

物質中の電磁シャワー
シミュレーション

1 1 0 6 1 0 3 1

田中大地

目次

1.研究目的	1
2.実験を行うための基礎知識	1
i.電磁相互作用	1
ii.空気シャワー	1
iii.電磁カスケードシャワー	1
3.EGS5 とその他機能について	2
i.EGS5 の概要	2
ii.PEGS について	2
iii.CGVIEW について	2
4.計算結果	3
i.電子の透過率	3
ii.光子の透過率	6
5.まとめ	8
6.EGS5 のインストール方法・起動方法・使用方法	9
7.CGVIEW のインストール方法・起動方法・使用方法	11
8.各ファイルの説明	13
i.準備するファイル	13
ii.CG 入力ファイルの説明	13
iii.PEGS 入力ファイルの説明	15
iv.ユーザーコードの説明	16
メインプログラム	STEP1
メインプログラム	STEP2
メインプログラム	STEP3
メインプログラム	STEP4
メインプログラム	STEP5
メインプログラム	STEP6
メインプログラム	STEP7
メインプログラム	STEP8
9.その他	27
10.参考文献	27
11.謝辞	28

1.研究目的

宇宙線断層撮像計画や空気シャワー粒子の横分布測定に役立てるために、電磁シャワーシミュレーションプログラム EGS5 を使用して電子とガンマ線（光子）を鉛に入射し、鉛を通過する電子数を調べるシミュレーションを行う。

2.実験を行うための基礎知識

i. 電磁相互作用

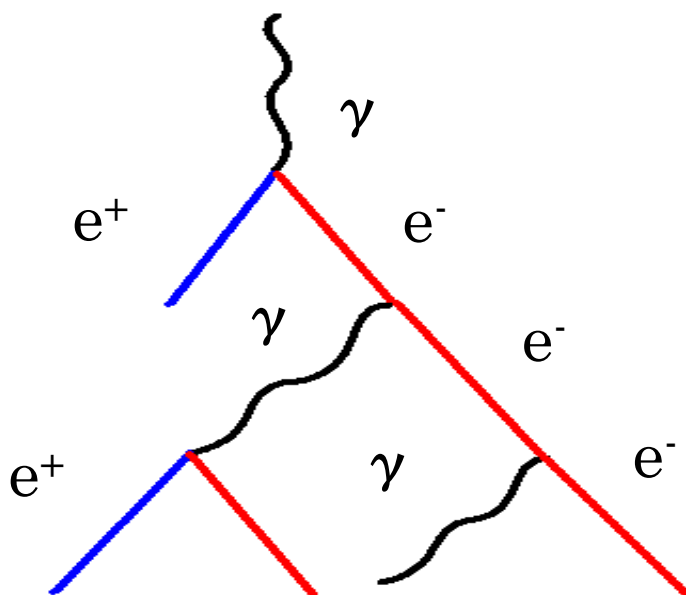
荷電粒子と電磁場の相互作用および電磁場を媒介とした荷電粒子間の相互作用をいう。

ii. 空気シャワー

極めてエネルギーの高い 1 個の宇宙線が大気中入射して、多数の二次的粒子を発生し、それらが数百 m の範囲に渡って同時に地上に到来する現象のこと。

iii. 電磁カスケードシャワー

高いエネルギーの電子や光子が物質に入射すると、電子の制動放射による光子生成や光子による電子・陽電子対生成が起こり、新たに光子や電子・陽電子の対が生成される。二次的生まれた光子や電子も、物質の中を走行中、さらに電子・陽電子対生成や制動放射を起こす。物質の層が十分厚いと、この過程が繰り返し起こり、物質層の深さが増すにつれ、電子、陽電子、光子の数は増していく



3. EGS5 とその他機能について

i .EGS5 について

EGS とは Electron Gamma Shower の略称である。EGS コードシステムは 1Kev～数百 Gev の範囲のエネルギーで、電子、光子輸送計算のモンテカルロシミュレーションを行うものである。EGS は多くの応用分野で用いられているが特に医療分野や放射線測定研究で活躍している。

EGS5 では EGS4 の MORTRAN3 言語とは違い、FORTRAN77 言語を使用している

ii .PEGS について

PEGS(EGS のためのプリプロセッサ※)は、EGS5 モンテカルロ・シャワー・コードの用途のために資料データを生成し、さらに電気磁気相互作用を研究する研究者にユーティリティを供給する 1 セットの FORTRAN サブ・プログラムである。

※プリプロセッサとはある処理を行うソフトウェアに対して、データ入力やデータ整形などの準備的な処理を行うソフトウェアである

iii .CGVIEW について

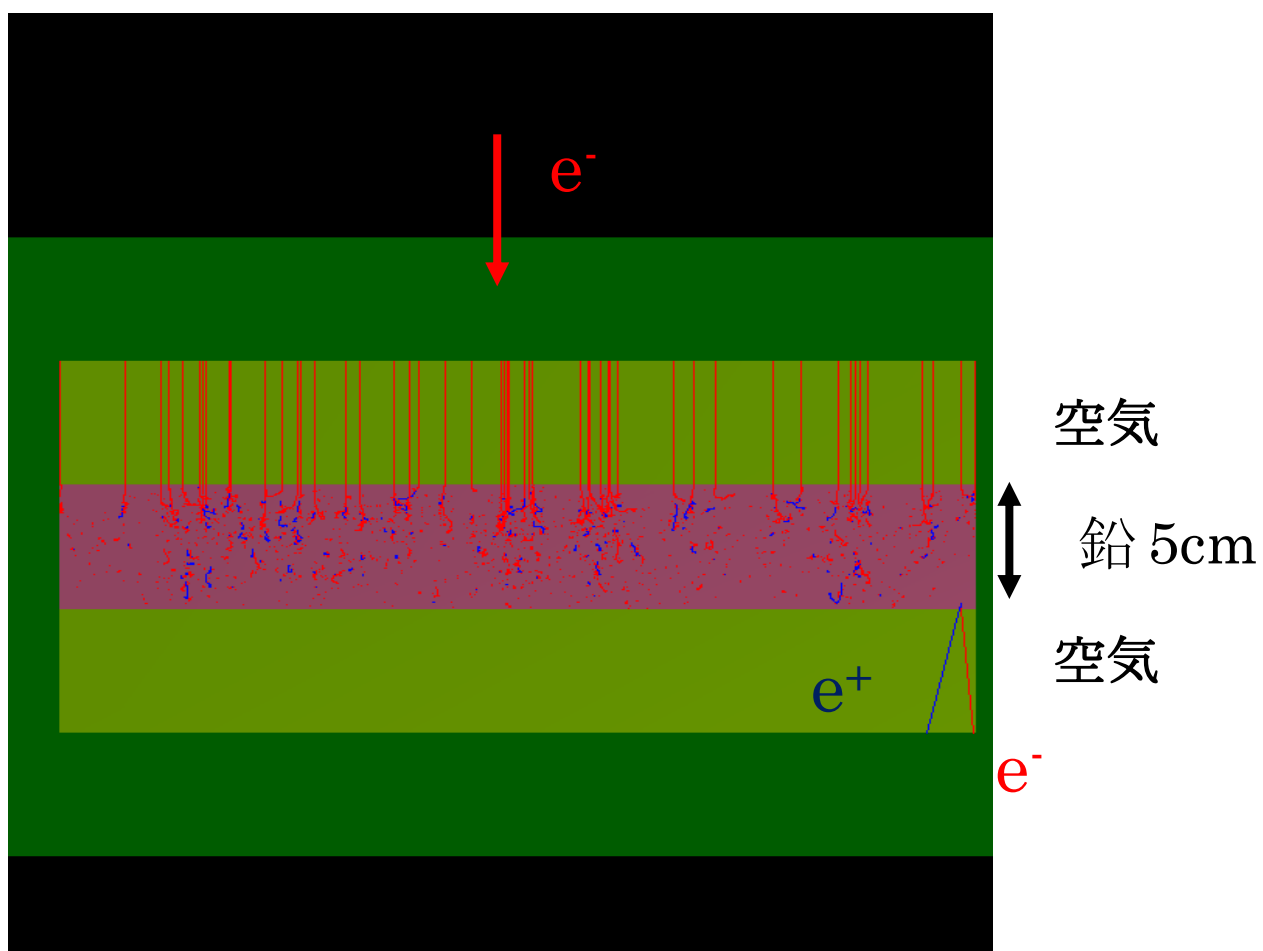
軌跡とジオメトリ表示プログラムは、電子・光子・陽電子の 3 次元軌跡と計算ジオメトリを PC 上で表示するためのシステムである。粒子の軌跡は EGS コードシステムで計算される。

4.計算結果

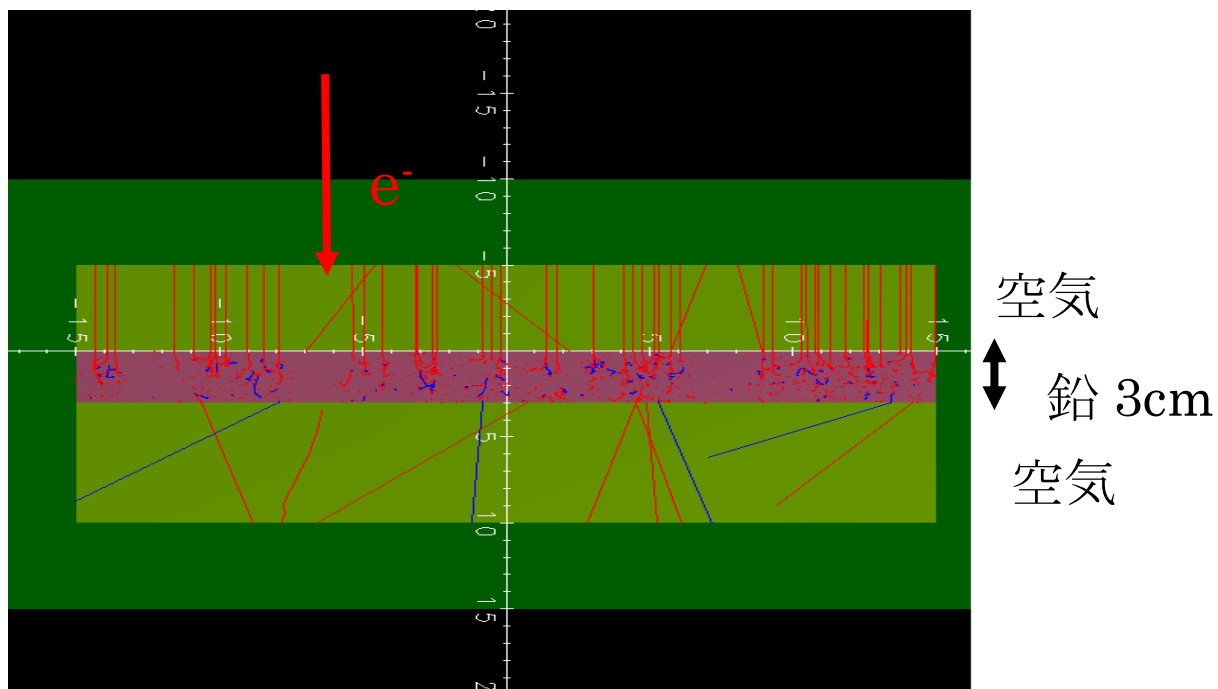
i.電子の透過率

5ページの図は、50個の電子を鉛に入射したときの通過する電子の割合を示している。結果は鉛の厚さが増していくと、通過する電子の数も減少していることが分かる。そして、電子のエネルギーを小さくしていくと通過する数はさらに減少することが分かる。

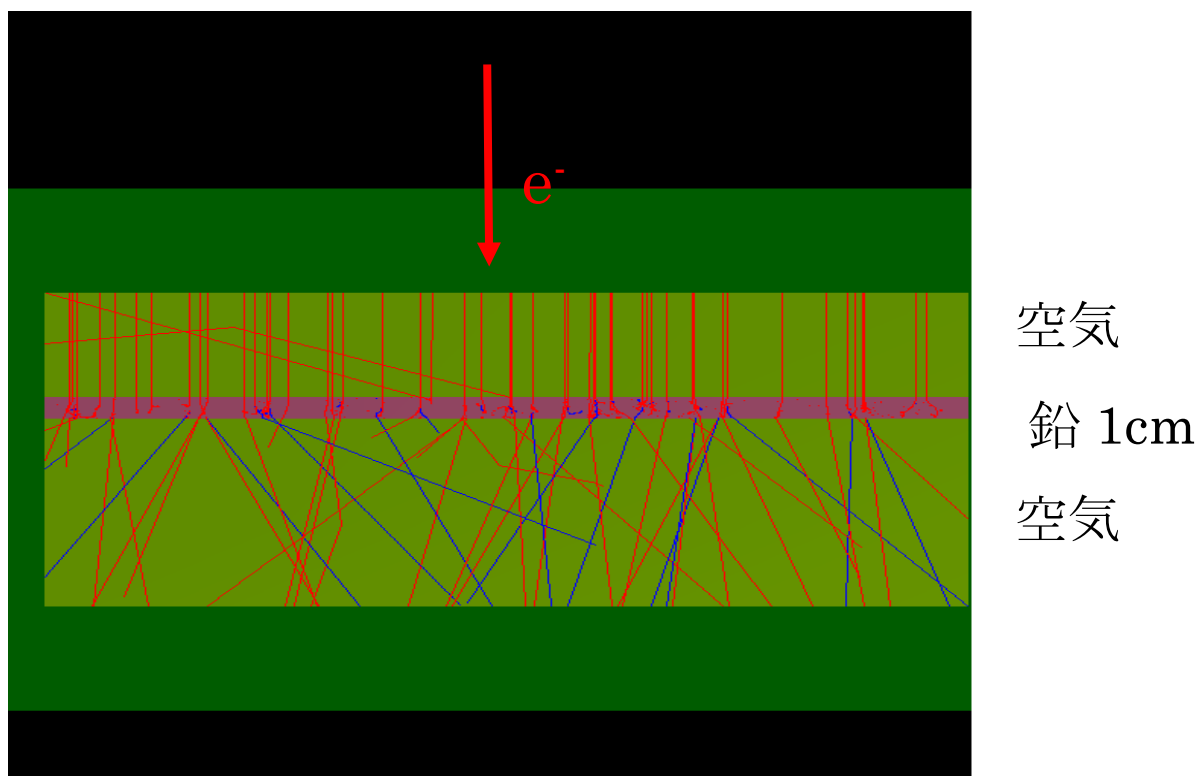
100MeV の電子を 5cm の鉛に入射した場合

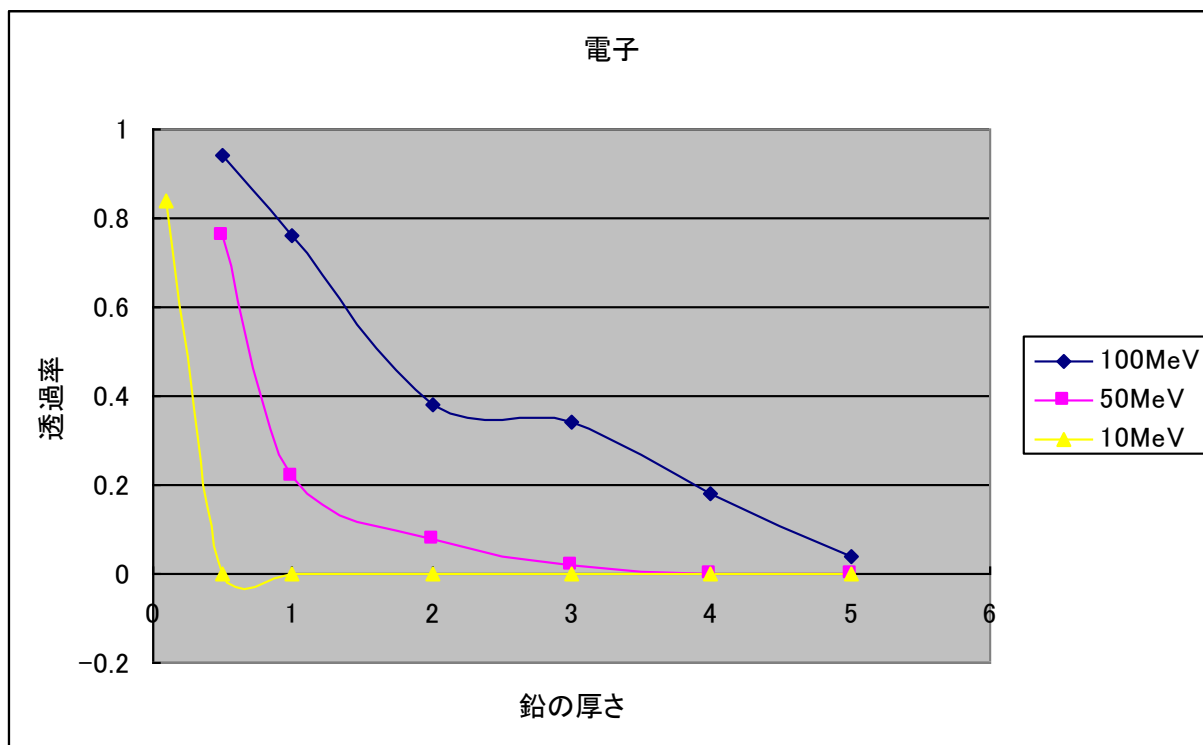


100MeV の電子を 3cm の鉛に入射した場合



100MeV の電子を 1cm の鉛に入射した場合



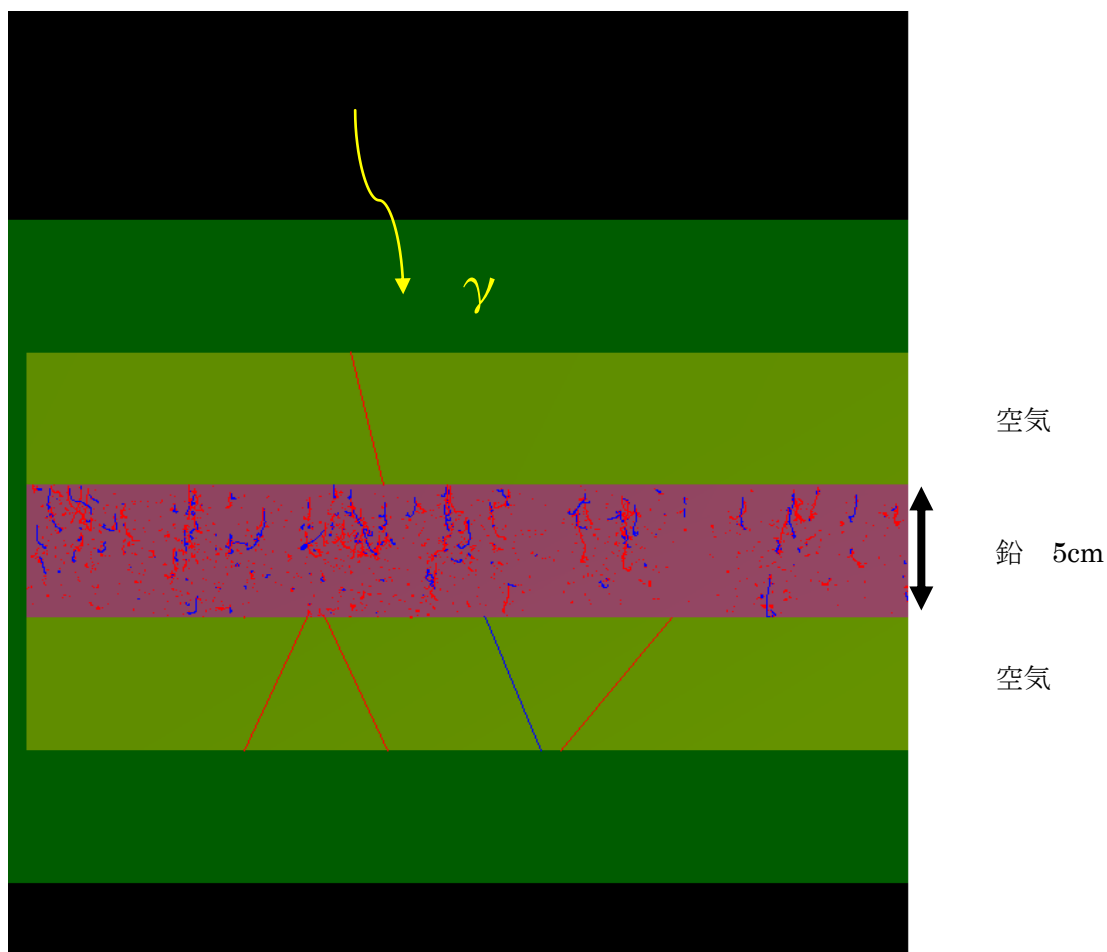


鉛の厚さ	100MeV		50MeV		10MeV	
	透過率	通過数	透過率	通過数	透過率	通過数
5cm	0.04	2	0	0	0	0
4cm	0.18	9	0	0	0	0
3cm	0.34	17	0.02	1	0	0
2cm	0.38	19	0.08	4	0	0
1cm	0.76	38	0.22	11	0	0
0.5cm	0.94	47	0.76	38	0	0
0.1cm					0.84	42

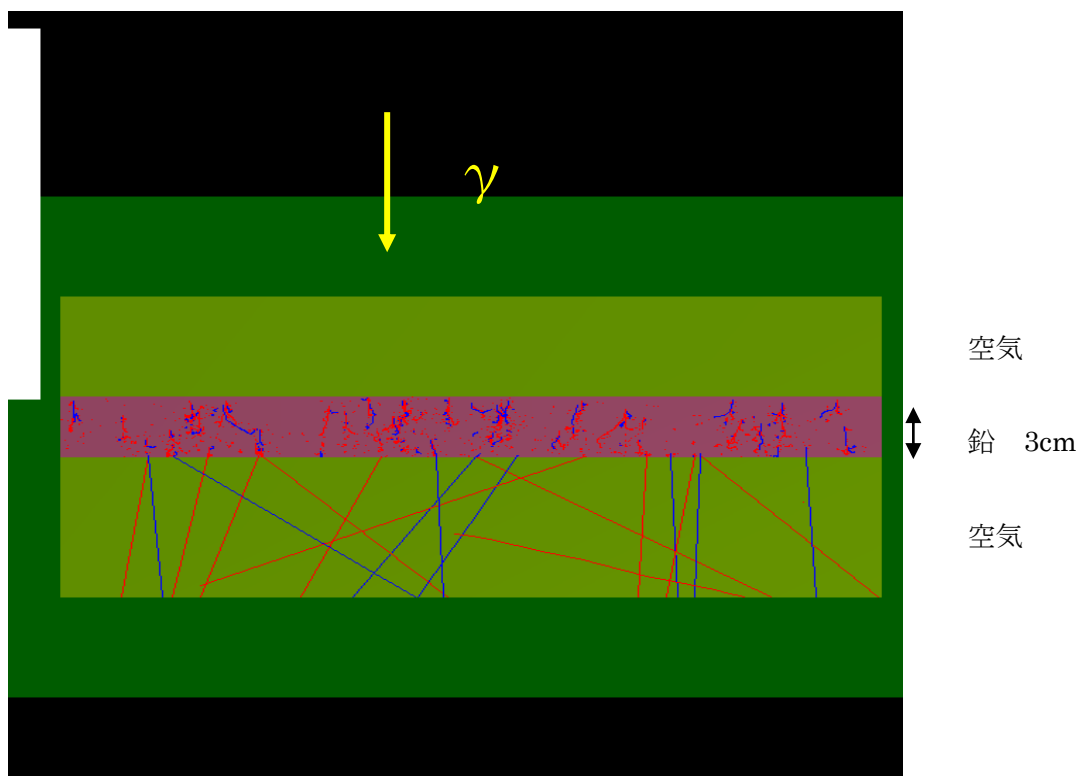
ii. 光子の透過率

7 ページの図は、50 個の光子を鉛に入射したときの通過する電子の割合を示している。結果は電子を入射した時と同じように通過する電子は減少しているが、電子を入射したときよりもさらに減少している。

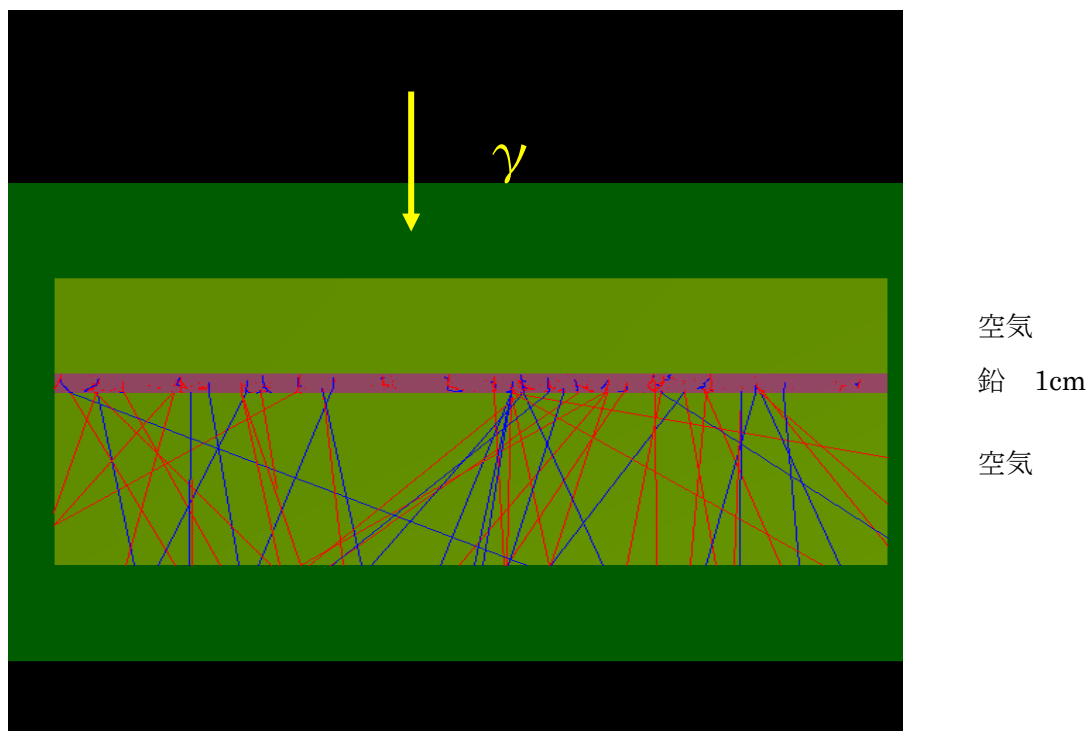
100MeV の光子を 5cm の鉛に入射した場合

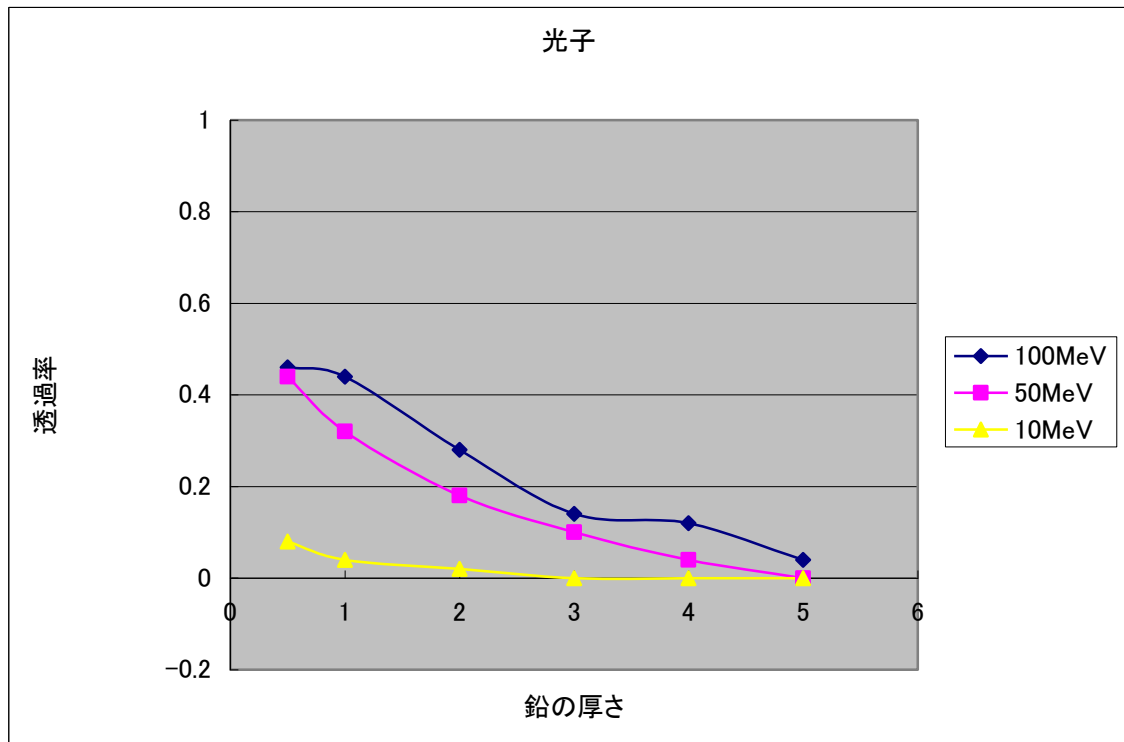


100MeV の光子を 3cm の鉛に入射した場合



100MeV の光子を 1cm の鉛に入射した場合





鉛の厚さ	100MeV		50MeV		10MeV	
	透過率	通過数	透過率	通過数	透過率	通過数
5cm	0.04	2	0	0	0	0
4cm	0.12	6	0.04	2	0	0
3cm	0.14	7	0.1	5	0	0
2cm	0.28	14	0.18	9	0.02	1
1cm	0.44	22	0.32	16	0.04	2
0.5cm	0.46	23	0.44	22	0.08	4

5.まとめ

- ・ 10~100MeV の電子や光子を入射した時、鉛の厚さが増していくと、鉛を通過する電子数は減少していく。
- ・ また、入射粒子のエネルギーを 10~100MeV まで大きくしていくと、鉛を通過する電子の数は増加する。
- ・ 電子と光子の鉛に対する透過率が求まった。

6. EGS5 のインストール方法・起動方法・使用方法

・ EGS5 のインストール方法

- ・ EGS 研究会のページからダウンロードする。
README-J.TXT に従って、ファイルを展開
最低 150 MB の空ディスクが必要
- ・ egs5.tar.gz をホームディレクトリにコピー（ファイル名には version 番号が付いている場合もある。）
egs5 というディレクトリ、ファイルが事前に存在してはいけない。
- ・ gunzip □ -c □ egs5.tar.gz □ | □ tar □ vxf □ -（gunzip は圧縮済みファイルを解く
コマンド。□ は半角スペース）[緑色は unix 向け説明] Windows の解凍ソフト
は使用不可。egs 実行時にエラーを起こす可能性があるため。
- ・ cd □ egs5（cd はディレクトリを移動するコマンド。その後にスペースを空けて、
移動先のディレクトリ名を書く）
- ・ egs5 をインストールしたディレクトリで pwd と打ち込んで表示される ディレクト
リ名を調べる。[pwd は print working directory. 今いるディレクトリ名を得る。]
例:/home/hirayama/egs5
- ・ それを egs5run 内の BASKET という変数に書き込む。
例 BASKET=/home/hirayama/egs5
- ・ [エディターを使ったことがない方には、TeraPad, EmEditor などフリーソフトの
エディタでの変更をお勧めします。Word などのワープロソフトは制御記号が自動的
に挿入され混乱するので、なるべく使用しない。]
- ・ Cygwin 以外の場合には MY_MACHINE という行を書き換える。
- ・ mkdir □ userdir [mkdir はディレクトリの新規作成コマンド。この場合 userdir と
いうディレクトリが作成される]
- ・ cd □ userdir

• cp□../egs5run□./egs5run [cp□A□B でファイル A がファイル B に複写される。..
は一つ上のディレクトリ。./は現在のディレクトリ。/はディレクトリ区切り記号。後
半の./egs5run は.と省略可能]

• cp ../samplecodes/ucsampl5/*□ ! サンプルファイルをコピーする。
[* (半角)はすべてのファイルを表すワイルドカード]

• 起動方法

egs5 のディレクトリに移動して
./egs5run と入力する。

Enter name of User Code
(file extension must be **'.f'**)

! ここで、使用するユーザーコードファイルの名前を入力する。

Enter name of READ(4) data file
(file extension must be **'.data'**)
(<CR> for same file name as User Code)

! **CG** 入力ファイルの名前を入力する。

Enter name of UNIT(25) (pegs input file)
(file extension must be **'.inp'**)
(<CR> for same file name as data file
or same file name as User Code)

! **PEGS** 入力ファイルの名前を入力する。

！ ここで、コンパイルが行われるが少し時間が掛る。

！ エラーが出る場合は、正常に起動しない。

Does this user code read from the terminal?

(Enter **1** for yes, anything else for **no**)

！ そのまま計算を開始する場合は、**1**

バックグラウンドで計算する場合は、それ以外を入力する。

・使用方法

上手く計算が行われると、**egs5job.out** に計算結果が表示される。

また、CG を表示する場合、**egs5job.pic** というファイルが必要になる。これはユーザーコードで設定する。

7.CGVIEW のインストール方法・起動方法・使用方法

・インストール方法

1. <http://rcwww.kek.jp/research/egs/kek/cgview/>からダウンロードする。

2. インストールするディレクトリを作成し移動する。

```
[/]  
[# mkdir /usr/local/cgview ! 空白
```

```
[/]  
[# cd /usr/local/cgview
```

3. ダウンロードしたものをコピーし、展開する。

4. Root 権限がある場合は、

```
[ /usr/local/cgview]  
[# cp /libborqt-6.9.0-qt2.3.so /usr/lib
```

```
[ /usr/local/cgview]  
[# cd /usr/lib
```

```
[ /usr/lib]  
[# ln -s /libborqt-6.9.0-qt2.3.so libborqt-6.9-qt2.3.so
```

5. Root 権限がない場合は、

同一ディレクトリ内にシンボリックを作成する。

```
[ /usr/local/cgview]  
[# ln -s /libborqt-6.9.0-qt2.3.so libborqt-6.9-qt2.3.so
```

・起動方法

1. cd□~/
2. cd□usr/local/cgview
3. xset□fp+□/usr/share/fonts/alias/TrueType ! 起動して文字化けする場合。
4. cgview□&

・使用方法

画像の出し方

起動画面が出たら、ファイル → 体系・飛跡データ読込 → **egs5job.pic** を選択すると画像が表示される。

拡大方法

拡大したいときは、右クリックを押したまま下方向に移動する。

その他の機能

電子のみ、または光子のみなど表示させたいときは設定で変える事が出来る。

参考: <http://rcwww.kek.jp/egsconf/>

8.各ファイルの説明

i.計算で使用するファイル

- *.f ファイル(ユーザーコード)
- *.inp ファイル(PEGS 入力ファイル)
- *.data ファイル(CG 入力ファイル)

・ユーザーコードでは、線源、体系、検出器の設定を行う。

・PEGS 入力ファイルでは、物質の情報(物質名、元素記号、密度 等)を入力する。

・CG 入力ファイルは体系データを作成するためのものである。

ii.CG 入力ファイルの説明

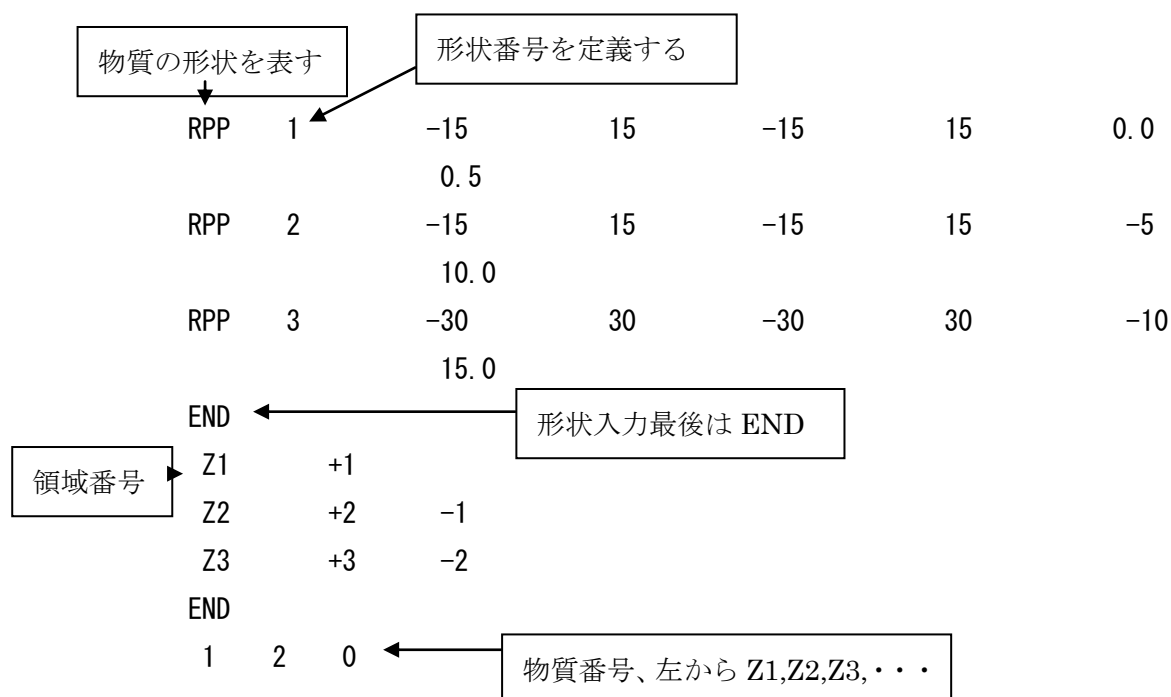
*.data ファイルに入力する。

形状記号は一番左端に記述する。今回の実験では長方形を設定するので、**RPP** とする。
形状パラメータは、左から、**Xmin、Xmax、Ymin、Ymax、Zmin、Zmax**、とする。

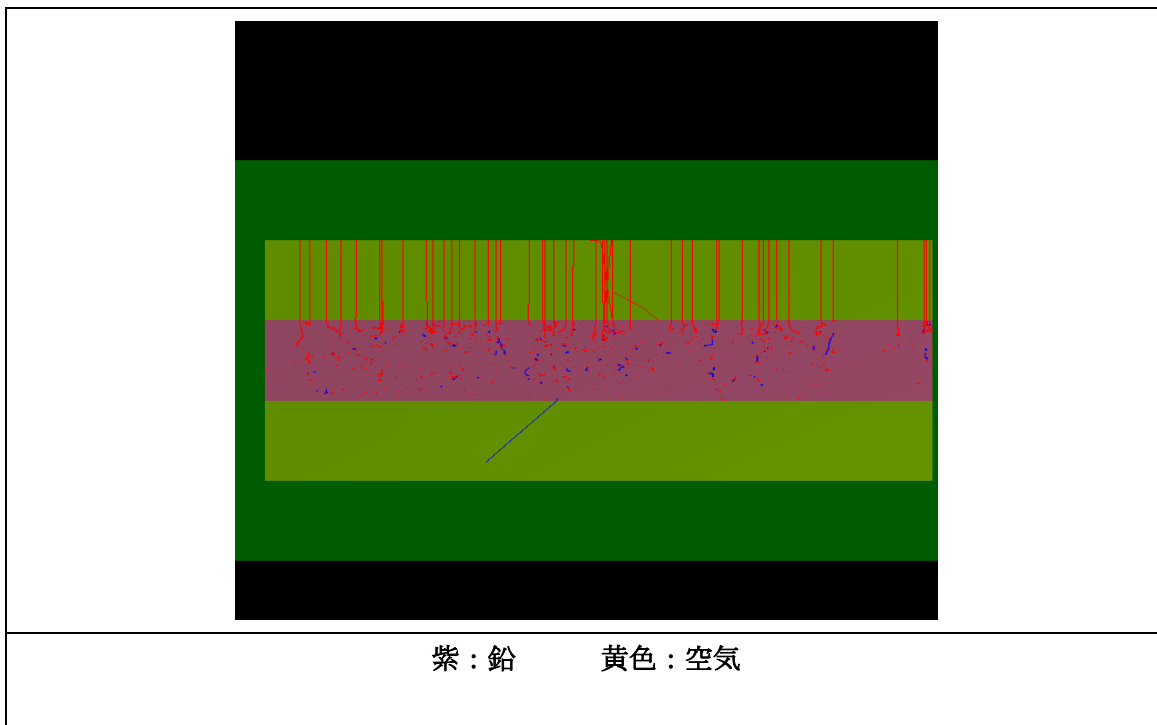
Xmin,Xmax : X 方向の最小値と最大値

Ymin,Ymax : Y 方向の最小値と最大値

Zmin,Zmax : Z 方向の最小値と最大値



これらを入力したらこのような形になる。



iii. PEGS 入力ファイルの説明

AE、AP、UE、UP はエネルギー範囲指定部分で、
AE、UE は電子の下限、上限、AP、UP は光子の下限、上限を表す。
単位は MeV。

NE : 元素の数、RHO : 密度(g/cm³)、RHOZ : 元素の質量比

IRAYL : レイリー散乱断面積、IBOUND : 束縛電子コンプトン断面積

INCOH : 束縛コンプトン散乱角度分布、ICPROF : ドップラー広がり

IMPACT : K 殻電子衝突電離

以下入力コード

ELEM

&INP IRAYL=0,IBOUND=0,INCOH=0,ICPROF=0,IMPACT=0 &END

PB

PB

!物質名

PB

ENER

&INP AE=0.6,AP=0.5,UE=2000.0,UP=2000.0 &END !

TEST

&INP &END

PWLF

&INP &END

DECK

&INP &END

ここは変えない

MIXT

&INP NE=3,RHO= 1.2929E-03,RHOZ= 0.75575,0.23143,0.01282,

IRAYL=0,IBOUND=0,INCOH=0,ICPROF=0,IMPACT=0,

GASP=0.93174

&END

AIR AT NTP

AIR-GAS

!物質名

N O AR

ENER

&INP AE=0.6,AP=0.5,UE=2000.0,UP=2000.0 &END

TEST

&INP &END

PWLF

&INP &END

DECK

&INP &END

ここは変えない

iv. ユーザーコード

・メインプログラム STEP1

```
!-----  
!----- main code -----  
!-----  
!-----  
! Step 1: Initialization  
!-----  
    implicit none  
  
! -----  
!   EGS5 COMMONs  
! -----  
    include 'include/egs5_h.f'           ! 必ず必要なファイル  
  
    include 'include/egs5_bounds.f'     ! 必要なファイルだけで良い  
    include 'include/egs5_brempr.f'  
    include 'include/egs5_edge.f'  
    include 'include/egs5_media.f'  
    include 'include/egs5_misc.f'  
    include 'include/egs5_thresh.f'  
    include 'include/egs5_uphiot.f'  
    include 'include/egs5_useful.f'  
    include 'include/egs5_usersc.f'  
    include 'include/egs5_userxt.f'  
    include 'include/randomm.f'  
  
! -----  
!   Auxiliary-code COMMONs  
! -----  
    include 'auxcommons/aux_h.f'        ! Auxiliary-code "header" file  
  
    include 'auxcommons/edata.f'  
    include 'auxcommons/etaly1.f'
```

```

include 'auxcommons/instuf.f'
include 'auxcommons/lines.f'
include 'auxcommons/nfac.f'
include 'auxcommons/watch.f'

! -----
! cg related COMMONs
! -----

include 'auxcommons/geom_common.f' ! geom-common file
integer irinn

common/totals/                                !使用する変数を定義する
* maxpict
integer maxpict

!**** real*8                                    ! Arguments
real*8 totke
real*8 rnow, etot

real*8                                          ! Local variables
* availke, tnum, wtin, wtsum, xi0, yi0, zi0,
* esbin(1), spg(1), spe(1), r0, r1, phai, rr0

real
* tarray(2), tt, tt0, tt1, cputime, etime

integer
* i, icases, idin, ie, ifti, ifto, ii, j, k, n, nd, ner, nsebin

character*24 medarr (MXMED)

! -----
! Open files
! -----
!-----
! Units 7-26 are used in pegs and closed. It is better not

```

```
! to use as output file. If they are used, they must be opened
! after call pegs5. Unit for pict must be 39.
```

```
!-----
```

```
open(6, FILE=' egs5job. out', STATUS=' unknown' )
open(4, FILE=' egs5job. inp', STATUS=' old' )
open(39, FILE=' egs5job. pic', STATUS=' unknown' )
```

```
! =====
! call counters_out(0)
! =====
```

・メインプログラム STEP2

```
!-----
```

```
! Step 2: pegs5-call
```

```
!-----
```

```
! =====
! call block_set           ! Initialize some general variables
! =====
```

```
! -----
! Define media before calling PEGS5
! -----
```

```
nmed=2 !使用する物質数
if(nmed.gt.MXMED) then
  write(6, '(A, I4, A, I4, A/A)')
*   ' nmed (' ,nmed,') larger than MXMED (' ,MXMED,')',
*   ' MXMED in iclude/egs5_h.f must be increased.'
  stop
end if
```

```
medarr(1)=' PB           ' !物質名
medarr(2)=' AIR AT NTP   ' !空白も含めて 24 文字入れる
```

```

do j=1,nmed
  do i=1,24
    media(i,j)=medarr(j)(i:i)
  end do
end do

chard(1) = 3.0      ! optional, but recommended to invoke
chard(2) = 3.0      ! automatic step-size control

write(6,fmt="( ' chard =' ,5e12.5)") (chard(j),j=1,nmed)

! -----
! Run KEK PEGS5 before calling HATCH
! -----
write(6,100)
100 FORMAT(' PEGS5-call comes next' /)

! =====
! call pegs5
! =====

```

・メインプログラム STEP3

```

!-----
! Step 3: Pre-hatch-call-initialization
!-----
med(1)=1  !物質番号 真空は0 .data ファイルで定義している。
med(2)=2

!-----
! Initialize cg related parameter          !CGVIEW を使う時はいる。
!-----

npreci=3  ! PICT data mode for CGView in free format

ifti = 4   ! Input unit number for cg-data
ifto = 39  ! Output unit number for PICT

```

```

write(6,fmt="( ' CG data' )")
call geomgt(ifti,6) ! Read in CG data
write(6,fmt="( ' End of CG data' ,/)" )

if(npreci.eq.3) write(ifto,fmt="( ' CSTA-FREE-TIME' )" )
if(npreci.eq.2) write(ifto,fmt="( ' CSTA-TIME' )" )

rewind ifti
call geomgt(ifti,ifto)! Dummy call to write geom info for ifto
write(ifto,110)
110  FORMAT('CEND')

!-----
!   Get nreg from cg input data
!-----

nreg=izonin

! Read material for each refion from egs5job.data
read(4,*) (med(i),i=1,nreg)

do i=1,nreg

!オプション on=1,0ff=0
iphter(i) = 0 !光電子の角度分布のサンプリング
iedgfl(i) = 0 !K & L 特性 X 線の発生
iauger(i) = 0 !k & L オージェ電子の発生
iraylr(i) = 0 ! レイリー散乱
lpolar(i) = 0 ! 光子散乱での直線偏光
incohr(i) = 0 ! S/Zrejection
iprofr(i) = 0 !ドップラー広がり
impacr(i) = 0 !電子衝突電離
end do

!-----
! Random number seeds. Must be defined before call hatch
! or defaults will be used. inseed (1- 2^31)
!-----

```

```

luxlev = 1
inseed=1 !乱数
write(6,120) inseed
120  FORMAT (/, ' inseed=', I12, 5X,
*      ' (seed for generating unique sequences of Ranlux)')

!  =====
!  call rluxinit ! Initialize the Ranlux random-number generator
!  =====

```

・メインプログラム STEP4

```

!-----
! Step 4: Determination-of-incident-particle-parameters
!-----
! Define initial variables for incident particle normally incident
! on the slab
nsebin=1
iqin=0          !入射粒子 電子=-1, 光子=0.
ekein=10       !エネルギー
xin=0.0        !入射位置
yin=0.0
zin=-5.0
uin=0.0        ! 入射方向
vin=0.0
win=1.0
irin=0         !スタートするリージョン
wtin=1.0       ! 重さ 普通は1
r0=0           !半径 0cm~15cm をランダムに入射するために定義する
r1=15

```


・メインプログラム STEP5

```
!-----  
! Step 5: hatch-call  
!-----  
    emaxe = 0.00 ! dummy value to extract min(UE, UP+RM).  
  
    write(6,130)  
130  format(/' Call hatch to get cross-section data')  
  
! -----  
! Open files (before HATCH call)  
! -----  
    open(UNIT=KMPI, FILE=' pgs5.job. pgs5dat', STATUS=' old')  
    open(UNIT=KMPO, FILE=' egs5.job. dummy', STATUS=' unknown')  
  
    write(6,140)  
140  FORMAT(/,' HATCH-call comes next',/)  
  
! =====  
    call hatch  
! =====  
  
! -----  
! Close files (after HATCH call)  
! -----  
    close(UNIT=KMPI)  
    close(UNIT=KMPO)  
  
    write(39,fmt="(' MSTA' )")  
    write(39,fmt="(i4)") nreg  
    write(39,fmt="(15i4)") (med(i), i=1,nreg)  
    write(39,fmt="(' MEND' )")
```

・メインプログラム STEP6

STEP6 は体系を決めるところだが、CG 入力ファイルで既に決めているので空白で良い。

・メインプログラム STEP7

```
!-----  
! Step 7: Initialization-for-ausgab  
!-----
```

```
ncount = 0  
ilines = 0  
nwrite = 0  
nlines = 0  
idin = -1  
totke = 0.  
wtsum = 0.
```

```
! =====  
call ecnsv1(0,nreg,totke)  
call ntally(0,nreg)  
! =====
```

```
esbin(1)=ekein  
! Zero the variables  
do j=1,nsebin  
  spg(j)=0.D0  
  spe(j)=0.D0  
end do
```

```
! Set histories and histories to write trajectories
```

```
ncases=1000 !ヒストリー数
```

```
! Set maximum number for pict
```

```
maxpict=49
```

入射粒子数を決める。
この場合 50 個

```
tt=etime(tarray)
```

```
tt0=tarray(1)
```

・メインプログラム STEP8

```

!-----
! Step 8: Shower-call
!-----
! Write batch number
write(39,fmt="( '0 1' ")

do i=1,ncases
!-----
! Start of batch -loop
!-----

wtin = 1.0

wtsum = wtsum + wtin           ! Keep running sum of weights

!-----
! Determine source energy
!-----
ekein = ekein
spg(1)=spg(1)+1.0

etot = ekein + iabs(iqin)*RM    ! Incident total energy (MeV)
availke = etot + iqin*RM      ! Available K.E. (MeV) in system
totke = totke + availke       ! Keep running sum of KE

!-----
! Determine source position
!-----
1100 call randomset(rnnow)
      xi0=2.0*rnnow-1.0
      call randomset(rnnow)
      yi0=2.0*rnnow-1.0
      rr0=sqrt(xi0*xin+yi0*yi0)
      if (rr0.gt.1.0.or.rr0.lt.r0/r1) go to 1100
      xin=r1*xi0
      yin=r1*yi0

```

ここで複数の粒子
をランダムに発生
させる

```

! -----
! Get source region from cg input data
! -----

if(irin. le. 0. or. irin. gt. nreg) then
  call srzone(xin,yin,zin,iqin+2,0,irinn)
  if(irinn. le. 0. or. irinn. ge. nreg) then
    write(6,fmt="(' Stopped in MAIN. irinn = ',i5)")irinn
    stop
  end if
  call rstnxt(iqin+2,0,irinn)
else
  irinn=irin
end if

! -----
! Compare maximum energy of material data and incident energy
! -----

if(etot+(1-iabs(iqin))*RM. gt. emaxe) then
  write(6,fmt="(' Stopped in MAIN.',
1 ' (Incident kinetic energy + RM) > min(UE, UP+RM).')")
  stop
end if

! -----
! Verify the normalization of source direction cosines
! -----

if(abs(uin*uin+vin*vin+win*win-1.0). gt. 1. e-6) then
  write(6,fmt="(' Following source direction cosines are not',
1 ' normalized.', 3e12. 5)")uin, vin, win
  stop
end if

! =====
call shower (iqin, etot, xin, yin, zin, uin, vin, win, irinn, wtin)
! =====

```

```

        ncount = ncount + 1          ! Count total number of actual cases

                                          ! -----
end do                                  ! End of batch loop
                                          ! -----

call plotxyz(99, 0, 0, 0. D0, 0. D0, 0. D0, 0. D0, 0, 0. D0, 0. D0)

write(39, fmt="( ' 9' )")             ! Set end of batch for CG View

tt=etime(tarray)
tt1=tarray(1)
cputime=tt1-tt0
write(6, 150) cputime
150  format(' Elapsed Time (sec)=', G15.5)

```

STEP9 は省略する。

9.その他

nmed	ユーザーコードで使用する物質数
nreg	リージョンの数
med(i)	物質番号
medarr(i)	物質名 空白も含め 24 文字で指定する
inseed	乱数
ncyl	円筒数
nplan	平板数
cyrad	円筒の半径(cm)
zpl	平板の位置(cm)
xin,yin,zin	入射粒子の位置(cm)
irin	入射リージョン番号
uin,vin,win	入射粒子の方向余弦 uin=vin=win=0 のときは等方線源
ncese	ヒストリー数
ekein,iqin	入射粒子のエネルギー、電荷
ebin(i)	各エネルギービンの最大値

10.参考文献

・ KEK 研究会 : <http://rcwww.kek.jp/egsconf/>

・ ユーザーコードの書き方

http://rcwww.kek.jp/research/egs/kek/egs4_japan/usercode.pdf

・ EGS5 用サンプルプログラムリスト

http://rcwww.kek.jp/egsconf/2004-course/sample_user_code.pdf

・ 線源ルーチン : http://rcwww.kek.jp/egsconf/2006-course/source_0607ppt.pdf

10.謝辞

本研究を進めるにあたり、梶野先生、山本先生には適切なご指導をしていただきました。特に梶野先生には、EGS5の使い方を教えていただいたり、研究を進めるためのアドバイスもしていただきました。ありがとうございました。