FLUKA
Estimators and Scoring

7th FLUKA Course

NEA Paris Sept.29-Oct.3, 2008

FLUKA Scoring & Results - Estimators

- It is often said that Monte Carlo (MC) is a "mathematical experiment" The MC equivalent of the result of a real experiment (*i.e.*, of a measurement) is called an estimator.
- Just as a real measurement, an estimator is obtained by sampling from a statistical distribution and has a statistical error (and in general also a systematic one).
- There are often several different techniques to measure the same physical quantity: in the same way the same quantity can be calculated using different kinds of estimators.
- FLUKA offers numerous different estimators, *i.e.*, directly from the input file the users can request scoring the respective quantities they are interested in.
- As the latter is implemented in a very complete way, users are strongly encouraged to preferably use the built-in estimators with respect to user-defined scoring
- For additional requirements FLUKA user routines are provided

Built-In and User Scoring

- Several pre-defined estimators can be activated in FLUKA.
- One usually refers to these estimators as "scoring" capabilities
- Users have also the possibility to build their own scoring through user routines, HOWEVER:
  - Built-in scoring covers most of the **common needs**
  - Built-in scoring has been **extensively tested**
  - Built-in scoring takes BIASING **weights automatically into account**
  - Built-in scoring has **refined algorithms** for track subdivision
  - Built-in scoring comes with **utility programs** that allow to evaluate statistical errors
- Scoring can be geometry dependent AND/OR geometry independent FLUKA can score particle fluences, current, track length, energy spectra, Z spectra, energy deposition...
- Either integrated over the "run", with proper normalization, OR event-by event
- Standard scoring can be weighted by means of simple user routines

Related Scoring Commands

- USRTRACK, USRCOLL score average $d\Phi/dE$ (differential fluence) of a given type or family of particles in a given region
- USRBDX scores average $d^2\Phi/dEd\Omega$ (double-differential fluence or current) of a given type or family of particles on a given surface
- USRBIN scores the spatial distribution of energy deposited, or total fluence (or star density, or momentum transfer) in a regular mesh (cylindrical or Cartesian) described by the user
- USRYIELD scores a double differential yield of particles escaping from a surface. The distribution can be with respect to energy and angle, but also other more "exotic" quantities
- SCORE scores energy deposited (or star density) in all regions
- The output of SCORE will be printed in the main (standard) output, written on logical output unit LUNOUT (pre-defined as **11** by default)
- All other detectors write their results into logical output units assigned by the user (the unit numbers must be **>20**)

More “Special” Scoring

- RESNUCLEi scores residual nuclei in a given region
  - more details are given in the respective lecture on activation
- DETECT scores energy deposition in coincidence or anti-coincidence with a trigger, separately for each “event" (primary history)
- EVENTBIN is like USRBIN, but prints the binning output after each event instead of an average over histories
- ROTPRBIN sets the storage precision (single or double) and assigns rotations/translations for a given user-defined binning (USRBIN or EVENTBIN)
  - more details will be given in the lecture about the use of LATTICE
- TCQUENCH sets scoring time cut-offs and/or Birks quenching parameters for binnings (USRBIN or EVENTBIN) indicated by the user
- USERDUMP defines the events to be written onto a “collision tape” file
- AUXSCORE defines filters and conversion coefficients

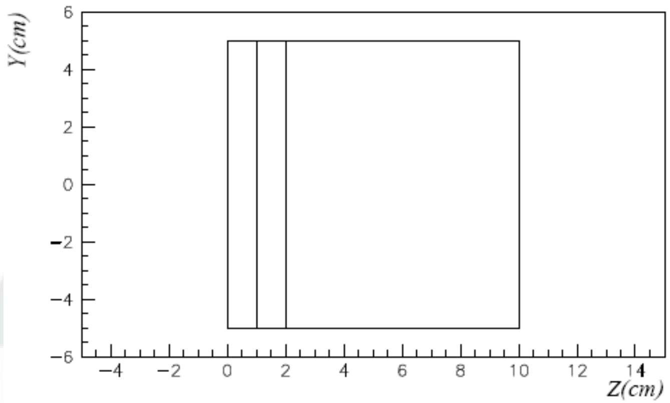
The FLUKA Output Files

The respective Fluka output consists of:

- A main (standard) output, written on logical output unit LUNOUT (predefined as 11 by default) [.out]
  - for details refer to the **lecture explaining the FLUKA output**
- A file with the last random number seeds, unit LUNRAN (2 by default) [ran*]
- A file of error messages, unit LUNERR (15 by default) [.err]
- Any number (including zero) of estimator output files. Their logical unit number is defined by the user [*fort_xx*]
- The available range of logical output numbers is: 21-99
- Generally, the user can choose between formatted and unformatted (binary) scoring (negative or positive sign)
- Possible additional output generated by the user in any user routine

Extending ex_4 with Scoring: ex_5.inp

- Cylinder along Z, filled by water-aluminum-lead and surrounded by Air



Evolution of ex 4.inp:
added new material and scoring

- the USRBIN command allows to superimpose to the geometry a 3-D grid, either cartesian or R-Z-Φ
- On this grid, one can score energy deposition, particle fluence (total or by particle type), as well as the density of interactions
- There is an equivalent EVENTBIN command, that outputs the same quantities event-by-event
- using USERWEIG the results can be weighted by the comscw.f or fluscw.f functions

USRBIN

```
** energy deposition
USRBIN      11.0     ENERGY     -40.0      10.0            15.0 TargEne
USRBIN       0.0                 -5.0     100.0           200.0 &
```

- This is an R-Z-Φ binning (what(1)=11), scoring energy deposition (generalized particle ENERGY, or 208), writing the unformatted output on unit 40, spanning $0<R<10$ in 100 bins, $0<\Phi<2\pi$ in 1 bin (default), $-5<z<15$ in 200 bins.

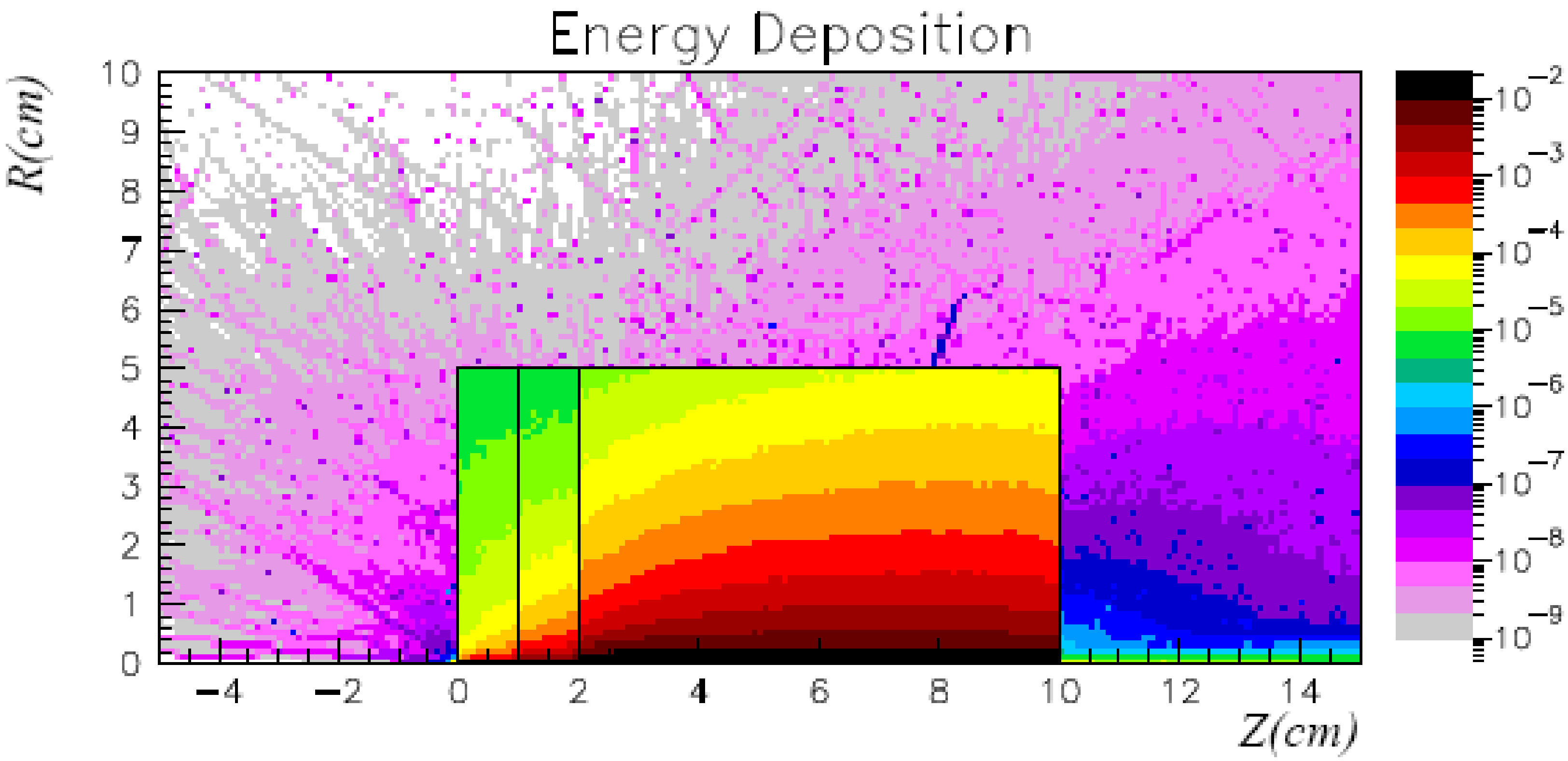
```
** neutron fluence
*           R-Z   EM energy  output unit    Rmax    axis Y    Zmax
*          Rmin       axis X        Zmin  # R-bins # Phi-bins # Z-bins
USRBIN      11.0    NEUTRON     -40.0      10.0            15.0 TargNeu
USRBIN       0.0                 -5.0     100.0           200.0 &
```

- This is a R-Z-Φ binning (what(1)=11), scoring neutron fluence, writing the unformatted output on unit 40, spanning $0<R<10$ in 100 bins, $0<\Phi<2\pi$ in 1 bin (default), $-5<z<15$ in 200 bins.

USRBIN → The Result

WHAT(2) = ENERGY :Energy deposition from a 3.5 GeV proton beam hitting at [0.,0.,0.] directed along z

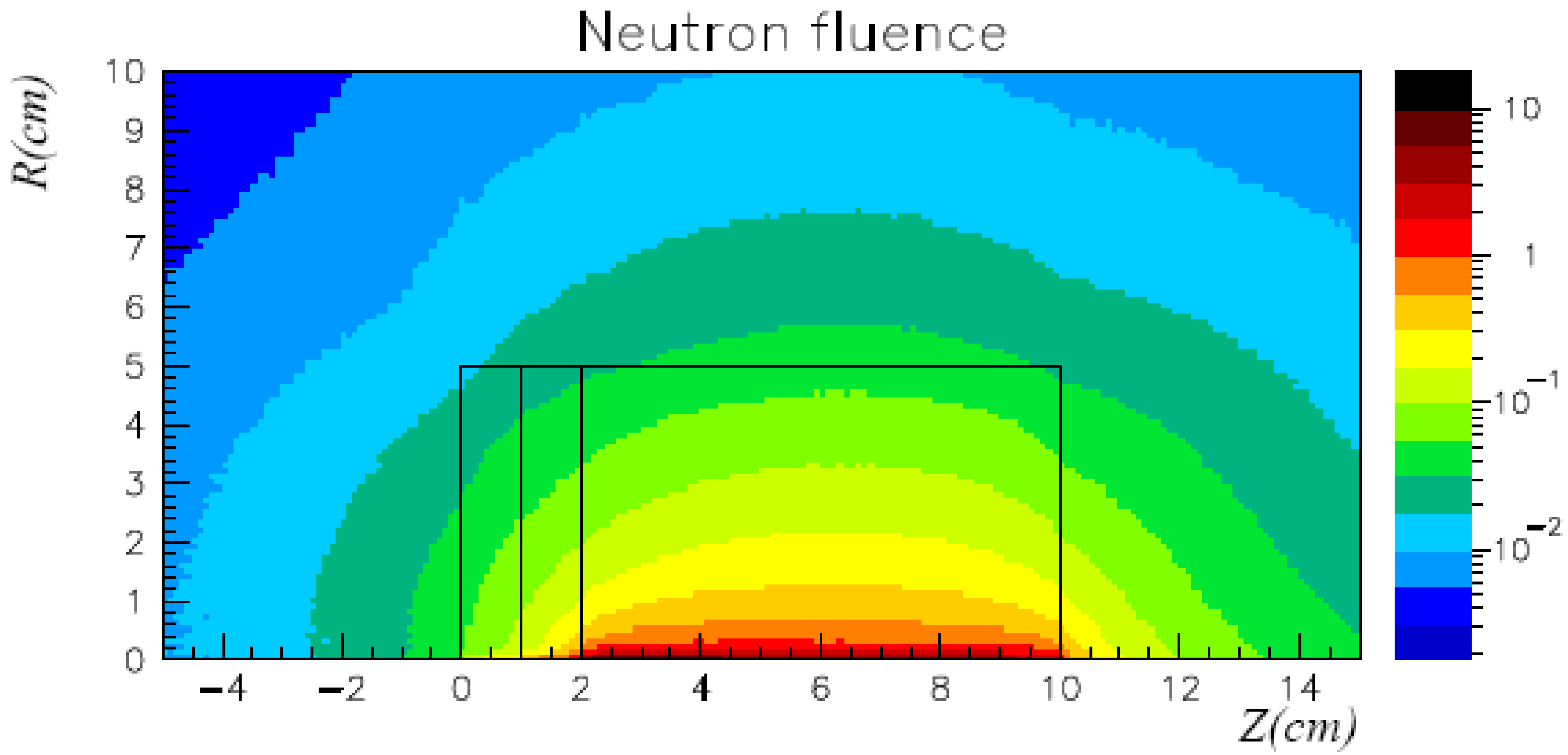
results are normalized to GeV/cm³ per primary



USRBIN → The Result

Same, WHAT(2)= NEUTRON to get neutron fluence

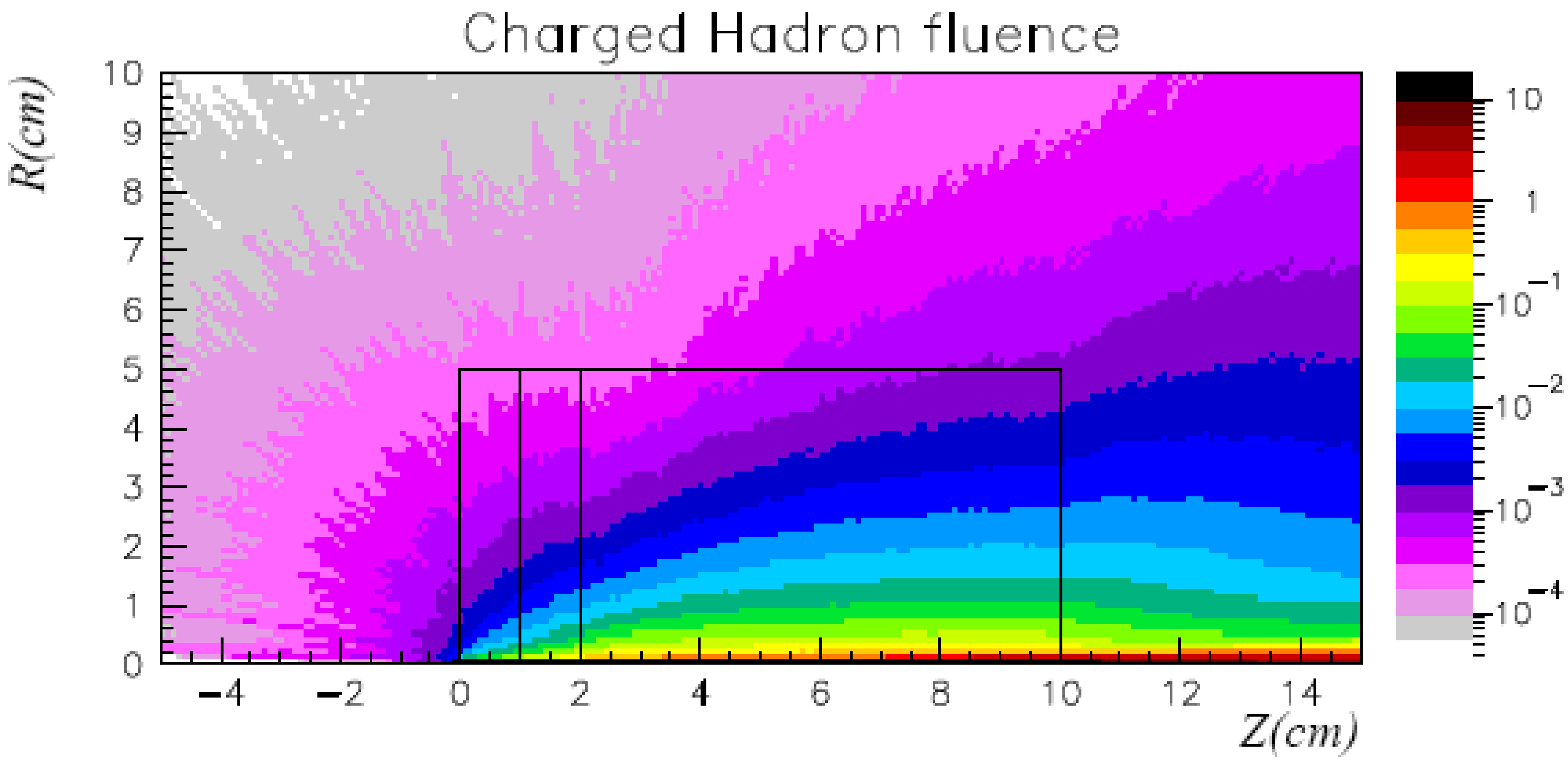
results are normalized to particles/cm² per primary



USRBIN → The Result

Same, WHAT(2)= HAD-CHAR to get charged hadron fluence

results are normalized to particles/cm² per primary



USRBDX

- USRBDX scores double differential (energy and angle) particle distributions across a boundary surface. The angle is with respect to the normal of the surface. The distribution can be fluence or current, one-way or two-ways, according to WHAT(1):

```
*out from lead
USRBDX      99.0  HAD-CHAR       -50.    TARGS3    INAIR   329.87 Sp3ChH
USRBDX      10.0     0.001        40.                             &
```

- Score charged hadrons at the outer surface of the lead segment (from TARGS3 to INAIR). WHAT(1)=99 means: fluence, one-way only, log. intervals in energy. From 1 MeV to 10 GeV in 40 intervals, and one angular interval (default). WHAT(6) is a normalization factor: setting it equal to the surface area provides results normalized to cm^{-2} sr^{-1}. Output to unformatted unit 50

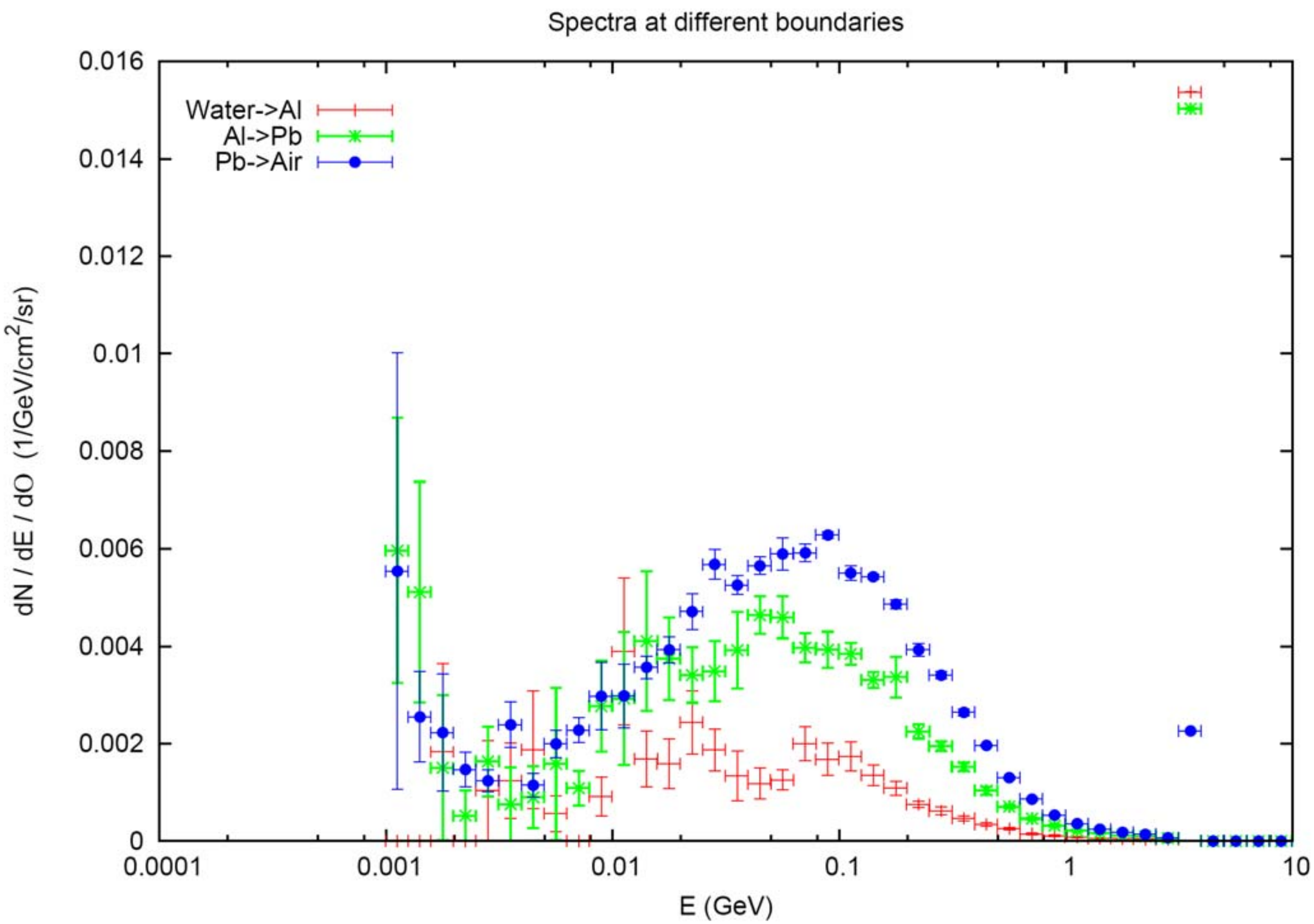
```
USRBDX      99.0  HAD-CHAR       -54.    TARGS2   TARGS3  78.5398 Sp2ChHA
USRBDX      10.0     0.001        40.                         3.0 &
```

- Score at the surface between 2nd and 3rd target section, same as before but in 3 angular bins.

USRBDX → The Result

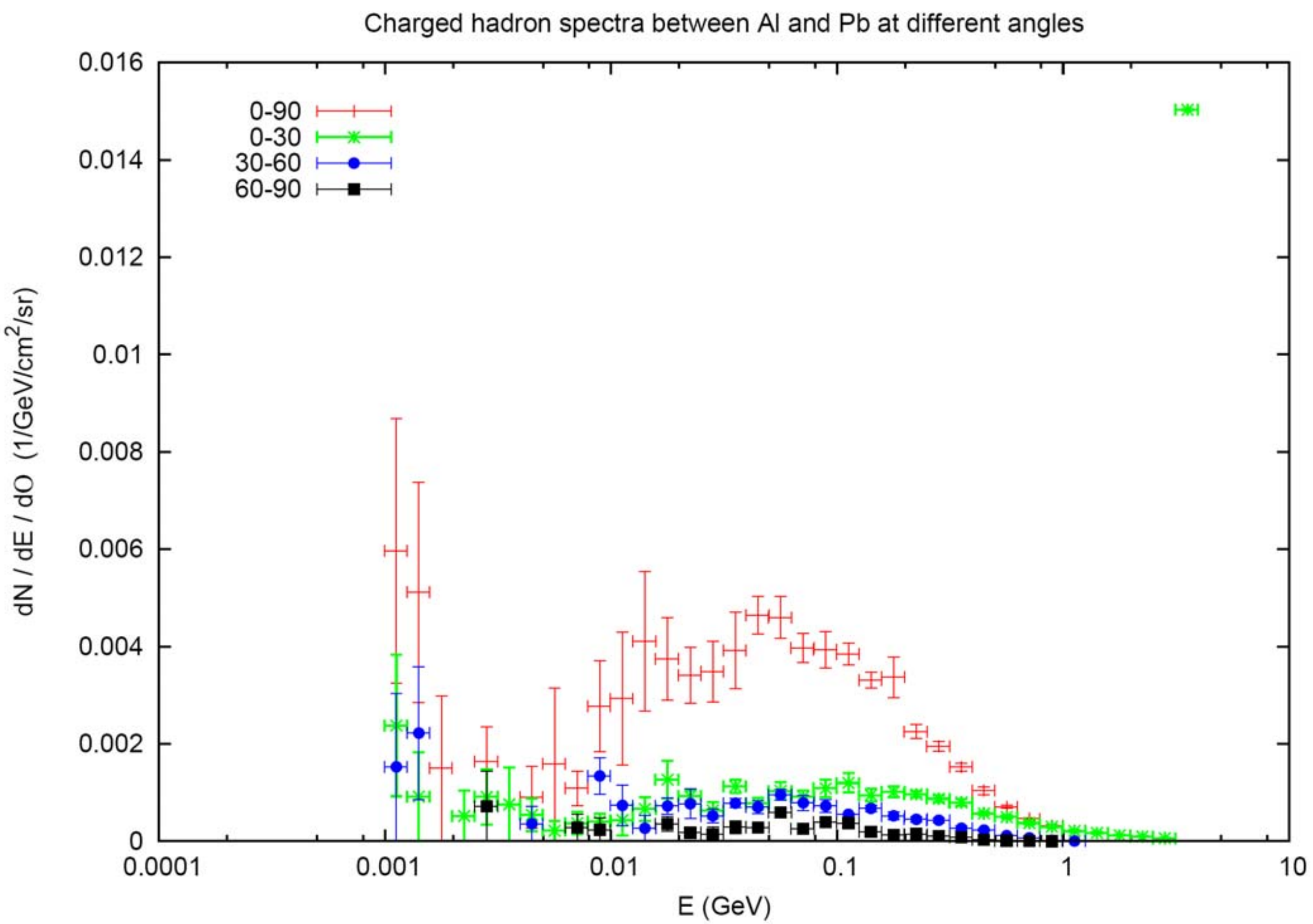
This is true only if the surface area is explicitly given

- Evolution of charged hadron spectra at the various surfaces
  results are normalized to /GeV/cm²/sr per primary



USRBDX → The Result

- Double differential charged hadron spectra for different angles; results are normalized to /GeV/cm²/sr per primary



USRTRACK

- Calculates fluence as a function of energy by scoring track-length in a given volume. Results are normalized to /GeV/cm^2/primary

```
*                log      neutrons    outp. unit        region     volume       # bins
*               Emax          Emin
USRTRACK    -1.0    NEUTRON        -55.     TARGS3   628.31        40. TrChH
USRTRACK    10.0      0.001                                            &
```

- *remember: USRBDX scores on a surface, while USRBIN scores fluence in volumes and gives no differential information*

USRYIELD

- Scores a double-differential particle yield around an extended or a point target.
- "Energy-like" quantities

Kinetic energy , total momentum , total energy , longitudinal momentum in the lab frame , longitudinal momentum in the c.m.s. frame LET

- "Angle-like" quantities (in degrees or radians)

Rapidity in the lab frame , rapidity in the c.m.s. frame , pseudorapidity in the lab frame , pseudorapidity in the c.m.s. frame , Feynman-x in the lab frame , Feynman-x in the c.m.s. frame , transverse momentum , transverse mass , polar angle (*) in the lab frame , polar angle (*) in the c.m.s. frame , square transverse momentum , charge , weighted angle in the lab frame , weighted transverse momentum

USRYIELD

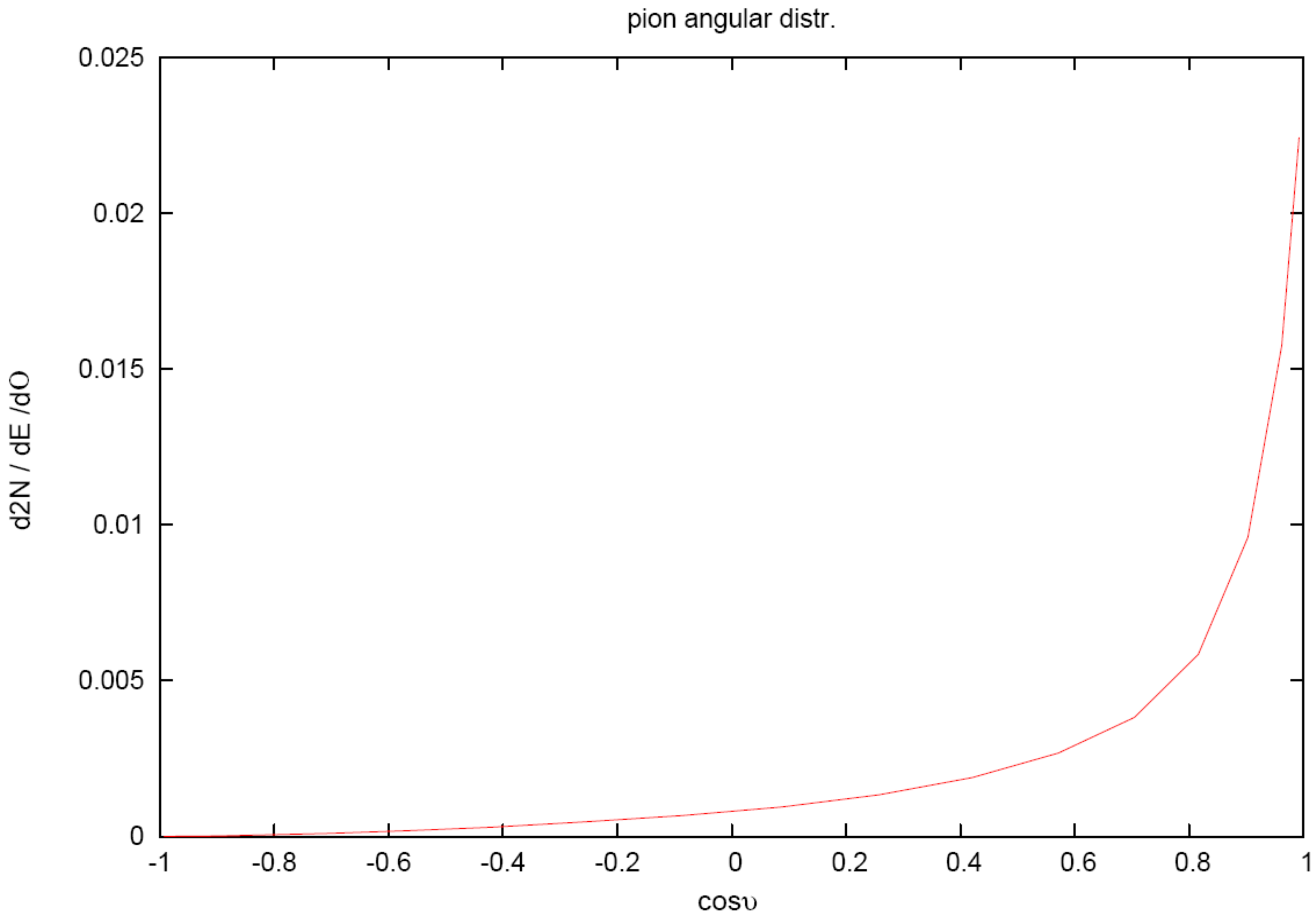
- While option USRBDX calculates angular distributions WITH RESPECT TO THE NORMAL to the boundary at the point of crossing, USRYIELD's distributions are calculated WITH RESPECT TO THE BEAM DIRECTION (or a different direction specified with SDUM=BEAMDEF).

```
*    124 = 24 + 1 * 100 => polar angle (in degrees) and kinetic energy
*                        outp. unit      Reg1       Reg2      Norm
*          Amax      Amin   # A bins     Emax       Emin   dbl.differential
USRYIELD     124.0    PIONS+-      -57.   TARGS3      INAIR       1.0 YieAng
USRYIELD     180.0        0.0       18.     10.0        0.0       3.0 &
```

- Only one interval is possible for the second variable, BUT results are normalized as Double Differential:
  (in this case, particles/GeV/sr)

USRYIELD -> The Result

- pion angular distribution



FILTERS : AUXSCORE

There is the possibility to filter the estimators, restricting the scoring to a selected subset of particles.

For instance: USRBIN energy deposition by muons only

```
USRBIN      11.0      ENERGY     -40.0      10.0            15.0 TargEne
USRBIN       0.0                  -5.0     100.0           200.0 &
AUXSCORE USRBIN        MUONS               TargEne TargEne
```

Assign the "muons" filter to the USRBIN estimator named TargEne

Another example: score the yield of 56-Iron ions (very useful: there is no separate name for each ion specie, except light ones. HEAVYION score all isotopes heavier than alpha's together!)

```
USRYIELD    124.0   ALL-PART   -87.     TARGS3      INAIR       1.0 Fe56
USRYIELD    180.0        0.0    18.       10.0        0.0       3.0   &
AUXSCORE USRYIELD -5602600.               Fe56       Fe56
```

The requested ion is coded in what(2) according to its **A**, **Z** and (optionally) isomeric state **m**:

what(2) = - (100***Z** + 100000***A** + **m***100000000)

with 0==all , i.e. 2600 == all Iron isotopes

User Conversions / Weighing

- Scored fluences are often folded with user-provided response functions to obtain dose equivalent, material activation, *etc.* This can be done off-line or (at some cost in CPU but with higher accuracy) on line at the time of scoring.
- Command USERWEIG:
  - with WHAT(3) > 0. makes all fluences and yields scored by USRBIN, USRBDX, USRTRACK , USRYIELD, USRCOLL to be multiplied by a user-written function FLUSCW at scoring time,
    when USRBIN is used to score **tracklength**
  - with WHAT(6) > 0. makes all energy and star densities scored by SCORE, USRBIN to be multiplied by a user-written function COMSCW at scoring time,
    when USRBIN is used to score **dose or stars**
- For details concerning these conversions please refer to the lecture covering the user routines

Built-in Conversions and AUXSCORE

For some quantities, there is the possibility to get built-in conversions, without the need for user routines: done with generalized particles:

DOSE (obvious..) in GeV/g

SI1MEVNE Silicon 1 MeV-neutron equivalent fluence

HADGT20M Hadrons fluence with energy > 20 MeV

DOSE-EQ Dose Equivalent (pSv)

The set of conversion coefficients used to calculate DOSE-EQ can be selected by the user among a list (see manual) with AUXSCORE:

```
USRBIN      11.0    DOSE-EQ     -40.0      10.0                 15.0 TargDEQ
USRBIN       0.0                 -5.0     100.0                200.0 &
AUXSCORE USRBIN                          TargDEQ TargDEQ             AMB74
```

Scores equivalent dose by folding the particle fluences with the “AMB74” conversion coefficients

WARNING : no coefficients available for heavy ions !!!

Standard Postprocessing Programs

- To analyze the results of the different scoring options, several programs are available
- The most powerful ones are kept in $FLUPRO/flutil.
- They assume that the **estimator files are unformatted**, and can calculate standard deviations and integral values over many cycles:
  - ustsuw.f to analyze USRTRACK and USRCOLL outputs
  - usxsuw.f to analyze USRBDX outputs
  - usysuw.f to analyze USRYIELD outputs
  - usbsuw.f to analyze USRBIN outputs
  - usrsuw.f to analyze RESNUCLEi outputs
- Each of these programs (except usbsuw) produces three files:
  - a text file with extension *_sum.lis* which contains averages, standard deviations, **cumulative (integral)** quantities
  - an unformatted file which can replace the *N* unformatted estimator files and can be used for further calculations
  - a text file with extension *_tab.lis* to be easily readout by graphic codes

Simpler programs are also provided in the manual, as guides for users who would like to write their own analysis program.

Standard Postprocessing Programs

- Example of tab.lis for usrbdx

```
# Detector n: 1 Sp2ChH (integrated over solid angle)
# N. of energy intervals 40
1.000E-03 1.259E-03 1.343E-03 4.688E+01
.
..
# double differential distributions
# number of solid angle intervals 3
# 0.000E+00 2.094E+00 2.094E+00 4.189E+00 4.189E+00 6.283E+00
#
   1.000E-03 1.259E-03 4.337E-04 5.493E+01 2.077E-04 9.900E+01
   0.000E+00 0.000E+00
   1.259E-03 1.585E-03 2.360E-04 6.883E+01 0.000E+00 0.000E+00
   5.481E-04 9.900E+01
.
.
```

- First comes the angle-integrated quantity then the limits of the angular bins, then the double differential distribution

		In 1st ang. bin		In 2nd ang. bin		In 3rd ang. bin	
Emin	Emax	result	error	result	error	result	error

Flux/Fluence: A Common Confusion

- The term Flux is often used, sloppily, to indicate a vaguely defined quantity visualized as "a flow of particles through a surface".
- But Flux is defined by ICRU as dN/dt (particles per unit time). [Where? For which purpose? It looks like a very useless quantity, and is not a "flow"]
- What we really need is a quantity that is proportional to effects such as induced activity, dose, radiation damage. These effects are proportional to the *number of interactions* in a given volume: a "flow of particles" is not what we need!
- The number of interactions in a volume is equal to the number of mean free paths travelled by the particles in that volume: therefore it is proportional to the total particle path length. The quantity

$$\Phi = \lim_{\Delta V \to 0} \frac{\Sigma_i L_i}{\Delta V} \qquad [\mathrm{cm} \times \mathrm{cm}^{-3} = \mathrm{cm}^{-2}]$$

is called Fluence

although its "official" definition is $dN/da_{\perp}$ with *N* being the number of particle crossing an element of surface *da* PERPENDICULAR to the particle direction. This definition is equivalent but hides its actual physical meaning.

Flux/Fluence: A Common Confusion

- Fluence is a point quantity, a function of position (like temperature)
- But we are generally interested on
  - its average over a volume (total track length density divided by the volume: USRTRACK, USRBIN)
  - its average over a surface (USRBDX)
- How can a track length be calculated on a surface? Imagine the surface to have an infinitesimal thickness *dt* : a particle incident with an angle θ with respect to the normal to the surface will travel a segment *dt/cos* θ. Therefore, we can calculate an average surface fluence by adding *dt/cos* θ for each particle crossing the surface, and dividing by the volume *S dt* (*S* being the area of the surface)
- Fluence is DIFFERENT from CURRENT across a given surface: $I=dN/da$
- Φ is independent from *S*, *I* is NOT!
- The interaction rate on a given surface is proportional to Φ, not to I
- NOTE: If the path-length is measured in units of mean free paths $\lambda = 1/\Sigma$, this expression leads naturally to the density of collisions $\Sigma\Phi$

Formal Equivalence of Fluence Definitions

- If *dA* is the surface of the ICRU sphere of cross-sectional area *da*, then of course is *dA=4da*
- It is known that for a convex body the mean chord length is *L=4V/A*
- Therefore, according to the ICRU definition:

$$\Phi = \frac{\mathrm{d}N}{\mathrm{d}a} = \frac{4\mathrm{d}N}{\mathrm{d}A} = \frac{4\bar{L}\mathrm{d}N}{4\mathrm{d}V} = \frac{\bar{L}\mathrm{d}N}{\mathrm{d}V}$$

$\bar{L}\mathrm{d}N$ is the total chord length of the *N* particles crossing the sphere *(Proof from the book of I. Lux and L. Koblinger, p. 24. A different demonstration can be found in A.B. Chilton, Health Phys. 34, 716 (1978) and 36, 638 (1979))*

- But although the two definitions are equivalent, that of ICRU hides the fact that Fluence is a measure of the concentration of particle paths in an infinitesimal element of **volume** around a space point
- And the more *cm* travelled in that volume, the more are the interactions! (Or the potential interactions, if in vacuum)