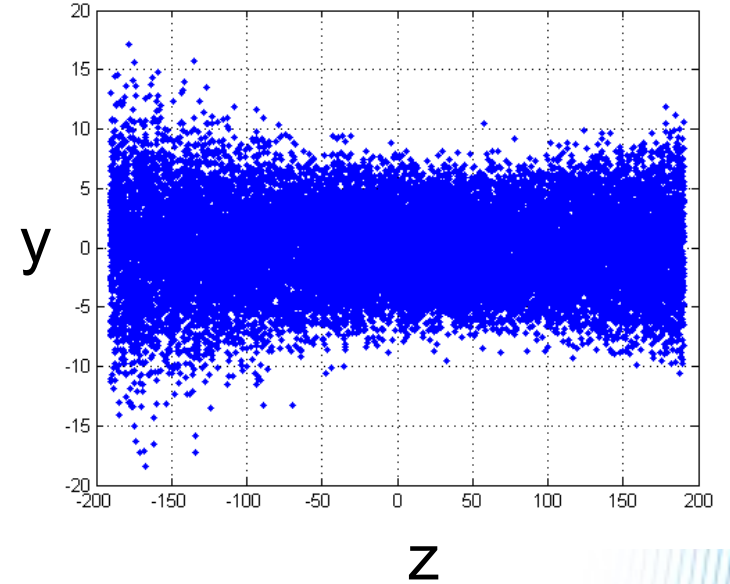
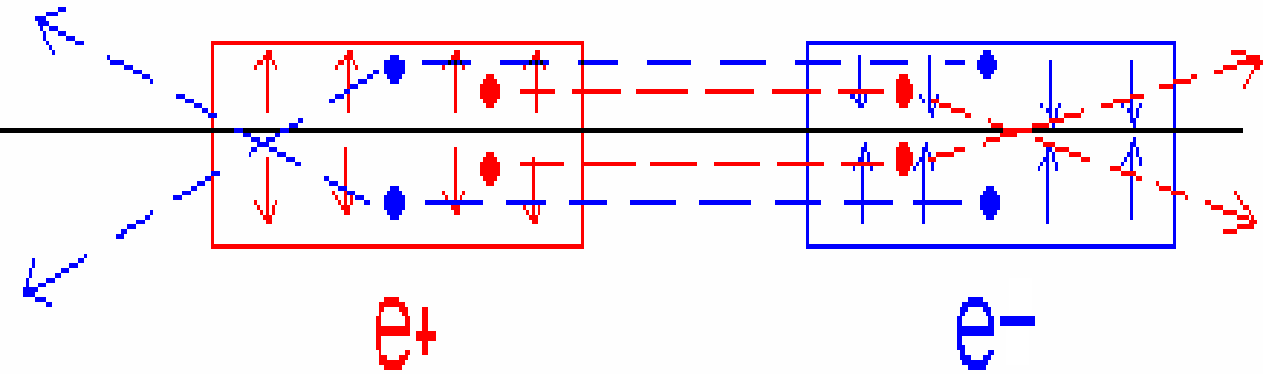
$$F = \frac{\sigma_z}{D_y}$$

σ_z = z kesit alanı
 D_y = dağılma (Disruption) parametresi⁴

Sapma açısı ~ lineer (küçük düşey demet aralığı için)

Demet-Demet Etkileri

* Kıştırma (Pinch) Etkisi



$$L = L_{geom.} \cdot H_D$$

H_D = Işınlık kazanç çarpanı

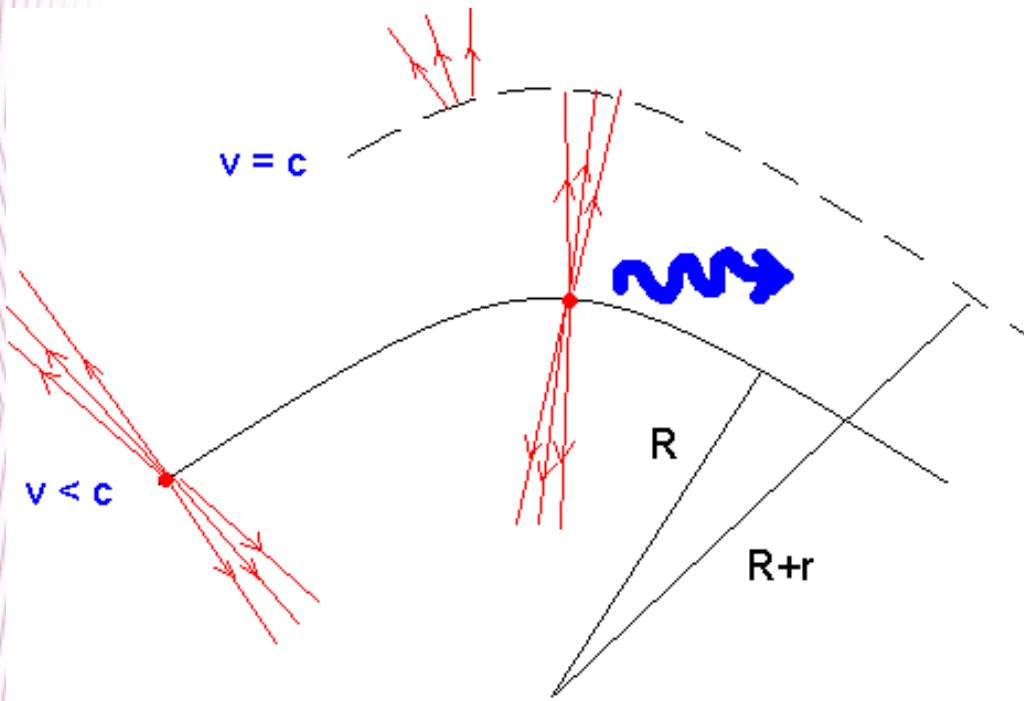
$$H_D = f(D_y)$$

Pinch Etkisi :) (e+e- için)

$$D_y = \frac{2Nr_e\sigma_z}{\sigma_y(\sigma_x + \sigma_y)}$$

Anti-Pinch Etkisi :((e-e- için)

* Demet Işıması



SR Gücü:

$$P_{SR} \approx \frac{r_e mc^2 \gamma^4 \beta^4}{R^2}$$

Demet ışıması enerji yayılması:

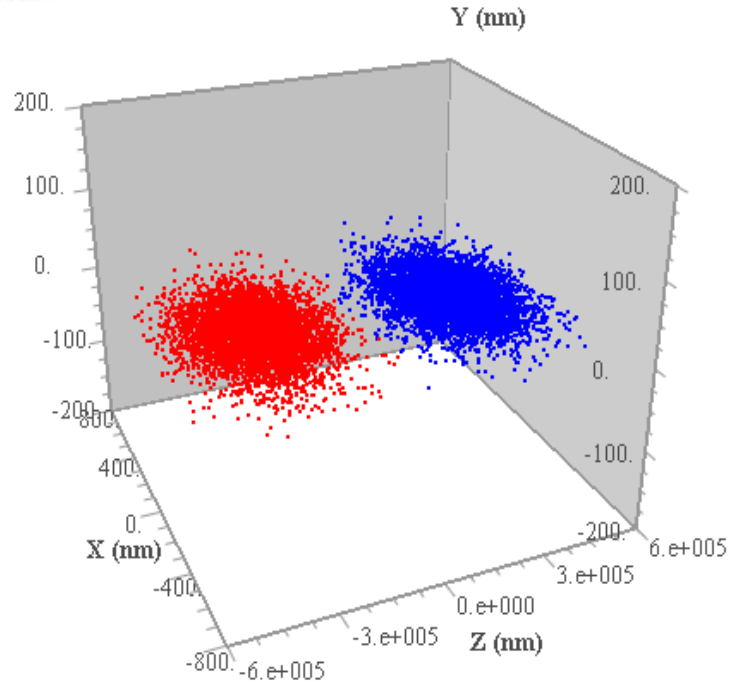
$$\delta_{BS} = \frac{\Delta E}{E} \approx \gamma_e^3 \frac{N^2}{\sigma_x^2 \sigma_z}$$

Çok sayıda yüksek enerjili foton demetlerle etkileşir ve e^+e^- çifti üretir.

Buradaki e^+e^- çiftleri arkaplan oluşturur → **Demet ışıması** :(

Demet-Demet Etkileri:

“Kink” kararsızlığı:



Her iki demetin (düşey)
bağıl hareketi için:

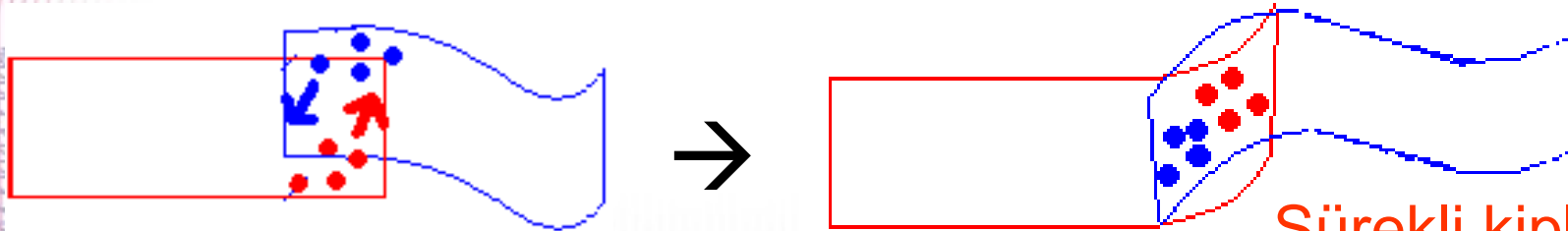
$$\left(\frac{\partial}{\partial t} + c \frac{\partial}{\partial z}\right)^2 y_1(t, z) = -\omega_0(y_1 - y_2)$$

$$\left(\frac{\partial}{\partial t} - c \frac{\partial}{\partial z}\right)^2 y_2(t, z) = +\omega_0(y_1 - y_2)$$

Klasik bağlaşımlı salınıcı

Frekans D_y' ye bağlıdır.

Demetlerden biri başlangıç aralığına (offset) sahipse:



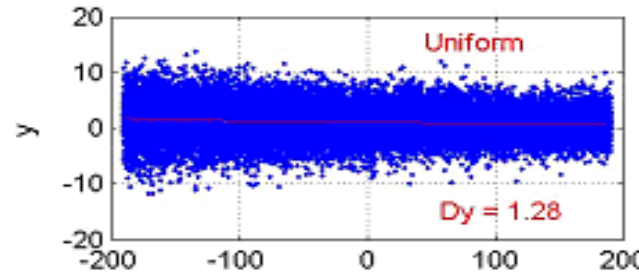
“Kink” Kararsızlığı:

Düzygün e-e+ demetleri çarpıştırıldığında (normal z-dağılımı)

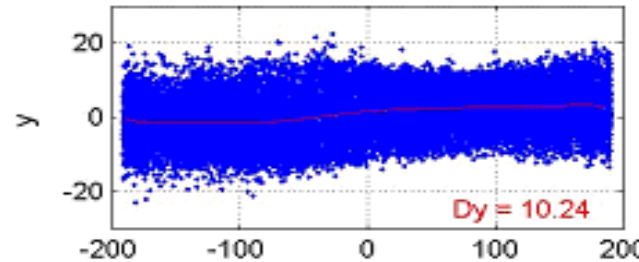
Salınım sayısı
(düzygün dağılım
durumu):

$$n = \frac{\sqrt{D_y}}{\pi\beta\left(\frac{4}{3}\right)^{1/4}}$$

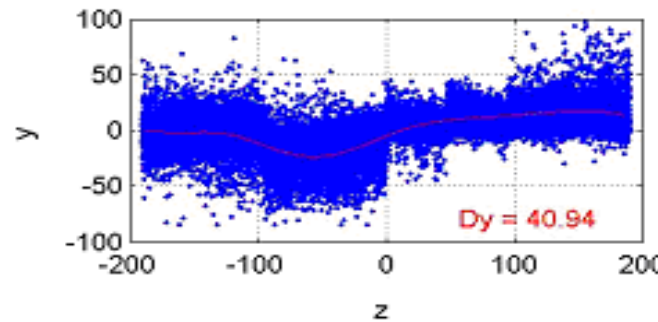
$D_y > 11.4$ için bir
salınımdan fazla
olmaktadır



$n \sim 0.3$



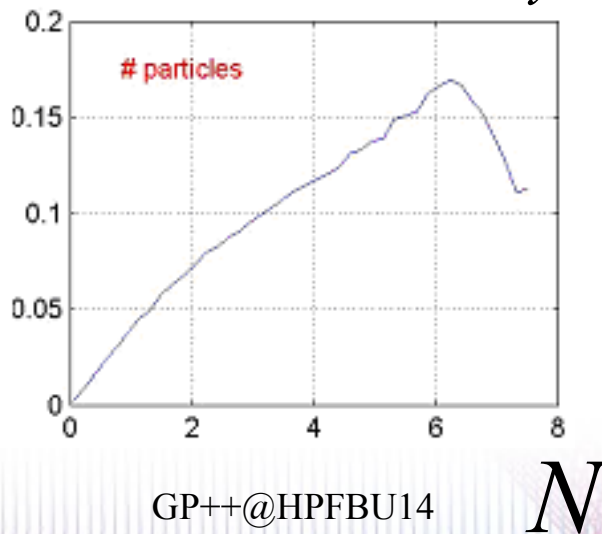
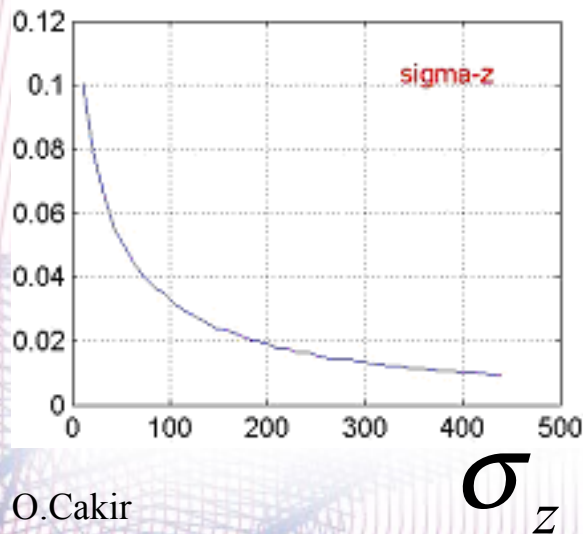
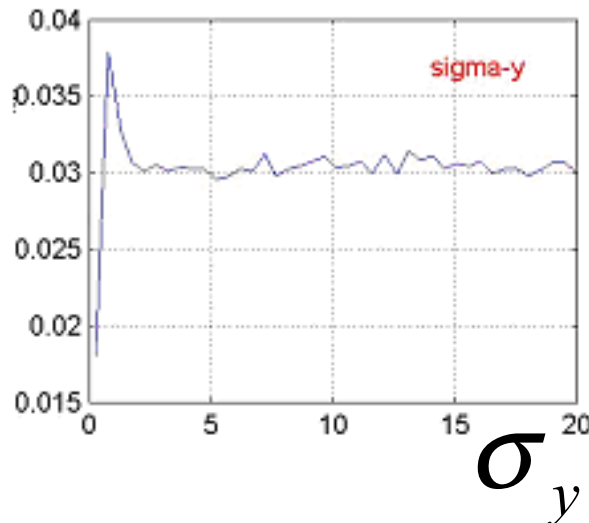
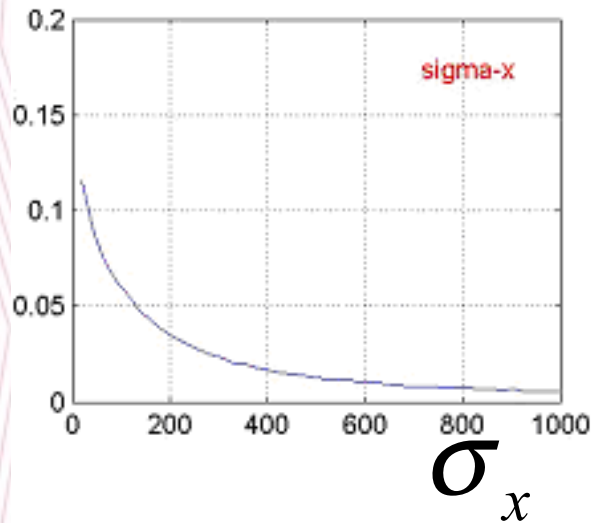
$n \sim 0.9$



$n \sim 1.9$

BS İyileştirme (optimization)

δ_{BS} Demet Işınması Enerji Yayılması (δ_{BS})



e+e-:

$$\delta_{BS} \sim \frac{N^2}{\sigma_z \sigma_x^2}$$

e-e-:

$$\delta_{BS} \sim \frac{N}{\sigma_z^{1/2} \sigma_x^{3/2}}$$

Işınlık İyileştirme

Işınlık (L)

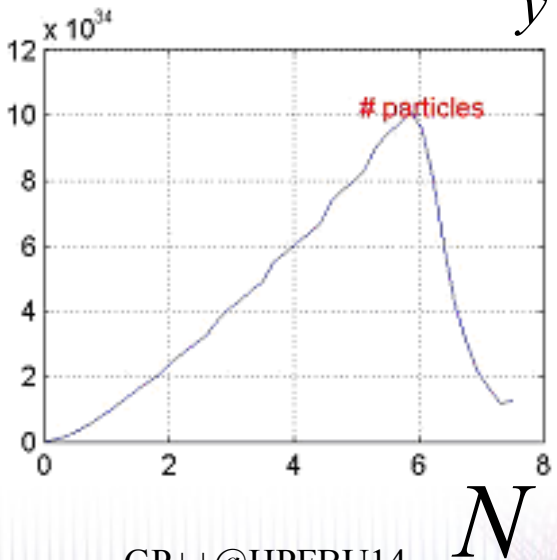
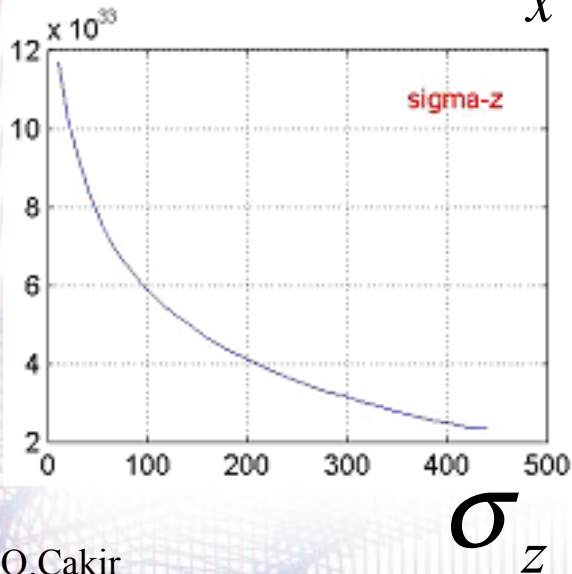
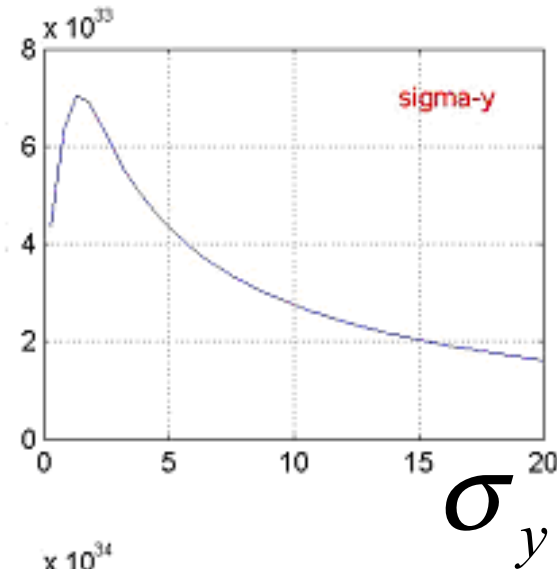
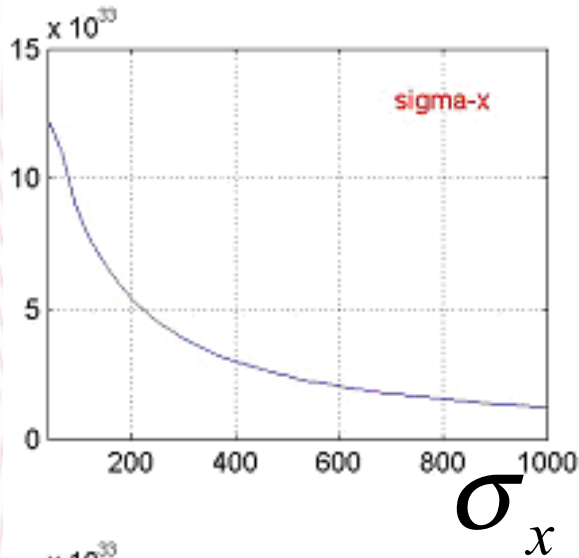
$e+e^-$:

$$L \sim \frac{N^2}{\sigma_x \sigma_y}$$

$e-e^-$:

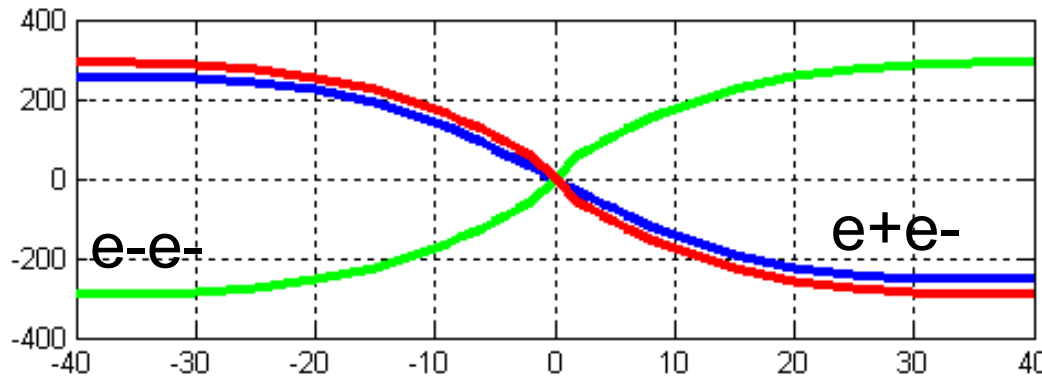
$$L \sim \frac{N^{3/2}}{\sigma_x (\sigma_y \sigma_z)^{1/2}}$$

L



e^+e^- / e^-e^- Karşılaştırması

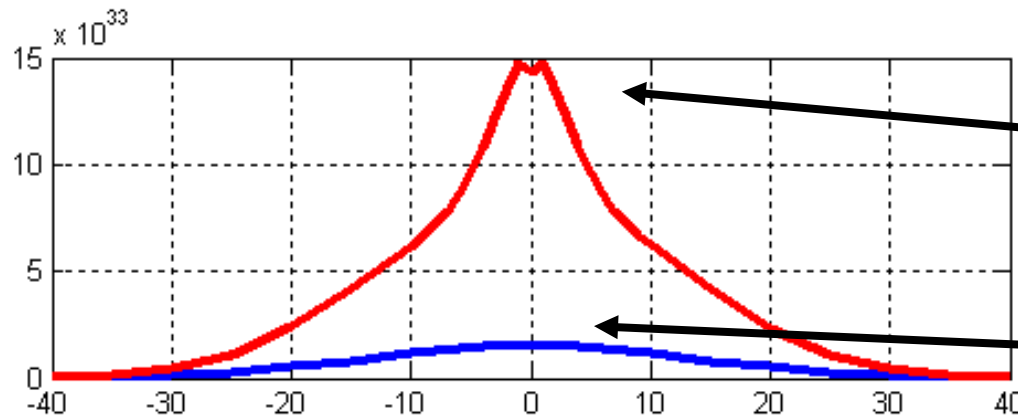
Sapma



$$e^+e^- \rightarrow \sigma_y = \sigma_{y0}$$

$$e^-e^- \rightarrow \sigma_y = 7 \cdot \sigma_{y0}$$

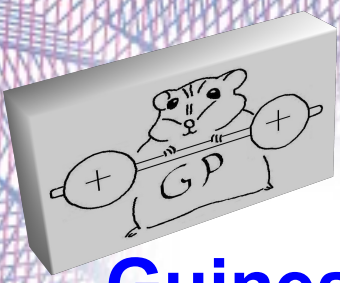
Işınlık



$$L_{e^+e^-}^{Max} = 1.47 \times 10^{34}$$

$$L_{e^-e^-}^{Max} = 1.54 \times 10^{33}$$

Offset



GP++ Kurulum ve Çalıştırma

Guinea-Pig++ (GP++) projesinin amacı, yeni özelliklerin ve işlevselliğin kolay eklenmesi için modüler, nesne yönelimli (C++), kullanışlı ve güvenilir bir program kodu sağlamak.

Web adresi: <http://flc.web.lal.in2p3.fr/mdi/BBSIM/guineapig++-1.0.16.tar.gz>
<https://trac.lal.in2p3.fr/GuineaPig/wiki> (program kılavuzu)

Kurulum:

```
> tar -xvzf guineapig++-1.0.16.tar.gz
> cd guineapig++-1.0.16/external/CMT/v1r18p20061003/mgr
> ./INSTALL
> source setup.sh
> cd ../../../../../../cmt/
> cmt make
```

###daha fazla bilgi için kullanıcı kılavuzuna bknz.

Çalıştırma:

“tests” dizinine geçerek çalıştırmak için aşağıdaki komutu verelim

```
../Linux -686/guineapig++.exe nominal faisceaux lumi_ee_out
```

GP++ Çalıştırılması

```
1.0.16/t : bash
File Edit View Scrollback Bookmarks Settings Help
ocakir@VAIO:~/guineapig++/1.0.16/tests> ../Linux-i686/guineapig++.exe nominal faisceaux lumi_ee_out
bonjour
random generator successfull test : 32-bits computer
*****
* guineapig++ Version 1.0.16
* Program written by Daniel Schulte at DESY and CERN
* object oriented by Guy Le Meur at LAL-Orsay
*****
random generator successfull test : 32-bits computer
checking parameters...
end of checking ...

-----
WARNING : there is a quite important quantity of particles travelling out of grid
miss.1 = 0.106061 out.1 = 265 miss.2 = 0.066055 out.2 = 230
maybe the grid dimensions are to be matched

-----
timer no 1 : 27
timer no 2 : 34
timer no 3 : 79
timer no 4 : 26
timer no 5 : 50
timer no 6 : 0
timer sum : 217.44
ocakir@VAIO:~/guineapig++/1.0.16/tests> ls
acc.dat      beam2.dat      bhabha_photon.ini.gz  lumi_ee_out      manipCompton.sh  positron.ini
acc.dat.c    beam.kumac     bhphotons.kumac      lumi_ee_out      manipPairs.sh
acc.dat.cpp  bhabha.ini.gz  compt.kumac           manipBhabhaC.sh  pair.kumac
beam1.dat    bhabha.kumac  electron.ini          manipBhabha.sh   photon.dat
ocakir@VAIO:~/guineapig++/1.0.16/tests> █
```

GP++ Girdi ve Çıktı Dosyaları

•Girdi dosyası:

- acc.dat (genel)
- electron.ini, positron.ini
(load_beam=1)

Çıktı dosyaları:

lumi.ee.out (olay dosyası)

lumi_ee_out

(bilgilendirme dosyası)

...

acc.dat



lumi_ee_out

.....

----- general results -----

lumi_fine = 1.44449e+34 m**(-2) (from charge densities)
lumi_ee = 1.46715e+34 m**(-2) (from beam particles collisions)
lumi_ee_high = 1.46715e+34 1/m2 (par bunch cross. above energy ecm_min)
lumi_pp = 0 m**(-2)
lumi_eg = 6.38922e+33 m**(-2) (e - gamma)
lumi_ge = 6.42876e+33 m**(-2) (gamma - e)
lumi_gg = 3.43759e+33 m**(-2) (gamma - gamma)
lumi_gg_high = 0 1/m2 (gamma - gamma, with c.o.m energy more than gg_cut)

.....

```
$ACCELERATOR:: nominal
{
energy=250.0;
particles=2;
beta_x=21.0;
beta_y=0.4;
offset_x=0.;
offset_y=0.;
angle_x = 0.0;
emitt_x=10.0;
emitt_y=0.04;
sigma_z=300.0;
charge_sign=-1;
}
```

```
$PARAMETERS:: faisceaux
{
n_x=32 ;
n_y=64 ;
n_z=32 ;
n_t=5 ;
cut_x=3.0*sigma_x.1 ;
cut_y=6.0*sigma_y.1 ;
cut_z=3.0*sigma_z.1 ;
n_m=10000 ;
.....
```

lumi.ee.out dosyasından grafikler

[PAW]

```
nt/create 1 " 17 !! e1 e2 x y z tt vx vy vz p1x p1y p1z p2x p2y p2z la
```

```
nt/read 1 lumi.ee.out
```

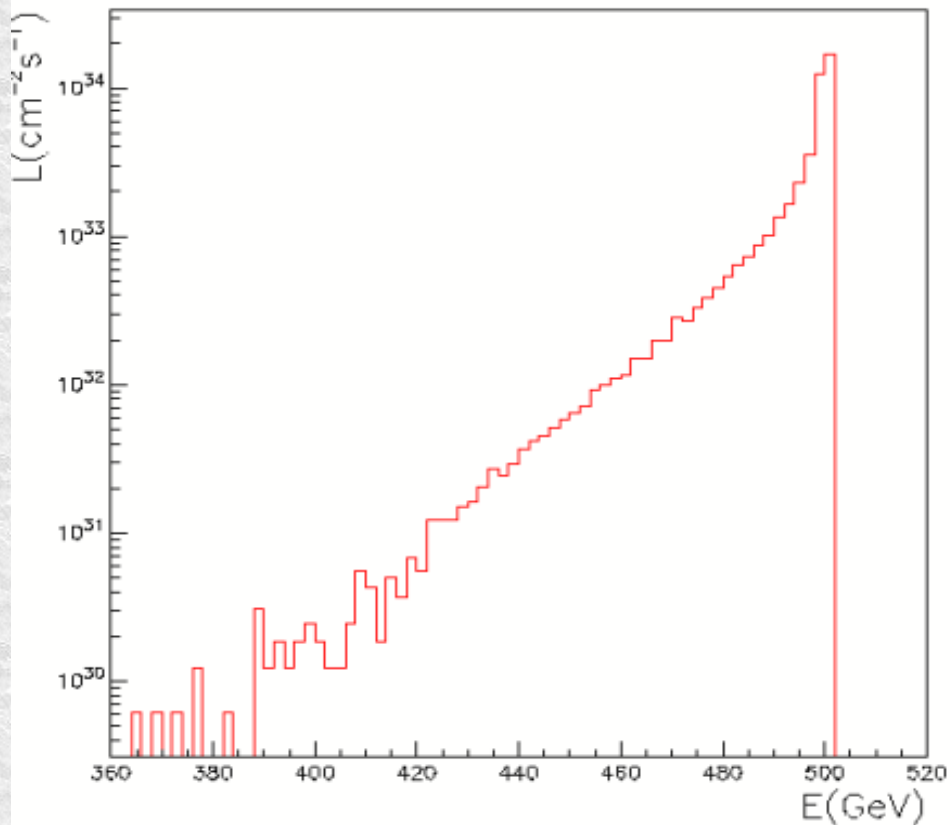
```
sigma lumf=1.46715e+34/75000
```

```
nt/plo 1.sqrt(e1*e2)*2 lumf
```

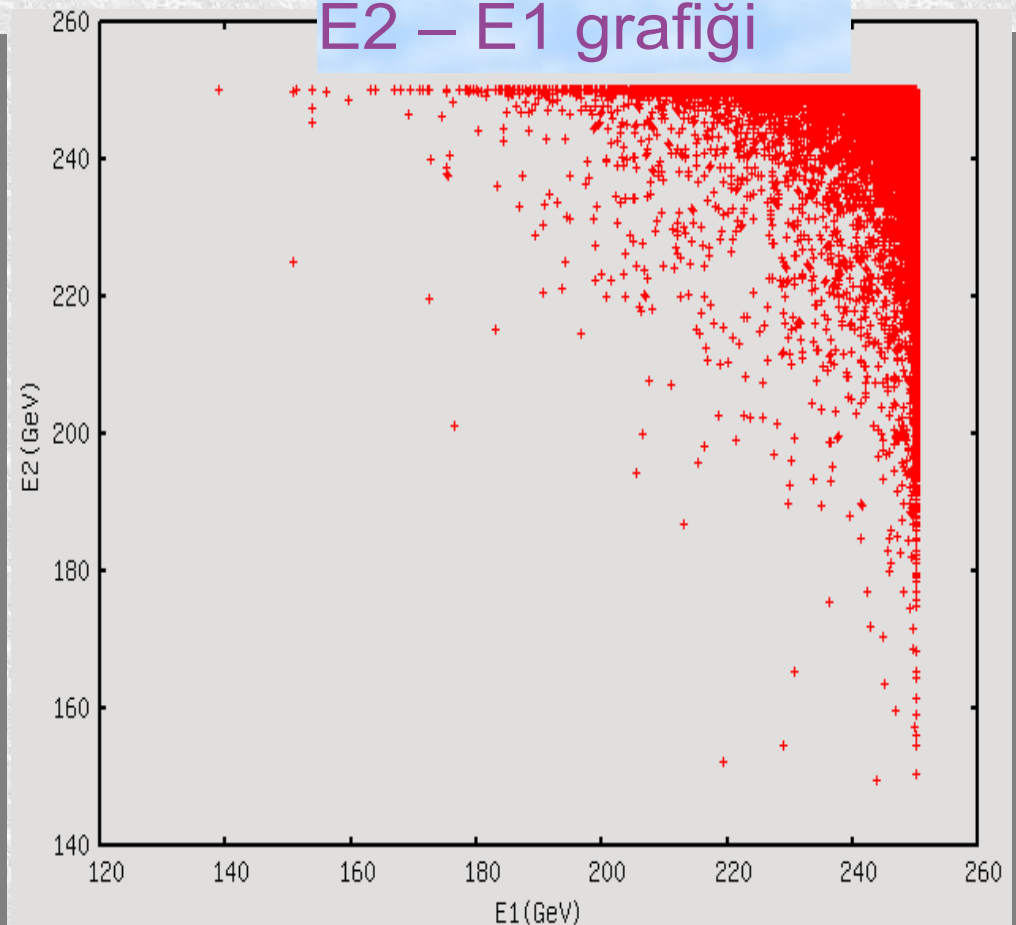
```
...
```

```
gnuplot> plot "lumi.ee.out" u 1:2  
gnuplot>.....
```

L – E grafiği

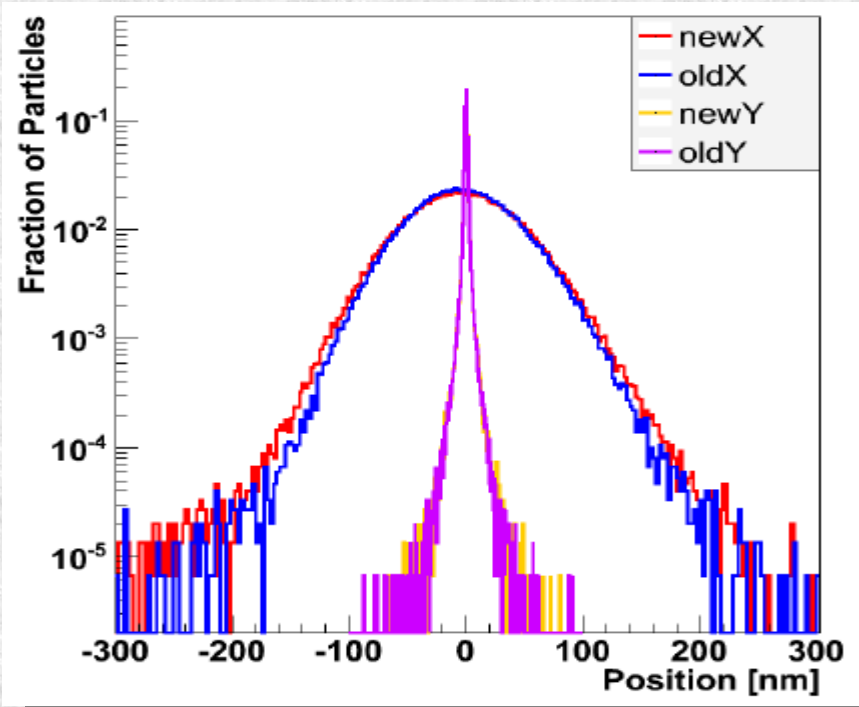


E2 – E1 grafiği



Demet Profili

“Demet Profili” =Guinea-Pig programına girdi olarak verilen demetlerdeki parçacıkların ilk dağılımları.

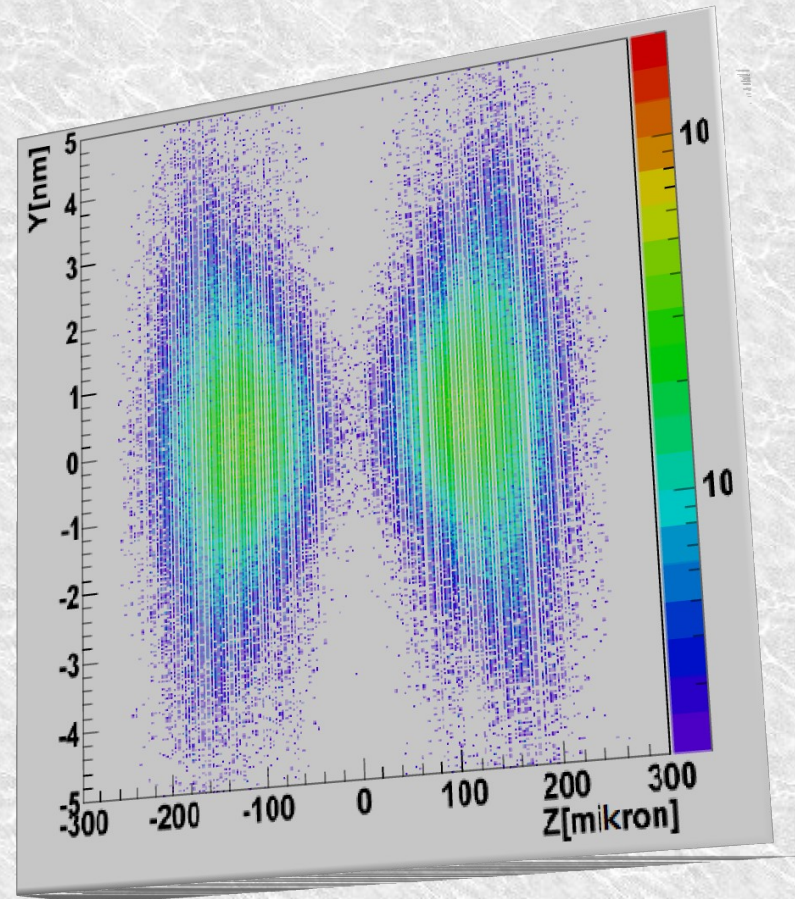


Yatay emittans ϵ

old: 0.55 mm mrad

new: 0.66 mm mrad

demet boyutları $\sigma \sim \sqrt{\epsilon}$



Çıktı Dosyaları Formatları

Demet dosyaları :

Demet-demet çarpışmasından sonra sırasıyla birinci ve ikinci demetin parçacıkları iki dosyaya `beam1.dat` ve `beam2.dat` yazılır. Bu dosyalarda bulunan parçacıkların sayısı benzetim için kullanılan makro parçacıklara karşı gelir. Gerçek parçacıkların sayısı ile normalleştirilebilir. Bu dosyaları oluşturmak için “`store_beam=1`” anahtarı kullanılır. Anahtar “`do_dump = 1`” etkileşme sırasında demetlerin atık (dump) dosyalarını verir: `b1.n` ve `b2.n` (n burada adım sayısıdır)

Demetişması fotonu dosyası :

Burada `photon.dat` dosyası etkileşmeden sonra demetişması fotonlarını içerir. Bu dosyadaki parçacıkların sayısı benzetimde kullanılan makro parçacıklara karşı gelir. Bu dosyayı oluşturmak için “`do_photons=1`” ve “`store_photons=1`” anahtarları kullanılır.

Çift üretim, Compton dosyası :

Diğer birdosya `secondaries.dat`, koherent olmayan çift üretimden veya Compton saçılmasından gelen ikincil parçacıkları içeren dosyadır. Bu dosyayı oluşturmak için “`do_pairs=1`” veya (tam olarak) “`do_compt=1`” veya “`do_bhabhas = 1`”, ve “`track_secondaries=1`” ve “`store_secondaries=1`” gereklidir (+ “`grid=7`”, “`pair_q2=2`” ve “`beam_size=1`”).

Işınlık dosyaları

Anahtar “`do_lumi=1`” ise `lumi.ee.out` (e-e çarpışması) dosyası oluşur.

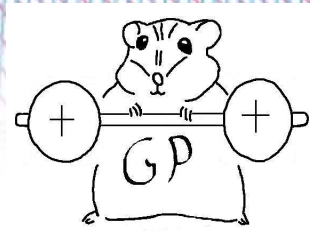
Anahtar “`do_lumi=2`” ise `lumi.eg.out` and `lumi.ge.out` (e-gamma ve gamma-e çarpışmaları) üretilir.

Anahtar “`do_lumi=4`” ise `lumi.gg.out` (gamma-gamma çarpışmaları) üretilir.

hadron dosyası :

`hadron.dat` ile verilir.

Çıktı Dosyası: *lumi.ee.out* dosya formatı



1) demet 1 den gelen, çarpışan parçacığın enerjisi (GeV) - **2)** demet 2 den gelen, çarpışan parçacığın enerjisi (GeV) - **3)** paketçik içinde çarpışmanın x konumu (nm) - **4)** paketçik içinde çarpışmanın y konumu (nm) - **5)** çarpışmanın z konumu (mikrometre) - **6)** çarpışmanın “zaman sayacı” değeri - **7)** birinci parçacığın x yönündeki normalize hızı, $Px1/E1$ - **8)** birinci parçacığın y yönünde normalize hızı, $Py1/E1$ - **8)** ikinci parçacığın x yönünde normalize hızı, $Px2/E2$ - **9)** ikinci parçacığın y yönünde normalize hızı, $Py2/E2$ - **10)** birinci parçacığın polarizasyon vektörünün x bileşeni - **11)** birinci parçacığın polarizasyon vektörünün y bileşeni - **12)** birinci parçacığın polarizasyon vektörünün z bileşeni - **13)** ikinci parçacığın polarizasyon vektörünün x bileşeni - **14)** ikinci parçacığın polarizasyon vektörünün y bileşeni - **15)** ikinci parçacığın polarizasyon vektörünün z bileşeni - **16)** etiket

```
...
1499.19 1318.57 -9.18206 -0.228961 6.45223 2432 3.49804e-06 2.40451e-05 1.02756e-05 -8.45759e-05 0 0 0 0 0 0 40965
1499.58 795.775 -17.9697 -0.0925338 33.4057 4135 0.000103721 4.66456e-06 0.000163938 -1.49925e-06 0 0 0 0 0 0 89639
1500 1500 -14.0141 0.327354 18.044 1535 2.87663e-05 1.29699e-05 8.86349e-06 -3.34156e-05 0 0 0 0 0 0 12249
1500 1257.94 -12.3822 -0.123999 -51.0919 3276 1.13257e-05 1.92278e-05 9.48633e-05 -2.38918e-05 0 0 0 0 0 0 70204
1500 1499.79 9.27209 -0.140635 -36.143 2254 6.6081e-06 1.04383e-05 -7.86901e-06 1.27223e-05 0 0 0 0 0 0 35301
1500 1500 22.6405 -0.325496 -6.54592 2103 -3.43871e-05 -3.31806e-05 -4.30838e-05 -2.13865e-06 0 0 0 0 0 0 30154
794.88 1370.3 11.249 0.329551 -38.0332 2934 -2.49779e-05 -6.28119e-05 -5.46076e-05 5.40458e-06 0 0 0 0 0 0 58827
1499.84 1499.57 -32.4568 0.0939886 -8.03672 3244 7.93479e-05 6.5674e-06 0.000100737 -7.90949e-06 0 0 0 0 0 0 69053
1453 1396.09 9.85077 0.0260521 38.5522 3041 -3.94339e-05 1.19399e-05 -3.22803e-06 5.10417e-05 0 0 0 0 0 0 62066
1204.45 1479.64 12.5286 -0.202892 33.1839 3293 -4.23088e-05 -2.57848e-05 -4.22718e-05 -2.40454e-05 0 0 0 0 0 0 70391
1204.77 1459.3 -9.10189 -0.256908 22.8266 3579 2.42964e-05 4.43498e-05 1.53359e-05 6.30874e-06 0 0 0 0 0 0 78504
...
```

GP++ Grid Üzerinde Çalıştırmak

GP++ programı GRID üzerinde de çalıştırılabilir, daha çok makroparçacıklı ve uzun süreli işler hazırlanabilir. Burada bir “gpgrid.tar.gz” dosyası içinde aşağıdaki dosyalar bulunmaktadır:

- *gp_tarball.tar.gz**, *launch.jdl*, *workers_exe.sh*

İsteğe göre “acc.dat”^{*} dosyası değiştirilir ve

Grid işi için “workers_exe.sh” dosyası hazırlanır:

- **LCG_CATALOG_TYPE=Ifc**
- **LFC_HOST=Ifc.[grid_ismi]**
- **export LCG_CATALOG_TYPE**
- **export LFC_HOST**

* “gp_tarball.tar.gz” dosyası içinde “acc.dat” girdi dosyası ve “gp++” çalıştırılabilir dosya bulunur.

Betik Dosyaları ile Çalışmalar

- Program çalıştıktan sonra çıktı dosyasının LFC (local file catalog) kaydı yapılır:

```
lfc-cr -d [lfc_ismi] --vo [vo_ismi] file:`pwd`/gp_tarball.tar.gz -l $1
```

- Grid iş dosyası “launch.jdl” de mantıksal isim “Arguments” e atanır.

```
Arguments = “lfn:[grid_kullanıcı_dizini/result.tar.gz”;
```

- Burada “acc.dat” dosyasında farklı parametreler için farklı isimlerle Grid'e iş gönderilebilir.
- Böylece, farklı “jdl” dosyaları ile birçok iş aynı anda gönderilebilir.

İş Gönderme/Alma Komutları

- Proxy oluşturma

```
voms-proxy-init - - voms [vo_ismi]
```

- İş gönderme

```
glite-wms-job-submit -a -o job1 launch.jdl
```

- İş durumu öğrenme

```
glite-wms-job-status -i job1
```

- İşin sonucunu alma

```
glite-wms-job-output --dir . -i job1
```

- İş için “guid” arama

```
cat gp_std.out
```

- Sonuçları yerel dizine alma

```
lcg-cp - -vo [vo_ismi] guid:<NO> file://[grid_makinesi_kullanici]/results.tar.gz
```

Özet olarak...

- “gp++” çalıştırılabilir ve “acc.dat” girdi dosyası
- iş tanımlama dosyası “mygp.jdl”

```
Executable="gp++";
```

```
Arguments="COLL COLL-GEN lumi.out";
```

```
StdOutput="gp_std.out";
```

```
StdError="gp_std.err";
```

```
InputSandbox={"gp++","acc.dat"};
```

```
OutputSandbox={"gp_std.out","gp_std.err","lumi.ee.dat","lumi.out"};
```

- iş gönderimi,

```
glite-wms-job-submit -a -o job1 mygp.jdl
```

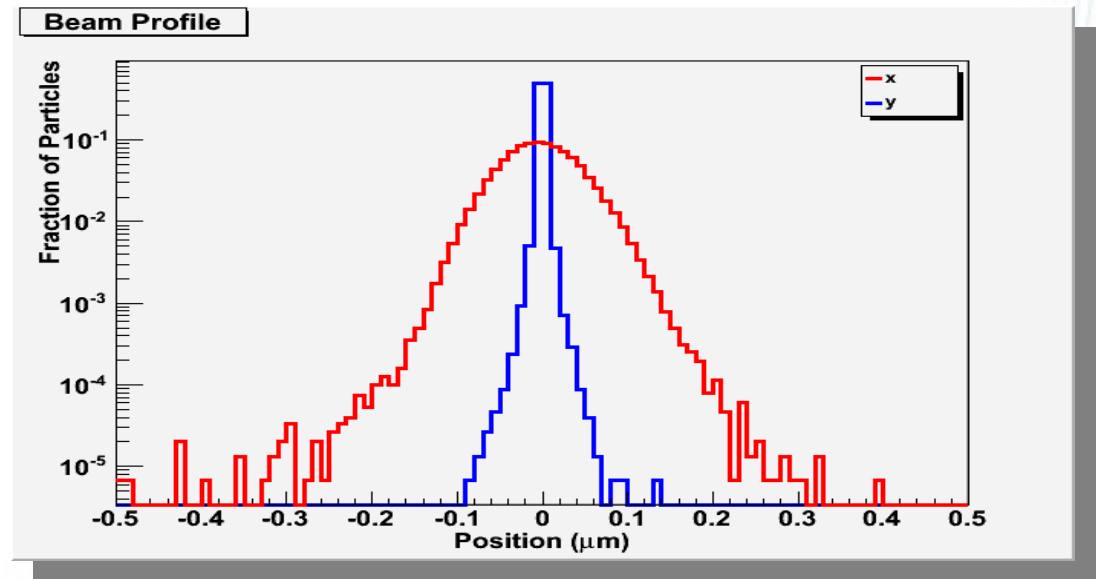


Ek 1: Demet profili grafiđi için Root makro

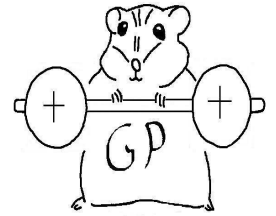
```
{
gROOT->Reset();
gStyle->SetOptStat(0);
// gStyle->SetOptTitle(0);
TH1F *h1=new TH1F("h1","Beam Profile",100,-0.5,0.5);
TH1F *h2=new TH1F("h2","Beam Profile",100,-0.5,0.5);
//
ifstream infile;
infile.open("electron.ini");
Int_t n=150000; Double_t tn=(double)n;
Double_t t[n],x[n],y[n],z[n],vx[n],vy[n];
Int_t nlines = 0;
//
for(Int_t i=0;i<n;i++) {
infile>>t[i]>>x[i]>>y[i]>>z[i]>>vx[i]>>vy[i];
h1->Fill(x[i],1/tn);
h2->Fill(y[i],1/tn);
if (!infile.good()) break;
if (nlines < 5) printf("x=%5f\n",x);
nlines++; }
infile.close();
printf("%d lines found\n",nlines);
h1->SetLineColor(2);
h1->SetLineWidth(3);
h2->SetLineColor(4);
h2->SetLineWidth(3);
h2->GetXaxis()->SetTitle("Position (#mum)");
h2->GetXaxis()->CenterTitle();
h2->GetYaxis()->SetTitle("Fraction of Particles");
h2->Draw();
h1->Draw("same");
leg = new TLegend(0.8,0.8,0.89,0.89);
leg->AddEntry(h1, "x", "l");
leg->AddEntry(h2, "y", "l");
leg->Draw();
c1->SetLogy();
c1->Update();
c1->SaveAs("lumiprof.eps");
c1->SaveAs("lumiprof.png");
}
```

electron.ini

```
...
1503.49232689818 -0.087516 5.53198e-05 -133.199996948242 17.6035 -4.92917
1503.89223077551 -0.0552711 0.000650333 -133.199996948242 8.74508 -8.44319
1503.87312342327 -0.0219723 0.000939495 -133.199996948242 -0.317417 4.49975
1503.77611597703 0.132688 -0.00174452 -133.199996948242 12.6488 -14.831
1503.95188034022 0.0265514 -0.00182947 -133.199996948242 9.89277 14.6361
1503.94232034454 0.0261959 0.00210938 -133.199996948242 5.87565 9.54996
1504.04473961735 0.0237455 0.000112934 -133.199996948242 4.26842 -14.4201
1503.98089612577 -0.0083646 -4.41492e-06 -133.199996948242 3.55048 6.84015
1503.90116308721 0.036871 0.000386696 -133.199996948242 14.4377 7.75466
1504.02399545205 0.0514242 -0.000641272 -133.199996948242 6.50994 -10.4408
...
```



Ek 2: GP++ @ Tr-Grid*



* <http://www.grid.org.tr/>

launch.jdl

```
#####GP++#####  
Executable    = "workers_exe.sh";  
Arguments     = "lfn:/grid/trgridb/users/ocakir/result1.tar.gz";  
StdOutput     = "gp_std.out";  
StdError      = "gp_std.err";  
InputSandbox  = {"workers_exe.sh","gp_tarball.tar.gz"};  
OutputSandbox = {"gp_std.out","gp_std.err"};  
#Requirements = other.GlueCEUniqueID ==  
"kalkan1.ulakbim.gov.tr:2119/jobmanager-lcgpbs-trgridb";  
#####
```

workers.sh

```
>voms-proxy-init --vo trgridb
```

```
>glite-wms-job-submit -a -o job1 launch.jdl
```

```
!/bin/sh  
LCG_CATALOG_TYPE=lfc  
LFC_HOST=lfc.ulakbim.gov.tr  
export LCG_CATALOG_TYPE  
export LFC_HOST  
  
tar xzf gp_tarball.tar.gz  
rm gp_tarball.tar.gz  
cd gp_tarball  
./gp++ ILC LC-GENERAL out > /dev/null  
  
rm gp++  
cd ..
```