A Short History and Acknowledgment

The origin of Cosmos can be traced back to even some 2? years ago (as of 1997; but not to the Big Bang) when I wrote a world-first simulation code for cosmic ray propagation in the atmosphere in an assembler language. The computer had only some 60-kB memory and the speed was 0.005 MFLPS (Can you believe that?)

After some blank years since that time, the simulation was revived in the late 1970's as a Fortran code on a main frame. S.Torii remarkably contributed in this period to the development of the program code. The treatment of multiple production and others were very crude, although we already included heavy irons and their break-up process. Some introduction was written in the proceedings of 1979 Kyoto Cosmic Ray Conference.

The code was gradually improved in accord with the Mt. Fuji and Mt. Kambala (in Tibet) emulsion chamber experiments. For example, the QCD jet production process was included by L. Ding from China.

Essential improvements were performed from late 1980's when the Kamiokande experiment started observing atmospheric neutrinos. We included codes developed for accelerator experiments at low energies: Among them, Lund Monte Carlo code Jetset, Fritiof, and Nucrin, Hadrin etc. The ad hoc multiple production model at high energies was also improved; say, the exact four momentum conservation, muon polarization and the geomagnetic effect. M. Honda's contribution in calculating atmospheric neutrino flux is great in this period.

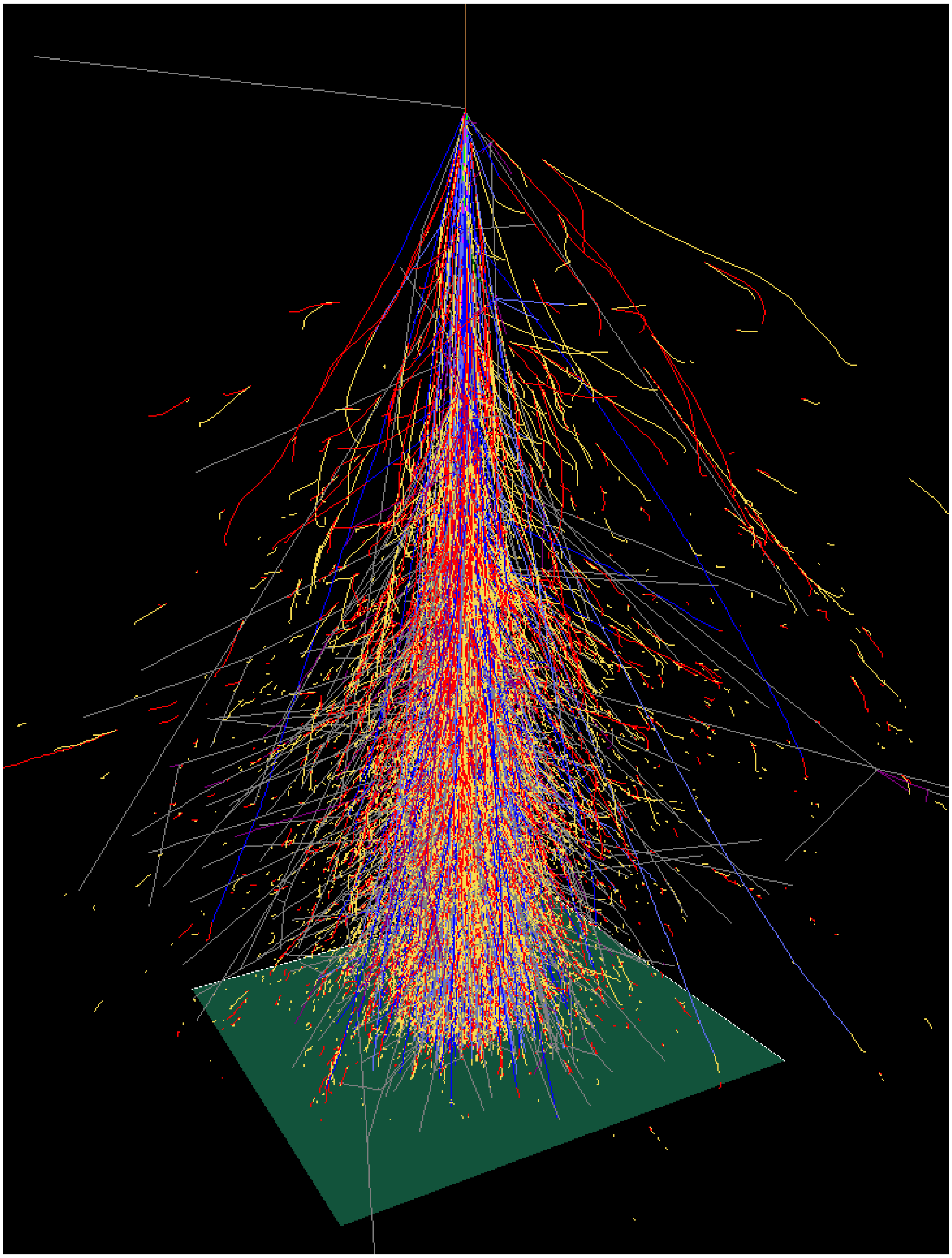
The main frame version was essentially rewritten in 1995 and Cosmos has been widely used in UNIX environments. C. Zhang of IHEP China did a lot of works for detecting bugs at the initial stage of the porting. Lots of bug fixes and improvements owe to many users from that time.

In the spring of 2001, we introduced a new interaction model, dpmjet3, Dr. S. Roesler, one of the authors of which helped me so much for its implementation.

On this occasion, I would like to express my sincere thanks to all those who have concerned with the development of Cosmos.

This work is partly supported by a Grant in Aid for Scientific Research (C) from the Japanese Ministry of Education, Science, and Culture.

February 24, 2002
K.Kasahara



The charged particle track image of a shower produced by a 20 TeV He. The square is 10 km $\times$ 10 km at 4300 m a.s.

The Users Manual of Cosmos

K. Kasahara*

February 24, 2002

Contents

*Department of Electronic Information, Faculty of Systems Engineering, Shibaura Institute of Technology, Ohmiya, Saitama, Japan

1 Major differences from older versions

This section is for the users who have been using Cosmos. The last manual was based on uv5.03 and the current one on uv6.30.

- The code was extended to be able to treat the AMS (Alpha Magnetic Spectrometer) type observation. During such a process, other normal observation scheme became somewhat unstable. These bugs were corrected now.
- The geomagnetic data is now updatable simply by appending the data found in `http://www.ngdc.noaa.gov/seg/potfld/magmodel.shtml` to *Cosmos*`/Data/Geomag/igrf`. For the wmm data, you will need some modification (to insert spaces).
- The user can implement his/her own atmospheric model.
- The old skeleton-flesh method is found unstable. This bug cannon be corrected without rewriting some of the interaction codes; however, it is almost impossible to do so for imported codes. Therefore, a completely new skeleton-flesh method has been implemented.
- Decay in flight is more accurately treated by considering energy loss during flight time.
- Some new platforms are added to the list of Cosmos play ground. They are: PC-Linux, Alpha-Linux, SGI, Mac OSX.
- A new hadronic interaction code, dpmjet3, has been implemented.
- The usage of an input parameter, IntModel, is changed; interaction model selection became easier for any combination of energy regions and interaction models.
- 3D display of charged particle tracks by Geomview has been supported systematically. You can add an air shower array in the view.

2 Introduction

Cosmos is not the cosmos. It is a versatile Monte Carlo code for simulating propagation of cosmic rays in the atmosphere and near earth region (up to ~ 10 earth radius). It can treat low energy phenomena such as atmospheric neutrino problems taking the muon polarization into account, or very high energy air showers in the GZK cutoff region with magnetic bremsstrahlung and pair creation effects and/or the LPM effect.

3 Where is it?

You can get it by using World-Wide Web. You can find the cosmos location after you are connected to

`http://eweb.b6.kanagawa-u.ac.jp/~kasahara/`

4 Platforms you can play Cosmos

First of all, Cosmos is written in Fortran77 (not C) with substantial extensions to it (a small C code is also included). On some platforms, f90 is used to get a correct object. The following platforms and compilers have been verified to work:

- HP9000 series with the HP Fortran.
- SUN O.S 4.x with the SUN Fortran.
- Solaris with the SUN Fortran.
- IBM AIX work stations with the IBM Fortran 90 compiler.

  Note, however,

  - you need the PK preprocessor to accommodate the structure construct used in the Fortran 77 extension. (Recent IBM compiler seems not to need the preprocessor).
  - The namelist data style is little bit different from the Cosmos standard (comma is needed to separate items).
- DEC ALPHA (Digital unix) with the DEC Fortran.
- Intel PC Linux with the Absoft Fortran (f90 is used)
- Itenl PC Linux with the Intel Fortran Compiler (f95)[1]
- Alpha Linux with the Compaq Fortran.
- Alpha Linux with the object code made by DEC Fortran.
- NEXTSTEP with the Absoft Fortran.
  - The Lund nuclear interaction code, Fritiof, and dpmjet3 cannot be used so that the user must give parameters to avoid using them.
- Intel PC Free BSD environments. You need, however, Intel PC Linux for compilation.
- Mac OS X with the Absoft fortran compiler.
- SGI with the SGI fortran.

[1] As of Feb.1, 2002, the compiler has a bug and cannot compile VAX type structure constructs; the bug is being eliminated soon and the coming new versions will comply with Cosmos, though a lot of warnings will be issued. However, it is known at this moment, the −g compiler option must be given otherwise a subroutine in dpmjet3 leads to the death of the compiler).

4.1 Platforms you cannot play Cosmos

- Convex. The Fortran compiler cannot cope with the extensions used in Cosmos.

No body has tried the Windows NT environment, although the MS Fortran is expected to be fully compatible with the Cosmos code.

5 Installation

For installation of Cosmos, a unix environment is indispensable and you needs a Fotran77 compiler with some extensions or Fortran 90/95. A very small C code is also included in Cosmos. The c-language preprocessor, cpp, is also indispensable. Unfortunately, the f2c converter and GNU g77 cannot be employed.

The extensions to Fortran 77 used in Cosmos are those compatible with the DEC Fortran extensions and are common to almost all current Fortran 77 compilers (support of the *VAX type structure construct* is essential).

Fortran 90/95 compiles are expected to work if they are a product of the companies which have supported the structure construct as a Fortran 77 extension.

If your "make" program has some problems during compilation, "GNU make" will settle them.

The installation of Cosmos is simple. The first task is to create a subroutine library. To do so,

1. Get the Cosmos.uv?.tar.gz, where ? denotes the version number.

2. Unpack it somewhere **in an empty directory** (such as Cosmos).

   (Say, by "`zcat Cosmos.uv?.tar.gz | tar xvfp -`")

   We shall express this top directory as "*Cosmos*".

3. Find an appropriate site.configXX and do

   `cp  site.configXX   site.config`

   For example, if you are going to use Intel PC Linux with Absoft Pro-Fortran, you may simply copy site.configPCLinuxProFort to site.config. If you want to use Intel Fortran Compiler, copy site.configPCLinuxIFC. For DEC Alpha, you may probably need site.configDECALPHA2. **Note that you must not change a Makefile in any directories**

4. For temporal assignment of two environmental variables, COSMOSTOP and COSMOSINC, and for adding *Cosmos*/Scrpt to the command search path, issue

   `source Scrpt/sevi`

5. Issue

   make clean; make

   You will need some few tens minutes for compilation on 600 MHz Digital Unix. Depending on the platform, there may appear a lot of warnings when stuff in the Import directory are compiled. With the HP fortran, there may appear other warnings which are related to the 'save' statement of the record construct. Digital unix Fortran will issue a huge amount of warning messages, if you have already a cosmos library.

   If you want to make two or more libraries for different machine architectures in the same Cosmos, choose an appropriate site.config as you did for the first "make", and issue "make clean; make".

   Then you will get the library in a directory specified in site.config (For example, *Cosmos*/lib/PCLinux/libcosmos.a)

6. On some platforms, it may happen that the C code in *Cosmos*/Manager/Ccode/kgetenv.c cannot be put in the library. (You can verify it by `nm libcosmos.a | grep kgetenv.o`). If kgetenv.o has not been included in the library, you have to put it manually. (Say by ar -r libcosmos.a kgetenv.o. You may also need ranlib libcosmos.a on the BSD-like system)

6 Setting up your environment

If you are going to use Cosmos frequently, it is better to setup the two environmental variables mentioned above and to add *Cosmos*/Scrpt to you command search path. For example, add the following in your .cshrc

- `setenv COSMOSTOP ~/Cosmos`
- `setenv COSMOSINC  $COSMOTOP/cosmos`
- `setenv PATH    $COSMOSTOP/Scrpt:$PATH`

7 Upgrading Cosmos

If you are going to upgrade Cosmos in accord with a release of a newer version, you may normally obtain a relatively small file which is denoted e as, for example, `Cosmos.uv4.01Diff.tar.gz`, and unpack it in *Cosmos*. After this, normally you may do 'make clean; make' . However, it is advised to delete the library once before doing so. If the Diff file is not appropprieate, get the entire source file.

8 First Kiss to Cosmos, a test run

It is time to confirm your setting by making a Cosmos executable and run it:

- Be sure `COSMOSTOP` and `COSMOSINC` environmental variables are assigned, and *Cosmos*/Scrpt has been added to your command search path.
- If you are using the IBM AIX system, issue[2]

  IBM
- In UserHook/FirstKiss issue,

  `make clean; make`

  If everything is good, an executable, cosmos`$ARCH`, will be created, in FirstKiss, where `$ARCH` denotes your machine architecture (say, PCLinux, NEXT486 etc). We shall, however, use 'cosmos' without adding such a flag.
- Issue,

  `mkdir /tmp/$USER`

  if such a directory does not exist. (`$USER` is your login name)[3]

[2] This is needed once for all to make the Namelist format compatible with the IBM style. This will replace some files by files specific for IBM in *Cosmos*/UserHook and *Cosmos*/Data/Namelist. The original files will be kept with the ".Orig" extension. To restore the status to the original one, use the "Orig"
command. This command may not work well in the recent versions. In that case, you may adjust the namelist data manually.

[3] This is the directory in which trace data (particle tracking data) is stored. You can change the dirctory by giving an explit directory to "TraceDir " in the "param" file (say, "./" for the current directory). The file name will be trace1, trance2, ... for event 1, event 2...

- Issue,

  `cosmos < param`

  Then, you will see some output on the screen (they are particle information observed at a given level specified in the 'param' file). **Congratulations!** (May they all be not error messages.)

- You can display the trace of particle tracks by using gnuplot or by using Geomview. A simple method for gnuplot should be seen in the output mentioned above. We recommend use of Geomview which gives you far better user-interface and much beautiful real 3-D output than gnuplot. If Geomview is already in your computer, set an environmental variable GEOMVIEW to be the path to the Geomview executable binary, go to *Cosmos*/Util/Geomview, and issue,[4] ./disptracebygeomv /tmp/$USER/trace1 You would see no trace view after successful invocation of Geomview; you need to push the "Look" button. For further details, see a separate manual.

  `/tmp/$USER` is to store data for the trace and is not used in the normal jobs. The standard trace data consists of the following lines.

```
x  y  z code  E  charge
x  y  z code  E  charge
x  y  z code  E  charge

x  y  z code  E  charge
x  y  z code  E  charge
x  y  z code  E  charge
x  y  z code  E  charge
 ...
```

  where 'code' is the particle code (1-photon, 2-electron, 3-muon, 4-pion, 5-kaon, 6-nucleon, 7-ν_e, 8-ν_μ. For further details see Sec.13.2). Consecutive data lines express a continuous track line. Two blank lines are the separator of different tracks. **You must remember that taking trace information consumes a lot of disk space and cpu time so that you would normally not want trace information**; be careful that "param" used in the FirstKiss has a specification for trace output.

9 Running your own job

9.1 Overview

Cosmos, in default, does not produce anything on the standard output. Error, warning or some other informational messages may appear on the standard error[5]. Therefore you have to write a fragment of codes (user-hook routines) for getting necessary quantities yourself.

For a particular application, you may

- copy Template (or some other directory) to make another appropriate directory (let's call it MyJob) in UserHook.

- Modify chook.f in MyJob. Normally you need to modify this file only.

  The chook.f file contains the template of all most all the user-hook routines with predefined names; they are called from Cosmos at appropriate timing.

[4] Note the scripts here assume 'tcsh' which is located in /usr/local/bin/. The NEXTSTEP user must copy '.geomview' file into the home directorybefore doing so.

[5] Messages from the codes in the Import directory may appear on the standard output in some case.

- Suppose you modified chook.f so that it can generate output you want. Then, in UserHook/MyJob, issue,

  make clean; make

  to make an executable, "cosmos", there[6].

- Set up a primary spectrum data in MyJob/'primary' and give PrimaryFile = 'primary' in the namelist 'param'.

- Give other data in the namelist.

  See for details, section9.4.

- Issue

  ```
  cosmos <  param
  ```

  Output is up to you. If you use standard output, you can redirect it to a file like,

  ```
  cosmos < param >  output1000TeV
  ```

To be able to set up input data to Cosmos, you have to learn the following .

9.2 Unit of physical quantities

The unit is based on SI. The following units are used internally and you will obtain outputs from Cosmos in those units in principle. It is also very important to note that **all real variables are given in the double precision**; you are recommended to output them in the single precision to save the disk space (for gnuplot, it is essential).

Length: m.

Energy: GeV. Note, however, you can specify the energy of primary particles in a variety of units, such as MeV, MeV/n, TeV etc or in momentum.

Magnetic field strength: Tesla. Note that 1 Gauss is 10^{-4} T.

Thickness of air: kg/m^2. 1 g/cm^2 is 10 kg/m^2. The air density is in kg/m^3.

Time: sec. However, time in the chookObs routine is already converted to nsec. The default Čerenkov output contains time factor in (length in cm)/beta for saving output space.

Angle: Angles in degree are used to specify the latitude and longitude of the observation place. The declination angle is also in degree.

9.3 The coordinate system

9.3.1 The E-xyz system

Figure 1 illustrates the basic coordinate system which is internally used in Cosmos. The x-axis is directed to the longitude 0 and latitude 0. The y-axis is to the 90 degrees east and the z-axis to the north. The Earth is expressed by a complete sphere. The origin is at the center of the Earth. This coordinate system is abbreviated as the E-xyz system.

[6] As usual, $ARCH is added to the actual name such as like "cosmosPCLinux".

9.3.2 The detector and the primary system

You may setup several observation levels, at each of which you can record particle information passing through it. The plane of the levels may be horizontal or perpendicular to the primary particle direction. Since detectors are normally placed horizontally, we call the horizontal recording system the "detector system". (See Fig.2). From the figure the user might misunderstand that partiles are recorded when they cross a plane which is tangential to the surface of a sphere (this was acutually true once in older versions), but it is the surface of a sphere, i.e., particles are recorded when they cross this spherical surface. **However,** the $x - y$ plane of the coordinate system is tangetial as it is the rectangular system. The x-axis of the system is directed to the magnetic east (default), the y to the magnetic north and the z to the vertical.

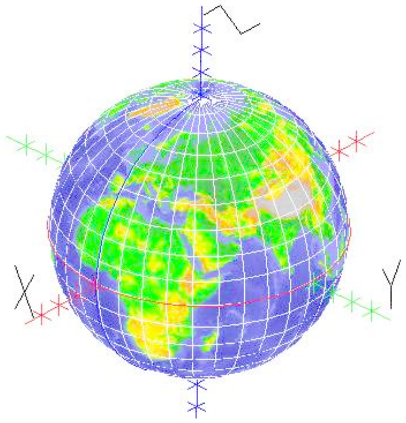


Figure 1: The basic coordinate system

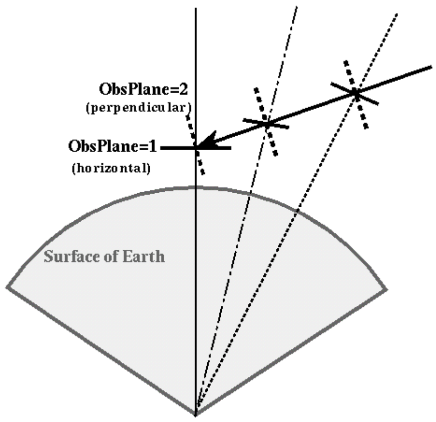


Figure 2: Geometrical information of the Observation Plane

The coordinate system whose $x - y$ plane is perpendicular to the primary is called the "primary system". Hence the z-axis is the primary direction (upward is positive), the x-axis is directed to the vector product of the z and the vertical direction, the y to the vector product of the z and the x-axis direction.

In both cases, the origin is assumed to be the location of a specified observation place.

If the primary direction is the vertical, both systems are almost identical, as far as near $z-$axis particles are concerned.

9.4 Setting up input data

In any Cosmos applications, you must prepare at least two files:

- One containing the primary spectrum data as explained in *Cosmos*/Data/Primary/sample.d. All details are contained there. We note here that
  - you can describe any spectrum shape as far as it can well be approximated by a maximum of 60 segments of lines (default) (mono-energetic beam can also be given, of course).
  - Any type of particles defined in Cosmos can be a primary beam. A maximum of 8 components can be specified. Therefore you may inject unrealistic primaries such as muons for a special purpose.
  - The spectrum may be given in kinetic energy, total energy or momentum. The energy or momentum unit may be one of eV, keV, MeV, GeV, TeV, PeV or EeV (c=1). Further you may give the scale by per nucleon or per particle. The flux values can be given by a form of (true flux)$\times E^X$, where E should be the same one as that you give for the

energy scale. (That means, if you give the energy scale in MeV, E should be also in MeV; it cannot be GeV and so on.) The flux may be given either in a differential or integral form.

  – You can limit the energy range where the actual energy sampling is performed. For example, for a table which containes energy region from 1 GeV/n to 10000 GeV/n, you can sample the primary from 10 GeV/n to 10000 GeV/n or from 10 GeV/n to 100 GeV/n.

- The other for namelist data.

  In this file, you have to give a file name in which the primary spectrum is described, the details of the primary direction, observation levels, number of events to be simulated etc.

Some applications, for example, one which needs a magnetic cutoff data around the Earth, requires an additional file and must be specified in CutOffFile in the namelist.

The description of the namelist data is given in Appendix and also found in ParamUsage1 and ParamUsage2 in the Doc directory. The default values of the parameters can be seen in ParamDefault in the same directory. **In normal applications you may give only the parameters explained in ParamUsage1.** For further details, see case studies in section10.

9.5 User-hook routines

A template of the user hook is chook.f in the *Cosmos*/UserHook/Template directory. In normal cases, you would simply modify the following few routines:

chookBgEvent : You may output some header information for each event. This is called when the system initialization for one event is completed.

chookObs : This routine is called whenever a particle crosses a predefined observation level. Then, you may record particle information (say, energy, particle id, position etc). For more details of the particle information, see the chookObs subroutine.

chookEnEvent : This is called whenever one event generation is finished. If hybrid air showers are generated, they should be output here.

You will need some flag to identify each event easily.

10 Examples of input data

10.1 Observing high energy particles at high mountains

The conditions and parameters to be set are as follows.

- You want to record particles at vertical atmospheric depths of 5400 and 6060 kg/m^2. For this you may give,

  DepthList = 5400, 6060, 0, ...

- You want to inject primaries in the zenith angle region from 60 degrees to vertical. You may specify the cosine of the angles as,

  CosZenith = (0.5, 1.0)

  Note that this zenith angle is the direction of primary particles measured at the deepest observation level (i.e., at 6060 kg/m^2). At other depths, the value is not strictly the same.

- You want to simulate 10000 events.

  DestEvnetNo = 10000

  If you give 'DestEvnetNo = 10000 1000', the program run will stop at the completion of 1000 events. You may continue the job with 'cont=t' function (see below) for another 1000 events, and repeat such a process until all 10000 evnets are finished.

- You want to generate electro-magnetic cascade besides default hadronic cascade.

  Generate ='em'

- You want to specify the initial random number of the first event,

  InitRN = 12345, 77777333

  If you put a negative number for the second value, like 'InitRN =12345, −77777333', Cosmos will use a timer value and the host name to make the initial seed.

- You want to record particles of kinetic energy greater than 1 TeV.

  KEminObs = 1000

- The place of the observation is default. I.e., YangBajing in Tibet. You need not redefine LatitOfSite and LongitOfSite.

  If you like another place, give the latitude and longitude there. These are used to calculate geomagnetic field there. They are also used if you specify the primary particles from a point source. You need not worry about this in many applications.

- You want to record particles with the "detector system".

  ObsPlane = 1

  (This is default.)

- You want to use a primary spectrum data in *Cosmos*/UserHook/MyJob/primary.

  PrimaryFile='primary'

  Instead, you may give the full path to the file.

- You want to record relative arrival time of particles.

  TimeStructure = t

  Note that the relative arrival time is in nsec. Time, 0, is defined by the time a gamma ray arriving at the center of the detector without any interaction would record. In the case of ObsPlane = 1, some time data will be negative. If ObsPlane =2, negative time would not happen within a computational error ($\sim 10^{-2}$nsec).

At the end of the job, Cosmos always write a file which may be used when you continue the job. It is a file specified by ContFile. The default is "ConfInfo". If you give Cont = t in the namelist ('param') in the next job, this file is read and all parameters in the namelist you give will be overwritten by the information in the ContFile and you can safely continue the job. However, in normal job, Cosmos dose not try to append data to the files. It's your task.

10.2 How to observe air showers together with high energy particles

- You want to know the air shower size at each observation level in the previous example.

  Generate = 'as/em'

  ASDepthList = 5400, 6060, 0, ...

  If you follow the particles down to 1 MeV or so, you can get air showers. However, this is a time consuming process. For example, with a 200 MHz processor, you would need ∼1 hour to follow particle down to 1 MeV if the primary enery is 10^{14} eV.

  If you want to know only the air shower size, you need not follow each particle to such a low energy. This is managed by a hybrid method; Cosmos computes the size of each component air shower produced by an electron in the electro-magnetic cascade. The calculation is based on so called Approximation-B.

  The minimum energy of particles followed in the system is automatically adjusted. However, the chookObs routine is called only for those particle of which the energy is larger than

KEminObs. It will be important to set KEminObs to a large value (say, 10^{30} eV) if you need only air size but not each particle information for faster execution of the job.

You have to output the air shower size of each event in the chookEnEvent (i.e, when one event simulation is completed). An example is given in chook.f. You need some flag to identify the data.

When using the hybrid method for obtaining the air shower size, there are other important parameters you have to know: **WaitRatio, Ecrit** and **RatioToE0**. Also you have to realize that the hybrid air shower size is biased. These are explained in section 10.3.

10.3 Air shower transition curves

You want to construct air shower transition curves but don't need each particle information. We assume that the primary particle is a high energy gamma ray or electron. Here we list essential parameters only.

- You want to know the air shower size at every 1000 kg/m^2.

  ASDepthList = 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000, 0, ...

- Although you need not each particle information, you have to specify the deepest observation level to which the particle should be followed.

  DepthList = 10000, 0, ...

  Also you have to give a very high minimum observation energy so as not to let the system follow the particles to a very low energy.

  KEminObs = 1.d20

  The actual minimum energy is adjusted in the system.

- Generate ='as'

  You should not give "em" in Generate. The minimal e-m cascade is automatically developed in the system.

- You want to include the natural fluctuation.

  WaitRatio = 0.01

  When the minimal e-m cascade is developed in the system, the size at a specified depth is calculated whenever an electron (positron) of a certain energy range makes an interaction, assuming it develops into an air shower as calculated by Approximation-B. Such a size is summed up to obtain the total number of electrons at each depth. Once the size is calculated for an electron, it is discarded in this case. (If Generate has "em", all its descendants may be followed until their energy reach the minimum energy but no air shower is generated from such descendants.) The value of WaitRatio controls the timing of air shower generation. If the electron energy becomes lower than WaitRatio×(the primary energy), the size is calculated. That is, the system waits until the electron energy becomes adequately low. If you set this to 1.0 (default), the fluctuation would be too small. (Suppose gamma ray primaries. The fluctuation is produced only by the pair creation; its path length and energy partition to the pair.)

  In the case of hadron primaries, you may use the default value (1.0) since there will be a lot of gamma rays[7].

- `RatioToE0` is used to determine the minimum energy of hadronic interactions by $RatioToE0 \times PrimaryEnergy(/nucleon)$. Normally $10^{-5} \sim 10^{-6}$ is enough.

[7]This method should not be used for primaries less than ∼1 TeV Because Cosmos neglects air showers generated by electrons of energy below 1 GeV. For primary energies of ∼300 to ∼2000 GeV, you should use higher values such as WaitRatio=0.1 ∼ 0.05 and must be careful about whether fluctuation is adequately incorporated.

If the primary is nucleus, you should consider the per nucleon energy here.

- `Ecrit` is the critical energy which is employed only when calculating air shower size in the hybrid air shower generation. The value would be dependent on the experimental purpose. The default value, 81 MeV, is bit too small in many applications (The air shower size is overestimated). Comparisons of sizes by the hybrid method and by the full Monte Carlo tell that

  N_e(full 3-D M.C)< N_e(hybrid AS with E_c = 81 MeV)< N_e(full 1-D M.C) $\approx$ N_e(by hybrid AS with E_c = 76 MeV).

  at around shower maximum. At shallow depths, however, we have

  N_e(full 3-D M.C)> N_e(hybrid AS with E_c = 81MeV)

  This is due to particle scattering. Therefore, the user must calibrate hybrid air shower size to match his/her purpose.

10.4 Make skeletons first and flesh them out later

The method described here is actually not applicable due to the following fact: We get an event by giving a some random seed for the first event. After generating some number of events, we may give the same random seed as the first event, then we should get exactly the same event as the first event if every thing goes well (the skeleton-flesh method here assumes such a good behaviour). Unfortunately, this is not always the case for some interaction code; the event generation seems to be dependent both on the seed and history. So we developed another way which is described in UserHook/SkelFlesh. However, the idea of skeleton-flesh can be explained using now obsolete method. For actual applications, see UserHook/SkelFlesh/Readme.

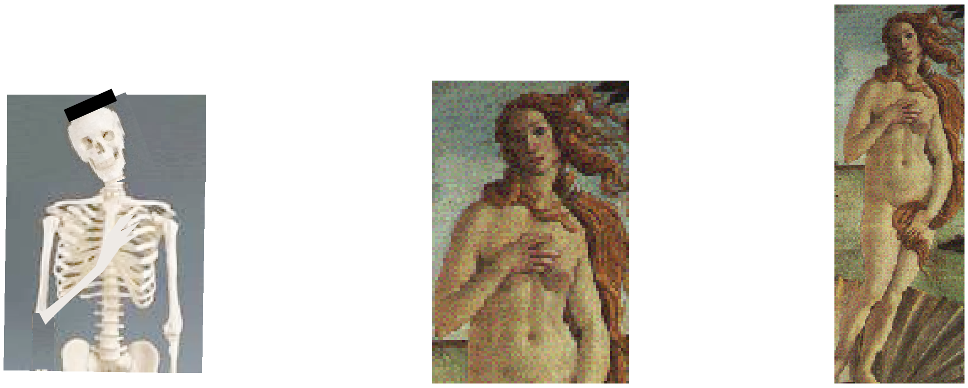


Figure 3: You can make a skeleton first (left), and if you like it, you can flesh it out (middle). You can even add your most interested lower part which is missing in the skeleton (right)

In section10.1, you learned how to observe each particle, and in section10.2 you could set parameters to observe each particle and an accompanying air shower simultaneously. However, the generation of air showers needs rather long time. If you are interested only in those events which have very high energy particles (say, > 1 TeV), the simultaneous simulation like section10.2 is not an effective way, since only a fraction of the events will have very high energy particles at the observation level.

Cosmos enables you to make skeletons first quickly and to flesh them out later by selecting events you love. You can mix the following three in an arbitrary way.

1. Adding air showers.
2. Generating electro-magnetic cascade showers when the skeletons have only hadronic showers.
3. Following particles to a lower energy.

4. Following particles to deeper depths. (Fig.3(right)).

The skeleton may be created by setting a high minimum energy (the event may or may not include electro-magnetic cascade showers). We list some essential parameters only.

- You want to see only high energy particles at first.

  KEminObs = 1000

- You want to see them at a high mountain level (6060 kg/m^2) at skeleton making time, but you want to follow particles to deeper depths at flesh-out time. (Note that you have to give the deeper depths from the first).

  DepthList = 6060, 7350, 8400, 0, ...

  EndLevel = 1

- You want to make the events skeleton.

  Job = 'skeleton'

- You want to generate e-m cascade too.

  Generate = 'em'

In the skeleton making job, you may compute some quantity by which you can judge that the skeleton be fleshed out or not. Since you are interested only in high energy events, you may compute the energy sum of observed high energy particles at the given depth and if you need the event, "call cwriteSeed" in the chookEnEvent routine. This will write two integer seeds for the random number generator and the event number on the disk file specified by SeedFile (default is "Seed"). The job also creates a copy of the parameter file as SkeletonFile (default is "SekeltonParam") which is referred by Cosmos at flesh-out time.

After completing the job, you can flesh out the events. You have to modify the SkeletonParam file as below.

- You want to flesh the events for which you saved the random number seeds in SeedFile.

  Job ='flesh'

- You want to follow particles to a lower energy and to a deeper depth.

  KEminObs = 300

  EndLevel = 3

- You want to add air showers.

  Generate = 'em/as'

- You have to specify Job='flesh' in the namelist parameter file (let its file name be 'param') used at the skeleton making job.

Then, you may run Cosmos as
cosmos < param

10.5 Primaries from a point source

Cosmos has convenient parameters to simulate observation of cosmic rays from a point source.

- Specify an observation location by LongitOfSite, LatitOfSite and DepthList. Instead of giving DepthList, you may give HightList and make the corresponding value of DepthList negative. (If a value in DepthList is negative, the corresponding term in HeightList has priority.)

  You have to also give DtGMT, the difference of local time from the GreenWich Time (in hour, not in sec).

- Give the declination of the point source. For example, for the Crab nebla,

  SourceDec = 22.6

- You want to observe it in the zenith angle region of 30 degrees to the vertical.

  CosZenith =(0.8660254, 1.0)

  Note: This does not necessary mean that some particles actually come from very near to the vertical in the simulation. In the worst case, if a given point source cannot be seen in a given zenith angle region, Cosmos issues an error message and stops.

- You have to specify that the primary is not isotropic but from a point source

  Za1ry ='ps'

  If you want to have primaries from around the point source, you may alternatively give

  Za1ry ='aps'

  and

  Ddelta = 5

  This last Ddelta means that the source is confined in the declination region of SourceDec ± Ddelta.

10.6 Atmospheric neutrino observation

- You want to see neutrinos from all directions.

  CosZenith = $(-1., 1.)$

- You have to give an observation site.

- Cosmos dose not follow particles inside the Earth. Therefore the neutrinos do not cross the observation level if they come from below horizon. To make the situation simpler, you have to set

  ObsPlane = 0

  and

  BorderHeightL = 0

  DepthList = -1

  HeightList = -1

  The last two is to set the observation level at -1 m from the sea level, and avoid a trouble when the primary zenith angle is very close to 90° You observe neutrinos not by judging if they cross the observation level (ObsPlane=0) but by seeing if they reach the height specified by BorderHeightL (0 is default). Note BorderHeightL is in the second group of the parameters, &Hparm.

10.7 Play-ground of particles

Normally, Cosmos particle tracking is performed between BorderHeightH and BorderHeightL, whose default values are both 0's. In that case, BorderHeightH is adusted to be HeightOfInj + 1 (m) and BorderHeightL to be the lowest observation height -1 (m).

10.8 Is an observed anti-proton really extraterrestrial ?

Suppose you observed an anti-proton at a balloon altitude. You may want to verify that it is not originated in the atmosphere. One "must" is to prove that the anti-proton can come from outside or, equivalently, a proton of opposite momentum can escape from the Earth.

- You may give the observed anti-proton as a primary. The observed direction should be given by CosZenith and Azimuth.
- You want to move particle in the reverse mode (given incident particle is charge-conjugated and made to go to a direction opposite to the given direction. All interactions except for magnetic deflection are neglected)

  Reverse=1

  If you give 2, energy gain (not loss) in air is considered.

  Note that to use Reverse !=0, the following point should be considered.
- HeightOfInj = balloon altitude where you observed the anti-proton

  Note this is in m. It's better to give the same height as an observation level. DepthList is by kg/m^2 so you may give a negative value to this and give the same value as HeightOfInj to HeightList.
- ObsPlane= 0

  Observation is at BorderHightH or BorderHightL.
- KEminObs=Lower than a test particle energy.
- TimeStructure=T

  This is to make PathLimit effective.
- Trace=41 or 0.

  If you want to see trace, get it in Exyz system.
- You may give PathLimit; if the particle reaches as far as this distance from the Earth center, Cosmos regards it escapes from the Earth. The default is 20$\times$ Earth radius. (10 will be enough).

10.9 The Magnetic and LMP effects

- You want to examine gamma primary air showers with inclusion of the LPM and magnetic effects. The magnetic effect works even at 5000 km away from the surface of the Earth (for 10^{20} eV)

  HightOfInj = 10000d3
- You are giving 10^{21} eV as the primary gamma energy. If you think the magnetic bremsstrahlung and LPM effects for electrons may not be negligible at 10^{18} eV,

  WaitRatio = 0.001
- You want inject a gamma ray of which the azimuthal angle is directed to the geomagnetic field line.

  Azimuth = $(90., 90.)$
- MagBrem =2 (1 is obsolete)
- MagPair = 1

  The above two are in the second group.

- LpmEffect = t

There are more relevant paramers for detailed control. See, MagBremEmin, MagPairEmin, UpsilonMin and LpmBremEmin.

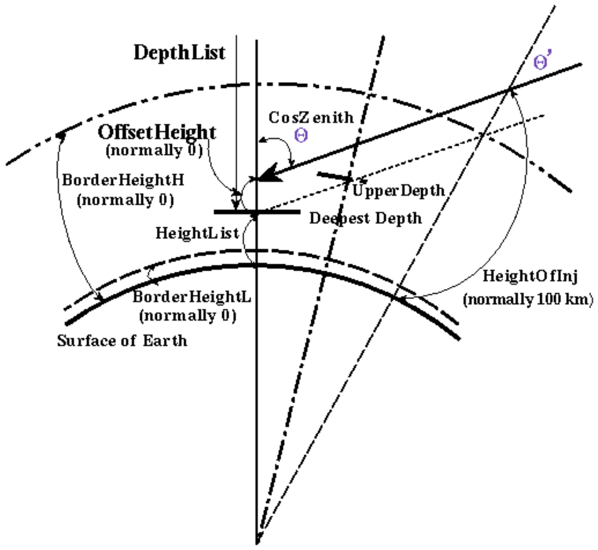


Figure 4: Geometrical information of an incident particle

10.10 AMS type observation

The AMS (alpha magnetic spectrometer) revealed a lot of new features of albedo cosmic rays. This section explains how to make a simulation corresponding to such an observation. The details will also be found in UserHook/AMS. The simulation consists of 2 major steps: Reverse and flesh phases, which use skeleton/flesh function of Cosmos (not a new skeleton/flesh function). The old skeleton/flesh function still works because there is no hadronic interaction in the skeleton making phase.

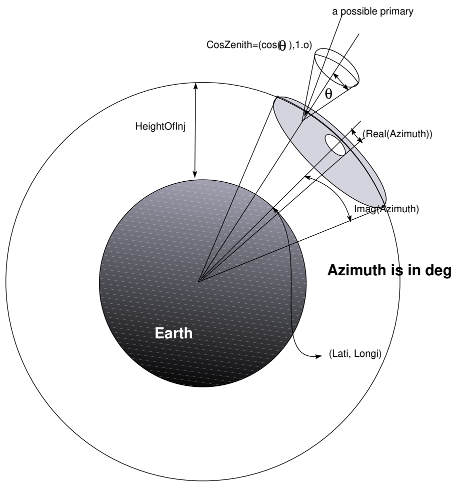


Figure 5: ObsPlane=3 is special

10.10.1 Reverse phase (back-tracking)

The procedures described here may be used to make a rigidity cutoff table at a given place.

To fix the primaries which can enter the earth at a given location, we first assume isotropic primaries at a height of 400 km (HeightOfInj = 400.d3). and sampled primaries are charge-conjugated and the direction is inverted to be traced backward (Reverse= 1); if it reaches to a $10 \times R_e$ height (BorderHeightH = 6500d4, radius of $11 \times R_e$ from the Earth center), it is regarded as the real primary and the random nubmer Seed is recored (In chookEnEvent, cwriteSeed is called). Otherwise, or the flight time exceeds 20 sec, it is regarded not a primary (DepthList= negative, HeightList = 399.99d3, PathLimit = 6.0d9). The seed file for the real primaries are saved for the next flesh phase.

The isotropic primary sampling is performed all over the spherical surface (Azimuth = (0,180), CosZenith = (0.0d0, 1.00d0). LongitOfSite and LatitOfSite are arbitrary. Note the meaning of Azimuth is completely deifferent from the normal cases where Obsplane ! = 3; see Fig.5 The geomagnetic field exists everywhere (HowGeomag = 2). This phase is skeleton making (Job='skeleton"). For the particle tracking, we use always the Runge-Kutta method (UseRungeKutta=6). The maxium step size is 1/10 of the Lamor radius (LamorDiv = 10d0).

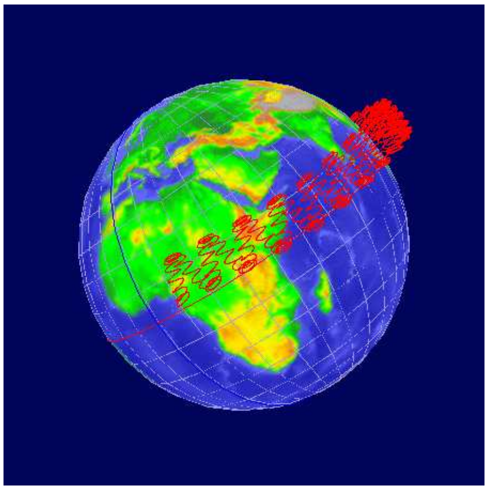


Figure 6: An example in the reverse phase: An anti-proton of energy 1.9 GeV starts upwards at a middle Africa; it bounces between the north and south while drifting to east. It cannot be a primary

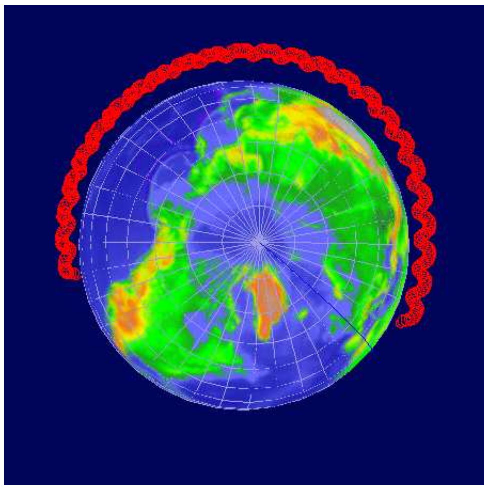


Figure 7: The same one as the previous Fig. seen from the north pole direction

10.10.2 Flesh phase

In this phase, the Seed file created in the Reverse phase is read and the sampled primaries are injected at 400 km. Interactions in the atmosphere, synchrotron radiation (Continuos loss by MagBrem = 1, MagBremEmin = 0.1) are considered and all produced particles are tracked down to the earth surface or to the escape sphere (at $11R_e$), or until the flight time exceeds 20 sec. The neutrinos are recorded at the time of birth if it is directed to the earth and not traced further. Neutral particles going up above 100 km are discarded. The charged particles passing through 380 km are recorded. These are set by modifying a param for the Reverse phase in the following part: DepthList = negative value, 0 HeightList = 380.d3, Job='flesh', BorderHeight = 1 (to follow particle to the surface of the earth), Reverse = 0.

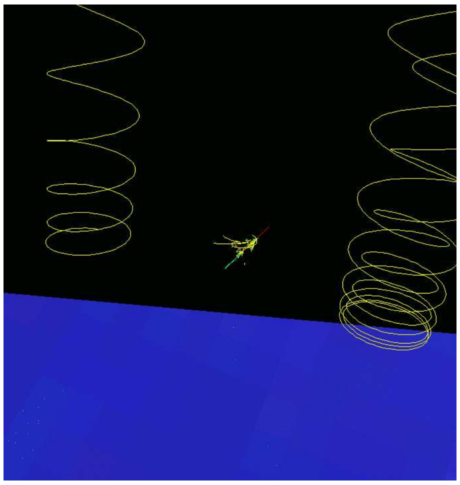


Figure 8: An example in the flesh phase: A primary proton interacts in the atmosphere (center) over Africa, and one of created electron goes up (left).

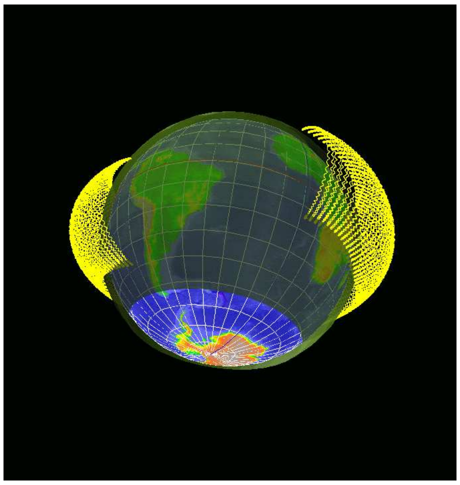


Figure 9: The electron travels long time drifting to east. The shadowed region is the AMS observation region; the electron crosses the height only several times (at near birth place and sink point)

10.11 Getting data for Čerenkov observation

Cosmos does not produce Čerenkov light directly, but you can get charged particle trace information dedicated to the Čerenkov observation use; you can obtain Čerenkov light by using that output.

- You want to produce trace information for Čerenkov data.

  Trace = 61 (or 62)

  This will produce trace information to be used for Čerenkov light observation. The system of coordinate will be the primary system. The user must open the file (this is different from usual trace setting). The fortran logical device number is 21 (=TraceDev). If Trace is 61, you have to open the file as "form=formatted" and if it is 62, the file should be a binary file.

  The trace data of all events is put in the same file (this is also different from usual trace).

  For each event, you get the following

  - Event No, primary particle code, primary energy, three components of direction cosines of the primary at the (deepest) observation level.

    Except for two integers (Event No and code), All are in the double precision.
  - For each of charged particle track segments,

    particle code, charge, energy, time at the segment top (in cm/beta), time needed to travel the segment (in cm/beta), segment top x, y, z segment bottom x, y, z

    All are integers except for energy, x, y and z. **The last one in the event will be all zero.** Thus you can identify one event from another.

  If you put Trace = 71, (or 72), the coordinate will be in the "primary system" except for the z which is in kilo-gramage as in usual trace. For Trace = 81, (82), and 91, (92), you will get the coordinate in "the detector" system in a similar fashion. If the value is even, binary output is made.

  You will obtain only charged particle tracks with energy higher than the Čerenkov threshold which is dependent on the air density.

If don't like this format, you can give a value larger than 100 to Trace, and can manage output at the chookTrace routine

10.12 Čerenkov output on fly

In the previous treatment, all track segment data is put in a disk file which can easily exhausts available disk space. You can get rid of this by giving Trace > 160, in which case you can manage the charged particle track information in the user hook. This means, you can convert the track information into Čerenkov light in your desired form on fly.

For this end, the following template routines are available in ctemplCeren.f in UserHook.

- chookCerenS: called when an event is started.
- chookCeren: called when a charged particle is moved to emit Čerenkov light.
- chookCerenE: called when the event is finished.

You may copy ctemplCeren.f into chookCeren.f and fill the content. (See ctemplCeren.f). The track info coordinate is the same as the one for Trace $\leftarrow$ Trace$-$100.

10.13 Job continuation

Suppose your available disk space is not large while you want to simulate lots of events. One way of overcome such situation is to give

- DestEventNo = 10000 1000

  At first run, you may run Cosmos in a normal way. Cosmos stops when 1000 event simulation is finished. Then, you may analyze the events.

- For the continuation of the job, give

  Cont = t

  in the same parameter file and use it in all the subsequent runs. Cosmos stops whenever 1000 event simulation is finished in each run. This can be continued until you get 10000 events.

10.14 Giving primary particles on fly

Normally, primary particles are sampled by reference to data you give in a file. In some case, however, you may want to generate primary particles in your hook routine.

You can do it by calling the cresetPrim routine from chookBgEvent.

- call cresetPrim(aPtcl, dircos)

where aPtcl is a /ptcl/ record and dircos is a /coord/ record. You must give energy, code, subcode, charge to aPtcl, and direction cos of a particle and direcos.sys='det', i.e., the direction cos should be given in the system where x is directed to south, y to east, z to the vertical. Code fragment will be

```
#include "Zptcl.h"
#include "Zcode.h"

...
call cmkptc(aPtcl, knuc, antip, -1)  ! make anti particle
aPtcl.fm.p(4) = energy              ! give total energy
dircos.x = ..
dircos.y = ..
dircos.z = ..
dircos.sys ='det'
call cresetPrim(aPtcl, dircos)
```

If you would like to give the coordinate in the E-xyz system, you have to use

```
#include  "Ztrack.h"
 record /track/ aTrack
         ...
         call  cresetPrim2(aTrack)
```

and put every track information in "aTrack" (i.e, particle code, subcode, charge, 4-momentum, mass, position, direction cosines, and aTrack.where which is the next observation level the particle is currently directed.)

10.15 Air shower generation by heavy primaries is too slow

For example, if you input a very high energy (E_0) iron as the primary for hybrid air shower generation, you will need very long execution time. You can get rid of such situation by first creating lots of component showers from protons/neutrons of energy ($E_0/56$) and employ such component showers later. An example is shown in chookASbyH.f

11 Parameters for user's use

There are character*100, integer and real*8 parameters of which usage is left for the user. They may be given in the second namelist in the input 'param'. They are arrays with the defualt size, 5, 10, and 10, respectively. The data may be given, for example, as

```
  UserHookc ='abc', 'xyz/ddd/file'
  UserHooki = 1  10  5
  UserHookr = 0.345
```

The $i-$th value in each array of these variables can be retrieved by the following subroutine call.

UserHookc: call cqUHookc(i, cv) where cv is a character variable with length enough to accomodate the data.

UserHooki: call cqUHooki(i, iv) where iv is an integer variable to accept the data.

UserHookr: call cqUHookr(i, rv) where rv is a real*8 variable to get the data.

If "i" is out of range, cv will be blank, iv -9999999 and rv -1.d-60. The user can change the default size by modifying `MAX_USERHOOKC, MAX_USERHOOKI, MAX_USERHOOKR` in *Cosmos*/comos/Zmaxdef.h.

12 About hadronic interaction models

Table 1: Interaction models and usable energy region; Table dose not necessary means the validity of the model

model	Usable at K.E(GeV)	comment
dpmjet3	> 0.5	NEXTSTEP cannot use this
fritiof7.02	> 9	
fritiof1.6	$4 \sim 1000$	NEXTSTEP cannot use this
nucrin	< 4.5	
gheisha	< 1000	be careful.
ad-hoc	> 4	

Cosmos employs several nuclear interaction codes.

The parameter, "IntModel" specifies which hadronic interaction model is used at what energies. At low energies where primary spectrum is well known ($\lesssim$ 100 GeV), calculations by Cosmos tells that the following choice best explains observed secondary cosmic rays (gamma rays and muons);

```
IntModel ='"dpmjet3"'
or
IntMovel ='"nucrin 4.5 fritiof1.6 9 fritiof7.02"'
```

The first choice is the default and denotes the "dpmjet3" is used in the entire energy region.

The second case tells that "nucrin" is used below 4.5 GeV(kinetic energy/n), below 9 GeV Fritiof1.6 and at higher energies Fritiof7.02. The model name and the standard energy region for it are summarized in Table 1.

The standard model that had been used before dpmjet3 is
`IntModel = '"nucrin 4.5 "fritiof" 500 "ad-hoc"'.`

The reference to each interaction model is as follows.

Nucrin/Hadrin: K.Hänssget and J.Ranft. Comp. Phys. Comm. **39**, 37(1986)

Gheisha: See. http://wwwcn.cern.ch/asdoc/geant_html3/node346.html

Fritiof v1.6: B.Nilsson-Almquist and E.Stenlund. Comp. Phys. Comm. **43**, 387(1987)

Jetset 6.3: used in Fritiof v1.6. T.Sjöstrand and M.Bengtsson, Comp. Phys. Comm. **43**, 367(1987)

ad-hoc: Phenomcnological ad-hoc model fitted to accelerator data including SPS UA5 data. Developed for Cosmos by K.Kasahara. The basic idea is similar to the UA5 simulation code.

Fritiof v7.02: a Lund code by Hong Pi. It consists of Fritiof 7.02, Jetset7.3(by T.Sjöstrand), Pythia5.5 (by T.Sjöstrand) and Ariadone4.02(by Leif Lonnblad). See ftp://thep.lu.se/pub/LundPrograms.

dpmjet3 S. Roesler, R. Engel, and J. Ranft: The Monte Carlo Event Generator DPMJET-III, SLAC-PUB-8740, hep-ph/0012252, to be published in Proceedings of the Conference "Monte Carlo 2000", Lisbon, Portugal, 23-26 October 2000

12.1 Testing hadronic interaction models

The basic physical quantities for examining each interacion model can be obtined by running testcol.f in *Cosmos*/Particle/Event/Test. The usage is given in Readme and parm files there.

13 Miscellaneous

13.1 Utilities

In your *Cosmos*, you will find a directory, Util, which contains sources of several complete programs. Some of them may be a shell script. When compiled, they serve as utilities. Issuing the " make" command there will tell you how to compile them. For example, you can verify your primary spectrum data by using "sss" and "sampPrim". The "sss" (show spectrum shape) shell script will extract data from your primary spectrum file, and output it in the form compatible with the gnuplot format. The "sampPrim" program will do actual Monte Carlo sampling of the primary spectrum by referring to your data and output the result on the standard output.

You may find other Util and Test directories in subdirectories of the Cosmos source. These may be also sometimes useful for your job.

For further details, see the Readme in *Cosmos*/Util.

You will find the Geomview directory here. You will be able to a nicer trace display than gnuplot dose by using Geomview. The details are given in a separate manual.

13.2 Particle code list

Cosmos uses a conventional particle code that differs completely from extensive one recommended in the Particle Data book. A particle is identified by the particle code, subcode and charge. When you need to identify a particle in the user hook routines, you may use the `#include "Zcode.h"` directive and refer the code names in that file rather than code numbers.

The following list is the names that represent the particles in Cosmos. They are roughly in the order of mass. The code for a heavy nucleus such as deuteron, alpha, ... must not be used when you judge particle type. It can be used to specify the primary partilce type only. To judge a particle type of a nucleus, you may use 'kgnuc' for particle code, and if it matches, you can identify the nucleus by testing the subcode and charge; the subcode expresses the mass number (A). To speicfiy a primary, you can also avoid using the naming below but use 'iso 3 2', for example, to express ^{3}He.

particle	code name	code number	particle	code name	code number
photon	kphoton	1	electron	kelec	2
muon	kmuon	3	pion	kpion	4
kaon	kkaon	5	nucleon	knuc	6
ν_e	kneue	7	ν_μ	kneumu	8
deuteron	kdeut	9	triton	ktriton	17
He	kalfa	10	LiBeB(A∼8)	klibe	11
CNO(A∼14)	kcno	12	H(A∼25)	khvy	13
VH(A∼35)	kvhvy	14	Fe(A∼56)	kiron	15
D meson	kdmes	16	ρ	krho	24
Λ	klambda	18	Λ_c	klambdac	21
Σ	ksigma	19	Ξ	kgzai	20
ω	komega	25	ϕ	kphi	26

The sub-code is used to discriminate the particle from anti-particle, if the difference is essential such as the neutron and neutrino, but not used for, say, anti-protons because the charge can tell it. For K0 mesons, the sub-code is used to distinguish between K0-short and K0-long. Cosmos does not assign the particle and anti-particle code to them. They are assumed to be produced in equal weight and the actual assignment is performed randomly when they interact. To identify the particle and anti-particle, you can use the sub-code name, "regptcl" and "antip". For K0-short and K0-long, the sub-code name "k0s" and "k0l" may be used.

The "KF" code used in Particle Data Book can be converted to the Cosmos code by calling ckf2cos as

- call ckf2cos(kf, code, subcode, chg)

  where kf is an input integer kf code, and others are the output for Cosmos code.

The inverse conversion is possilbe by

- call ccos2kf(code, subcode, chg, kf)

The following fragment of a program code will tell you how to use these code and sub-code system.

```
...
#include "Zcode.h"
...
      record /track/ aTrack
...
      if(aTrack.p.code .eq. knuc .and. aTrack.p.charge .eq. 0) then
!              neutron; judge if anti neutron or not.
         if(aTrack.p.subcode .eq. antip) then
!                 this is anti neutron
```

```
             elseif(aTrack.p.subcode .eq. regptcl) then
!                       this is neutron
             else
!                       error assignment
             endif
...
```

13.3 List of the user hook routines

The names of the subroutines in chook.f are listed below.

- chookBgRun

  This is called when the system initialization for the current run is finished. You may do your own initialization for the run such as opening a disk file.

- chookBgEvent

  This is called when the system initialization for each event is finished. You may write some header information of the event.

- chookObs

  This is called whenever a particle crosses a predefined observation level **from above**, or reaches a height specified by BorderHeigtL or goes backward (relative to the incident direction) above BorderHeightH.

- chookEnEvent

  This is called when one event simulation is ended. You may need to write some flag to identify the end of one event.

- chookEnRun

  This is called when all the events specified by DestEventNo is completed.

- chookTrace

  This is called if you give a number greater than 60 to Trace. The default trace routine is not called in this case and you have to write your favorite trace information by using TrackBefMove and MovedTrack.

- chookEInt

  This is called whenever an electron makes an interaction. However, to avoid overhead, once the subroutine argument is set to 1 here, all later call is disabled. If you give 2 or 3, the particle is discarded. If you give 4, the current event is discarded.

- chookGInt

  This is called whenever a gamma ray makes an interaction. The argument is treated in the same way as in chookEInt.

- chookNEPInt

  This is called whenever a non electron-photon particle makes an interaction. The arguement is treated in the same way as in chookEInt, except for the case of 2 for which some special treatment of heavy particle interaction is performed. (See chookASbyH.f in UserHook and section10.15).

  Here in this subroutine, you are able to know all the information of the interacting particle and produced particles.

- chookCerenS

  This and following two are called if you give a value of > 160 to Trace. This one is called when one event generation starts.

- chookCeren

  This is called when a charged particle moves to emit Čerenkov light. You have to manage that emission yourself here.

- chookCerenE

  This is called when one event generation is finshed.

13.4 Other interface with Cosmos

To obtain certain type of information stored in Cosmos, you may call inquiry subroutines from the user hook routines.

- call cqIniRn(ir)

  where "ir" is an integer array of dimension 2 to receive the initial seed of the random number generator for each event. You may call this at any moment after the event initialization has been ended.

- call cqIncident(aTrack, angleAtObs)

  where "aTrack" is a "record /track/" and angleAtObs "record /coord/". "aTrack" will receive the incident particle track information. You will see the ingredient by looking at the chookObs routine. "angleAtObs" is to get direction cosines of the incident particle in the "detector system" of the deepest observation level. angleAtobs.x, angleAtObs.y and angleAtObs.z are the three component. You may call this at any moment after the event initialization has been ended. The program unit containing this call must have `#include "Ztrack.h"`.

- call cqNoOfPrim(no)

  where "no" is an integer to receive the current event number, i.e, the number of primaries so far sampled. You may call this after the event initialization has been ended.

- call cqPrimE(pOrE)

  where "pOrE" is a double precision real variable to receive the primary energy or momentum as specified in the primary spectrum file (MeV/n etc). Note this is not necessary the total energy of the incident.

- call cqPrimary(prm)

  where "prm" is a "record /primaries/" to receive the input primary spectrum information. See Zprimary.h in the include directory.

- call cqFirstID(depth)

  where "depth" is a double precision real variable to receive the first interaction depth of the incident particle (vertical depth). After uv6.30, the knock-on process by p, He, etc is not regarded as the first interaction point; only their nuclear interaction is picked up. For non-nucleus, all interaction types are considered. If you want to have more detailed control, you may use chookNEPInt, chookGInt and/or chookEInt.

- call cqFirstIPI(aTrack)

  where 'aTrack' is the record /track/, which is to contain all the track information of the first interaction point. The definition of the first interaction point is the same as for cqFirstID.

- call cqEventNo(num, cumnum)

  where 'num' is an integer variable to receive the current event number, and 'cumnum' for cumulative event number (which will differ from 'num' if 'Cont=t' is used.

13.5 Random number generator in the user hook

In some case, you need to use random numbers in a user hook routine. You should not use "rndc" and "rndd" which are supplied by Cosmos. If these are used, the systematic skeleton/flesh mechanism will not work. To ease the user, Cosmos supplies the third generator, of which the calling sequence is
`call rnde(ua, n)`
where ua is a double precision array with at least dimension n. You will get n uniform random number ($0 \sim 1.0$; excluding the both limit) in ua. You may, of course, prepare your own.

13.6 External names in the user hook

Cosmos uses subroutines and external symbols with name headed by "c" and "k". One exception is rnd* related to the random number generators. Subroutines in the Import directory use rather non-systematic naming. If you use your own subroutines or external symbols in the user hook, you should be careful that the name does not collide with those defined in Cosmos.

13.7 Error check of input data

Cosmos does not make a systematic check of inconsistency of the input parameters[8]. You should be careful of the parameter setting before going into an extensive simulation.

14 Distributed jobs

From version uv5.00, Cosmos includes the scripts for managing distributed jobs over a number of workstations. They are gathered in *Cosmos*/DistJob/bin. *Cosmos*/DistJob is the directory which contains the distributed-job-related stuff. 'Readme' there will tell you how to use the scripts. Cont =t, Job='skeleton' and Job='flesh' will work normally.

[8]In the old main frame version, parameters were rather extensively examined. However, it was found that such a check was not so effective in actual applications.

15 Appendix

The description of parameters

File name convention: If a file name path contains `%_` at the top of the file name or at the top of directories, % of each of them will be replaced by the host name of the computer where the program runs. For example, if Cosmos is running on a host called 'hodge1', and a file path is specified by ../%_cont/%_Contfile, the actual path will be ../hodge1_cont/hodge1_Contfile.

Frequently used ones: &Param data

ASDepthList	real*8	This is DepthList for AS observation. Used only if Generate contains "as". See DepthList.
ASHeightList	real*8	This is HeightList for Air Shower observ. Used only if Generate contains "as". See HeightList.
Azimuth	complex*16	Range of azimuthal angle in deg. Say, (0, 45). Default is (0,360). Can be such as (300., 390.). Used when Za1ry is 'is' If ObsPlane=3 (spherical), this is used to show the half opening angle range where the primary injection position is uniformly distributed on a sphere. The center of the opening angle is (Latit, Longit, HeightOfInj). In this case, for the upper opening angle, min(Imag(Azimuth),180.) is used.
BaseTime	real*8	Rough cpu time needed for completing one event (say, for protons, or gamma rays) with energy BaseErg. The cpu time estimation is based on A * (E1ry par nucleon)**BasePower / BaseErg * BaseTime, where A is mass number (for nucleus; otherwise 1).
Cont	logical	If T, continuation from a previous job is assumed. Contfile content is used.
ContFile	character*70	Job continuation information file (full or relative path). default is "ContInfo". This will be created when job is finished normally.
CosZenith	complex*16	Range of cos(zenith angle). Say, (0.5, 1.0). Used when Za2ry is 'is' If ObsPlane=3 (spherical), real(CosZenith) must be >0, and means the zenith angle range at the incident point (not in Exyz system). In that case, azimuth is 0 to 2pi.
CutOffFile	character*70	Geomagnetic cut-off file
Ddelta	real*8	SourceDec $\pm$ Ddelta is the region for 'aps' (deg).
DeadLine	character*8	The dead line before which the job should terminate. Should be given like '10.11.15' which means the nearest 10th, 11 O'clock, 15 min. Not used if Within has non zero value.
DepthList	real*8	Depth List of Observation level in kg/m^2. If <0, HeightList has priority. (See HeightList)
DestEventNo	integer	2 integers: Final event no. to be generated and events to be generated in the current run. If negative, their absolute is used and counting includes discarded ones due to rigidity cutoff. If DestEventNo(2)=0, DestEventNo(1) is used. If it is negative, only DestEventNo(2) is checked to see events in the current run. For the flux calculation, negative ones are better.
DtGMT	real*8	Difference of the local time of the observation place from GMT (hour).
Freec	logical	if F, the first interaction point is forced to be the injection point else the interaction poin is randomly sampled.

Generate	character*16	specify what should be generated 1) Electro-magnetic cascade(em), 2) one dimensional hybrid AS(as/qas) and/or 3) AS Lateral distribution(lat). If Generate= ' ', hadronic cascade shower is generated. For example, you may give as follows: Generate='em,as' or 'em/as' (order/case/separator insensitive) is to generate EM-cascade and AS. Generate='as' will generate AS with some adequate EM cascade (EM cascade is automatically generated so that hybrid A.S can be observed, but the minimum energy in EM cascade is independent of KEminObs). If 'qas' is given, quick generation of AS for heavy primaries is tried. See chookASbyH.f
HeightList	real*8	Height of observation levels in m. This is made from DepthList internally. I.e., this one is usually not an input. However, if the DepthList values are negative, this is used as input and corresponding DepthList is computed internally.
HeightOfInj	real*8	The vertical height of primary injection point (m). If this is < deepest obs. level and zeinth angle of primary is < 0, the primary is assumed to be upgoing even if Reverse =0. NOTE: BorderHeightH must be given explicitly in this case.
Hidden	logical	Make T, if hidden parameters are to be written.
IncMuonPolari	logical	if T, consider muon polarization
InitRN	integer	Initial random number seed. 2 integers. If InitRN(1) < 0, file dev # 14 is assumed to have pairs of IR in each row, and they are read to initialize each event. This feature is ignored when Job = 'flesh' or 'newflesh'. The # 14 file should be opened by the user routine (chookBgRun). This is almost debug purpose. If InitRn(2)<0, timer and host name are used for the initialization.
IntModel	character*64	Interaction model description. Usage was changed from v6.0. One may list code name and upper energy bound for the code. E.g. IntModel = '"dpmjet3"' ; to specify the dpmjet3 in the entire energy region IntModel = '"dpmjet3" 100 "nexus2" to specify dpmjet3 at < 100 GeV and nexus2 at E>100 GeV IntModel = '"nucrin" 5 "fritiof1.6" 500 "adhoc" to specify Nucrin, fritiof1.6, and ad-hoc model in the respective energy region. This corresponds to the old IntModel='int1'. IntModel = '"nucrin" 5 "fritiof1.6" 10 "fritiof7.02" and IntModel = '"dpmjet3"' are most promissing models that fit the observed data (muons and gamma rays) for which the primary is well known by BESS and AMS observations (< 100 GeV).
Job	character*10	What kind of job you are going to do. =' ' (default). nothing special. ='skeleton'. Makes skeleton. ='flesh'. Flesh skeleton events. See manual. ='newskel' ='newflesh' see manual.
KEminObs	real*8	The min kinetic energy of particles for observation.
LatitOfSite	real*8	Latitude of the deepest observation level in degree. East is positive.

LongitOfSite	real*8	Longitude of the deepest observation level in degree. North is positive.
LpmEffect	logical	If t, the LPM effect is considered when Ee > LpmBremEmin for electrons and Eg > LpmPairEmin for gamma rays.
MinPhotoProdE	real*8	Below this energy, no photo-prod of hadron. See also PhotoProd.
ObsPlane	integer	How to observe particles. 0⇒ no detector plane is used for observation. BorderHeightL and BorderHeightH are used to detect particles. This is for, say, neutrino observation. See BorderHeightL,H. However, the primary is directed to the deepest depth. 1,-1⇒ detector at the observation place is horizontal. Note that the horizontal means not tangential plane, but rather a spherical surface 2,-2⇒ detector is perpendicular to the primary. 3⇒ spherical observation. See text. For ObsPlane=1,2, the user observation routine will receive coordinate values in the corresponding detector system. However, if it is 0, 3 or negative, Exyz values are obtained.
OneDim	integer	If 0, 3 dimensional simulation. if $\geq$1, one dimensional simulation is performed. 1: onedim without use of table. 2: table is used for thickness ↔ length conversion. if cos < .5 3: table is always used for any angle. (for height > 30 km, table is not used in any case).
PhotoProd	logical	Switch. if .false., no photo prod. of hadron is considered at all. See also MinPhotoProdE.
PrimaryFile	character*70	Primary Spectrum data file (full or relative path)
SeedFile	character*70	File to contain the initial random numbers for those events to which you want to flesh. You can create the file by calling cwriteSeed in a user hook routine (say, in chookEnEvent) at skeleton making time. Default is 'Seed'. For a normal run with Job=' ', if SeedFile is not ' ', two integer initial random numbers and the event number are automatically output on the speicfied disk file.
SkeletonFile	character*70	Skeleton information file (full or relative path). created if Job = 'skeleton'. Default is 'skeletonParam'. This is the Namelist data referred by Cosmos automatically if Job='flesh' is specified. For Job='flesh', you have to modify some part of this file.
SourceDec	real*8	Source declination of point source.(deg)
ThinSampling	logical	if F, thinsampling is not tried. if T, alla Hillas thinning. Don't use with the skeleton/flesh method
TimeStructure	logical	If T, time information is computed

Trace	integer	Flag for trace information output. 0 → no trace information is output. <10→ x, y, z in the primary system(say, 1) <20 → x, y, in the primary sys. z in kg/m^2.(say,11) <30 → x, y, z in the detector system <40 → x, y, in the detector system. z in kg/m2 <50 → x, y, z in 'xyz' system. <60 → x,y, in 'xyz' and z in kg/m2 61-100 → for Cherenkov observation. For Coord system, subtract 60. if the value is even, binary output is made on TraceDev. if the last digit is 1 or 2, trace is always taken. if the last digit is 3 or 4, trace is taken only if the particle is located below the heighest observation depeth. > 101 → subtract 100 and apply the above, but chookTrace or chookCeren are used. Primary system: Origin is the deepest detector. Z-axis is the primary direction. X-axis is Z x Vertical axis. X-Y plane is orthogonal to the primary. Detector system: origin is the deepest detector. Z-axis is the vertical one. X-axis is directed to the magnetic east. X-Y plane is horizontal. z in kg/m^2 : Vertical depth in kg/m^2 above the deepest detector to the particle.
TraceDir	character*70	Directory. Default Trace information is put TraceDir/trace1, 2,.. for event 1, 2, ... The directory should exist. Default is ' ' and in this case /tmp/YourLoginName/ is employed. The last "/" should not be given. *** NOTE that default Cherenkov output is made only using TraceDev, TraceDir is not used. You have to open the disk file at chookbgRun It can by binary or ascii file depending on Trace value.
WaitRatio	real*8	Wait A.S generation until the electron energy, Ee, becomes < WaitRatio* E0. This many be 1.0 for hadron origin case. But for gamma/electron primary, this should be as low as 0.01 to enjoy full fluctuation.
Within	integer	The job should end within this minutes from now. Default is 99999. If 0 is given, DeadLine is used.
YearOfGeomag	real*8	Like 1999.5. Year when Geomagnetic field is to be calculated.
Za1ry	character*4	Specify the primary angle sampling method by one of 'is', 'ps' or 'aps'. "is" is isotropic. The range is by CosZenith. "ps" is for point source (See also SourceDec) "aps" is around point source (See also SourceDec and Ddelta) If ObsPlane=3 (spherical), this must be "is".

Not frequently used ones: &HParam data

AnihiE	real*8	If E (positron) < AnihiE, annihilation is considered.
AtmosFile	character*100	path to the atmospheric data as in 'Cosmos/Data/Atmos/stdatmos2.d'
BackAngLimit	real*8	If the cosine of the angle between a particle and the primary becomes smaller than this value, the particle is discarded. See also BorderHeighH. If you give a value less than -1.0, such rejection will never happen. Default is -1.0

BaseErg	real*8	See BaseTime. The default is 1000 (GeV).
BasePower	real*8	See BaseTime. Default is 1.0
BorderHeightH	real*8	If a particle goes higher than this, discard it. This should be larger than HeightOfInj or 0. If 0, it is adjusted to be the same as HeightOfInj. NOTE: For upgoin primary cases, you have to set this one explicitly.
BorderHeightL	real*8	If a particle reaches this hight, call observation routine. No further tracking is done. This is for neutrino observation. See ObsPlane.
Cekaon	real*8	Obsolute
Ceneuc	real*8	`p -> n ; n-> p; p~->n~; n~->p~` prob.
Cepic0	real*8	Obsolute
Charge2heavyG	integer	charge of heavy $\rightarrow$ heavy group index conversion array.
Code2heavyG	integer	particle code $\rightarrow$ heavy group index conversion array.
Code2massN	integer	particle code $\rightarrow$ mass number conversion array.
Deltkp	real*8	k-p xsection increases as E^{Deltkp} (E> 100GeV)
Deltpip	real*8	pi-p xsection increases as $E^{Deltpip}$ (E> 100GeV)
Deltpp	real*8	p-p xsection increases as E^{Deltpp}(E> 100GeV)
DpmFile	character*70	control card to specify the dpmjet execution conditions. If ' ', Cosmos/Data/DPM/atmos.inp is assumed.
Ecrit	real*8	Critical energy in GeV. Employed only when calculating air shower size in the hybrid air shower generation. The value would be dependent on the experimental purpose. The default value, 81 MeV, is bit too small in many applications (The air shower size is overestimated). Comparisons of sizes by the hybrid method and by the full Monte Carlo tells that N_e (full 3-D M.C) $<$ N_e (hybrid AS with E_c = 81 MeV) $<$ N_e (full 1-D M.C) $\lesssim N_e$(hybrid AS with E_c = 76 MeV) at around shower maximum. Hybrid AS is always essentially 1-D.
Efermi	real*8	If Kinetic E $<$ Efermi, Fermi Momentum is considered for Nucleus target.
Elund2	real*8	obsolete (from v6.0)
Elund3	real*8	obsolete (from v6.0).
Elund	real*8	obsolete (from v6.0)
EndLevel2	integer	Don't worry. This is system use.
EndLevel	integer	Used for skeleton/flesh-out job. In a normal job, system default value 0 is reset by the system to be the max number of observation levels. (=NoOfSites). Its real use is in such a skeleton/flesh-out job that one first follows the particles up to some high depth and later chooses events and flesh them out to deeper depths. In such a skeleton-making job, the user must give the depth list which is used flesh-out job, too. In the skeleton job, particle tracking is terminated at the level specified by EndLevel. In such a flesh-out job, the user must give a larger value or 0 to EndLevel
ErrorOut	integer	Error output logical dev number.
Es	real*8	Modified scattering constant. 19.3d-3 GeV
Eta2Pi0	real*8	eta/pi0 ratio. this is used to see the effect due to non-decay of pi0 at very high energies. Only source of h.e gamma can be eta and LPM may work for them. default is 0.2
EthinRatio	real*8	if ThinsSamplig = F, thin sampling is performed if the energy of a particle is $<$ EthinRatio * PrimaryEnergy(/nucleon) (=Ethin) (EtinRatio> 0). If EthinRatio $<$ 0, Ethin will be —EthinRatio— (GeV).

EventNo	integer	cumulative event number counter.(excluding discarded ones due to cutoff).
EventsInTheRun	integer	Counter for event number in the run. Internal use. (excluding discarded ones due to cutoff).
ExactThick	logical	If T, a given length is converted into thickness with best accuracy even for very inclined trajectory by using numerical integration.
FragmentTbl	real*8	tbl(i,j)=<Number> of frag. j when a heavy of heavy group index i breaks up at air.
Generate2	character*16	don't touch this. for skeleton/flesh use.
GeomagFile	character*70	IGRF or WMM file path which contains geomagnetic field expansion coefficients. Their format is the same one as given in their web page. If ' ' (default), Cosmos/Data/Geomag/igrf is used.
HeavyG2charge	integer	heavy group index → charge of heavy conversion array.
HeavyG2code	integer	heavy group index → particle code conversion array.
HeavyG2massN	integer	heavy group index → mass number conversion array.
HeavyG2symbol	character*4	heavy group index → 'Fe' etc conversion array.
HowGeomag	integer	if 1, no magnetic field until first coll. 2, mag.f always exists. If Reverse not=0, use this. 11, same as 1 but mag.f is const. 12, same as 2 but mag.f is const. 21, same as 1 but mag.f is const. 22, same as 2 but mag.f is const. 31, same as 1 but mag.f is dependent on the position. const value is the one at deepest observation plane. for 11,12 or should be given by MagN, MagE, MagD for 21, 22. For normal applications, 11 is good. If no magnetic field is applied, energy loss by dE/dx is considered.(bef.4.92, and aft. 5.14)
HowIntNuc	integer	If 0, the number of interacting nucleons among a projectile heavy nucleus is determined as the number of first collision of each interacting nucleon inside the nucleus. If 1, the number is determined as the total number of collisions including successive interactions. Default is 1. (There is uncertaninity in interpretation of the formula; value 1 gives larger number of interacting nucleons.)
InclusiveFile	character*100	The path to the inclusive data file, xdist.d. Default is "../Contrib/Inclusive/xdist.d"
IncreaseXsec	real*8	how the xsection increases. 1.0→ power of E
KEminObs2	real*8	Don't touch this. skeleton/flesh use.
KnockOnRatio	real*8	KnockOnRatio* KEminoObs is used instead of RecoilKineMinE if KnockOnRatio <1.
Knockon	logical	Obsolete. Don't use this. See RecoilKineMinE and KnockonRatio.
Kpicns	real*8	See Kpilog. 0.077
Kpilog	real*8	K_{ch}/π_{ch}=(Kpilog*log(ss+.069)+Kpicns)*exp(-8/s') where ss(GeV**2)= effective s. s'(GeV**2)=s - 4.63. See also Kpicns.
LamorDiv	real*8	In the geomagnetic field, a charged particle can travel almost streight in (Lamor Radius)/LamorDiv. Default is 5. For AMS like tracking 20 may be needed.
LpmBremEmin	real*8	The LPM effect is taken into account for bremsstrahlung when LpmEffect is .true. and the electron energy is higher than this.
LpmPairEmin	real*8	The LPM effect is taken into account for pair creation when LpmEffect is .true. and the gamma energy is higher than this.

LundPara	integer	To control Lund program. LundPara(1) is set to kfr(7); kfr(7)=1 is for Frititof hard scattering. 2 is for Pythia H.S. 2 gives higher multiplicity but shape is strange. Default is 1. LundPara(2) is set to kfr(12): 1 by for OPAL hard scattering parameterization. 2 by DELPHI. Default is 2. (2 gives bit higher PT). LundPara(3) > 0 ⇒ Pythia message will appear. LundPara(4) > 0 Fritiof message; both on ErrorOut. LundPara(5) =0 ⇒ All kaons collisions are treated as pi- in Fritiof, else they are treated by adhoc model as they are.
MagBrem	integer	If 0, no magnetic bremsstrahlung is considered. if 1 and Ee > MagBremEmin, energy loss due to magnetic brems is considered if 2 and Ee > MagBremEmin, real sampling of gamma is performed. (note, actually upsilon is referred further). if generate='as' with really high energy primaries, WaitRatio must be made small so that WaitRatio*E0 ∼ MagBremEmin
MagBremEmin	real*8	E > this, magnetic bremsstrahlung by electrons may be considered. However, if MagBrem = 0, not considered at all MagBrem = 1, total energy loss due to brems is considered. MagBrem = 2, gamma energy is sampled actually. If upsilon (Ee/m * B/Bcr) is small, the effective treatment will be the same as MagBrem = 0 case.
MagChgDist	real*8	Distance where mag. can be seen as const.(m) at sea level
MagD	real*8	See HowGeomag (in Tesla)
MagE	real*8	See HowGeomag (in Tesla)
MagN	real*8	See HowGeomag (in Tesla)
MagPair	integer	If 0, no magnetic pair creation is considered. if 1 and Eg ¿ MagPairEmin, real sampling is tried. (note, actually upsilon is referred further). To see these magnetic effects, HowGeoMag=2 and HightOfInj ∼ 5000 km are desirable.
MagPairEmin	real*8	E > this, magnetic pair creation by gamma may be considered. However, if MagPair = 0, not considered at all. MagPair = 1, pair creation is sampled. However, again, actual occurrence will be dependent on the angle between B and photon direction.
MaxComptonE	real*8	Above this energy, Compton scattering is neglected.
MaxPhotoE	real*8	Above this energy, photoelectric effect is neglected.
Moliere	integer	0→ use Gaussian approx always (with air density change and energy loss effect) 1→ use Molière scattering for non-electrons (default) 2→ use Molière scattering for all charged particles. If negative, anglular-correlated displacement is made to be 0 since Molière theory cannot give it. (if > 0, we use Gaussian approximation for correlation).
MuBr	integer	parameter similar to MuNI but for bremsstrahlung by muons.

MuNI	integer	0→ nuclear interaction of muon is completely neglected 1→ energy loss by n.i is subsumed in dE/dx of muons as a continuous energy loss. Let v= Etransfer/Emu, the loss here is Int(vc:vmax) of (Emu vdsigma/dv). (vc ∼0, vmax∼1). 2→ (Default value). similar to 1 but as the continuous loss only v < vmin=10^{-3} of fractional muon energy is subsumed (Int(vc: vmin) of (Emu vdsigma/dv)). The portion of loss by v>vmin is treated as a stocastic process. However, the product from the n.i itself is neglected 3→ the same as 2, but the n.i is explicitly included to produce a number of particles. The n.i is treated as a photo-nucleus interaction.
MuPr	integer	parameter similar to MuNI but for pair creation by muons.
Mudirp	real*8	`DD~` # enhancement factor. D is only for prompt muon.
MulLow	integer	if 1, QCD predicted multiplicity law is used in the adhoc model else UA5 parametalization is used. Default is 1. (from v5), 0.6135exp(23/18sqrt(2log(roots/0.3))) is QCD jet prediction. 7.2roots**0.254 -7 is UA5 data. The branch point is set at roots = 900 GeV. (I have adjusted 0.6135 so that 900 GeV is the b.p)
NoOfSites2	integer	No of Sites for particle observation; not to be touched; for skeleton/flesh use.
OffsetHeight	real*8	The vertical offset height from the deepest detector. The primary is directed to this height above the detector. If ObsPlane is 3 (spherical), not used.
PathLimit	real*8	If the sum of (path/beta) of a particle exceeds this, it is judged as dead. (to avoid infinite cyclotron loop). However, for normal applications, this will not be effective because of BackAnglLimit. See Reverse. TimeStructure should be T if Reverse = 0 and PathLimit is to be effective.
PrevEventNo	integer	The event number already finished. System use for Cont job. (excluding discarded ones due to cutoff).
PtAvFrag	real*8	<Pt> of heavy fragments.
PtAvNonInteNuc	real*8	<Pt> of non interacting nucleons.
RatioToE0	real*8	In the A.S generation, hadronic interactions are followed down to at least RatioToE0 * E0/nucleon energy.
RecoilKineMinE	real*8	Recoil Kinetic Min Energy above which the recoil (=knock-on process) is treated. Below this energy, the effect is included as continuous energy loss. Used only if KnockOnRatio > 1. See also KnockOnRatio.
Reverse	integer	0→ Normal tracking. 1→ incident is tracked to a direction opposite to the given one. the incident is charge-conjugated. All interactions are ignored. (Use when to make cut-off table or to see a given particle (say, observed anti proton) can go out of Earth. 2→ same as 1 but energy gain (not loss) is taken into account TimeStructure should be T if Reverse = 0. See BackAnglLimit.
SeedFileDev	integer	logical device number of SeedFile.
SucInt	integer	The number of successive interactions inside A is affected by this parameter. If 0 → old formula (before uv3.500) is used, which give rather smaller number ($< Nsuc >$ in Air = 1.7 for 30 mb pp), if 1 → new one $< Nsuc >= 2.2$ for 30 mb pp). Default is 0 (from V5.00 again).

SucPw	real*8	In the 2nd, 3rd,.. collision of a nucleon inside a nucleus, the collision is made to be more elastic than normal one. The leading particle spectrum is sampled from x**SucPw dx. SucPw should be in 1 to 2.
TempDev	integer	Logical Dev. number for temporary disk use.
TraceDev	integer	Logical dev # for TraceDir/trace1,2,....
Truncc	real*8	coeff. for truncating path.
Truncn	real*8	coeff. for truncating path.
Truncx	real*8	coeff. for truncating path.
UpsilonMin	real*8	Magnetic bremsstralhung is considered only if upsilon > UpsilonMin.
UseRungeKutta	integer	How to calculate deflection by the geomagnetic field. Let L be the distance the particle travels. 0→Don't use RungeKutta method. Use the solution assuming the constant B, which is exact if B is const. Since the particle path is made short, this is enough for normal cases where particles are inside the atmosphere.(default) In every case below, if the particle height is < 30km (= cheight in ccomPathEnd.f), the same method as 0 is used. 1→ Use the Euler method. Time needed is 20As B, use the value at L/2 point obtained by using the current direction. 2→ mixture of 1 and Runge-Kutta-Gill method. If gradient of B is large, RKG is employed. This needs ~4 times more cpu time than case of 1 when making a cutoff table. The step size of RKG is ~1/10 of the Lamore radius. 3→ The same as 2 but use the Runge-Kutta-Fehlberg method instead of RKG. Step size is automattically adjusted (~1/20 ~1/30 of Lamor radius) 4→ As a middle point, use the point obtained by assuming the constant B at initial point. If grad B is still large, use RKG. 5→ The same as 4 but us RKF instead of RKG. 6→ Use always RKG 7→ Use always RKF. This takes very long time.(50 times of 0).
UserHookc	character*100	array size is MAX_USERHOOKC(=5). Usage is left for the user. To get the i-th component, the use may 'call cqUHookc(i, cv)' in the userHook routine, where cv is a character variable to receive the data.
UserHooki	integer	array size is MAX_USERHOOKI(=10). Usage is left for the user. To get the i-th component, the use may 'call ccqUHooki(i, iv)' in the userHook rouitne, where iv is an integer varialbe to receive the data.
UserHookr	real*8	array size is MAX_USERHOOKR(=10). Usage is left for the user. To get the i-th component, the use may 'call cqUHookr(i, rv)' in the userHook routine, where rv is a real*8 variable to receive the data.
X0	real*8	Radiation length in kg/m^2 for air. Normally the user should not touch this.
XaxisFromSouth	real*8	Angle between the horizontal detector X-axis and the south(deg). + is counter clockwise. If |XaxisFromSouth| > 360, it is computed so that the direction is to the magnetic east at the deepest observation point. Default is 361.

The default values of the parameters

```
$PARAM
ASDepthList = .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0,
ASHeightList = .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0,
Azimuth = (.0,360.00),
BaseTime = 10.0,
Cont = F,
ContFile = 'ContInfo ',
CosZenith = (.600,1.00),
CutOffFile = ' ',
Ddelta = 5.00,
DeadLine = ' ',
DepthList = .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0,
DestEventNo = 1, 0,
DtGMT = 8.00,
Freec = T,
Generate = 'em ',
HeightList = .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0, .0,
HeightOfInj = 100000.0,
Hidden = F,
IncMuonPolari = T,
InitRN = 0, 0,
IntModel = 'int1 ',
Job = ' ',
KEminObs = 1000.0,
LatitOfSite = 30.1100006103516,
LongitOfSite = 90.5299987792969,
LpmEffect = T,
MinPhotoProdE = .150000005960464,
ObsPlane = 1,
OneDim = 0,
PhotoProd = F,
PrimaryFile = ' ',
SeedFile = 'Seed ',
SkeletonFile = 'SkeletonParam ',
SourceDec = 30.0,
ThinSampling = F,
TimeStructure = T,
Trace = 0,
TraceDir = ' ',
WaitRatio = 1.00,
Within = 99999,
YearOfGeomag = 2000.500,
Za1ry = 'is ',
$END


------------------ Don't worry about the following for normal applications
$HPARAM
AnihiE = 2.999999932944774E-002,
BackAngLimit = .0,
BaseErg = 1000.0,
BasePower = 1.00,
BorderHeightH = .0,
BorderHeightL = .0,
```

```
Cekaon = .0,
Ceneuc = .350,
Cepic0 = .0,
Charge2heavyG = 1, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 7, 7, 7, 7, 7, 7, 7, 7, 7,
Code2heavyG = 0, 0, 0, 0, 0, 1, 0, 0, 0, 2, 3, 4, 5, 6, 7,
Code2massN = 0, 0, 0, 0, 0, 1, 0, 0, 0, 4, 8, 14, 25, 35, 56,
Deltkp = 7.999999821186070E-002,
Deltpip = 7.999999821186070E-002,
Deltpp = 7.999999821186070E-002,
Ecrit = 8.100000023841860E-002,
Efermi = 9.999999776482582E-003,
Elund = 4.98999977111816 or 500 (non Absoft)
Elund2 = 4.98999977111816 or 500(non Absoft)
Elund3 = 4.98999977111816 or 500(non Absoft)
EndLevel = 0,
EndLevel2 = 0,
Es = 1.930000000000000E-002,
Eta2Pi0 = .200000002980232,
EthinRatio = 2.999999924213630E-005,
EventNo = 0,
EventsInTheRun = 0,
ExactThick = F,
FragmentTbl = 1.00, 4.00, 5.80999994277954, 6.61999988555908, 10.500, 12.8000001907349, 18.100000381469
Generate2 = ' ',
HeavyG2charge = 1, 2, 4, 7, 12, 17, 26,
HeavyG2code = 6, 10, 11, 12, 13, 14, 15,
HeavyG2massN = 1, 4, 8, 14, 25, 35, 56,
HeavyG2symbol = 'p ','Alfa','L ','CNO ','H ','VH ','Fe ',
HowGeomag = 11,
HowIntNuc = 1,
InclusiveFile = '../../Contrib/Inclusive/xdist.d ',
IncreaseXsec = 1.00,
KEminObs2 = .0,
Knockon = T,
Kpicns = 7.699999958276750E-002,
Kpilog = 6.200000178068880E-003,
LpmBremEmin = 100000000.00,
LpmPairEmin = 1000000000.0,
LundPara = 1, 2, 0, 0, 1, 0, 0, 0, 0, 0,
MagBrem = 2,
MagBremEmin = 300000000.00,
MagChgDist = 20000.00,
MagD = .0,
MagE = .0,
MagN = .0,
MagPair = 1,
MagPairEmin = 20000000000.00,
MaxComptonE = 7.500000298023223E-002,
MaxPhotoE = .0,
Moliere = 1,
MuBr = 2,
MuNI = 2,
MuPr = 2,
Mudirp = 1.00,
```

```
MulLow = 1,
NoOfSites2 = 0,
OffsetHeight = .0,
PathLimit = 130000000.00,
PrevEventNo = 0,
PtAvFrag = .0,
PtAvNonInteNuc = 5.000000074505810E-002,
RatioToE0 = 9.999999747378751E-006,
RecoilKineMinE = 2.500000118743630E-004,
Reverse = 0,
SeedFileDev = 22,
SucInt = 0,
SucPw = 1.50,
TempDev = 11,
TraceDev = 21,
Truncc = 5.00,
Truncn = 9.999999776482582E-003,
Truncx = 2.00,
UpsilonMin = 3.000000026077032E-003,
X0 = 364.00,
XaxisFromSouth = 361.00,
$END
```