Monte Carlo transport codes for use in the space radiation environment. Pinsky L, Ferrari A.
https://three.jsc.nasa.gov/articles/MC-THREE-FLUKA.pdf. Date posted: 02-13-2015.

Monte Carlo Transport Codes for use in the Space Radiation Environment

Alfredo Ferrari and Lawrence Pinsky

FLUKA (FLUctuating KAskade)

Available at: http://www.fluka.org/fluka.php

FLUKA is an INFN (Italy) & CERN supported code developed by the FLUKA Collaboration. The current version of FLUKA is a multi-decade evolution of a code that was originally developed by Johannes Ranft. FLUKA, like the rest of the Monte Carlo codes described here, presents the user with a relatively "turn-key" code wherein the code itself is delivered basically as a ready to run executable. Like the other codes, FLUKA requires input files that specify the parameters that control the options available (e.g. lower limits to energy propagation of particles) as well as the physical conditions such as the geometry and the initial conditions for the particles to be transported. FLUKA is one of the codes that broadly employs physics-based models for the interactions, and is continually updated and refined as new data become available. FLUKA also has the capability to model the evolution of induced radioactivity due to incident radiation on materials in the geometry employed. This is useful in predicting long-term latent dose effects from activation of nuclei in the body of the astronauts themselves, as well as from the decay of such products in the surrounding spacecraft environment.

Of relevance to space radiation simulation, FLUKA has a collection of different heavy-ion event generators that are each employed over different ranges of the energy spectrum. At the highest energy, DPMJET3 is used, followed by a heavily modified version of rQMD-2.4 at energies down to ~100 MeV/A, at which point a model based on the Boltzmann Master Equation approach is used. One of the advantages of using complete coherent event generator models is that the interactions are usually simulated in the Center of Mass frame, so that all resulting fragments as well as any newly created particles are predicted not only by particle type (or specific fragment isotope) but also in terms of the specific direction of each one of them. This allows the accurate evaluation not only of neutrons as described earlier, but also to take into account any very low energy "target-fragments" (in the "lab" frame) that may result, which is of significant interest when the interactions occur within the astronaut's body. FLUKA includes also a sophisticated model for Electro Magnetic Dissociation of ions which can be an important process for the heaviest ones found in cosmic rays. In addition, FLUKA provides a built-in capability of generating Galactic Cosmic Ray spectra for all ions from H to Ni in a given problem, to bias them, (e.g. in order to better sample the high energy tail, to normalize automatically all computed results to the incident spectra, and/or to include the effect of the earth geomagnetic field in the Stormer approximation). This is all available to the user, who needs only activate the proper code options, possibly with the mouse in the Flair graphical interface, which is described below.

FLUKA provides the ability to employ biasing in a number of ways to enable the user to reduce the computing time needed to simulate rare events with high relative statistical significance. FLUKA also provides capabilities to simulate the interactions in common detector systems that allow the user to compare its predictions directly with experimental results by simulating to actual raw data files produced by the experimental apparatus. This allows the comparison to take place at a point where the vagaries of possible systematic differences, that might occur during the analysis of the raw data, are not a factor. Also, the availability of "history-files" from the Monte Carlo outputs can be useful to the experimenter to understand the source of at least some of the "outlier" events that are typically "cut" during the final analysis of the raw data.

FLUKA is written in Fortran, And it is pretty insensitive to the choice of the compiler used. On GNU/Linux systems the code is fully portable across all major distributions and it can be compiled and run with all open-source Fortran compilers, g77, gfortran, and g95. The user can take advantage of the ability within FLUKA to write and compile into the code a separate custom user scoring subroutine. Such a capability is useful when implementing the process of producing output tailored specifically to the users' needs. However, in general the built-in scoring suffices for most problems. An extensive array of estimators are available, including point, boundary and volume estimators. Several physical quantities can be automatically scored, including fluence spectra and spatial distributions, dose and dose equivalent (via ICRP60 Q(LET) relation, or through the use of several sets of conversion coefficients) 3D distributions, residual activity maps etc.

FLUKA comes with a very powerful interface, Flair, which allows one to describe a problem, run the code, collect and analyze the results all within a user friendly graphical tool based on python and a few other widely available tools (gnuplot etc). Recently the ability to plot (including 3D plots), debug and even build interactively a geometry with the mouse has been implemented into Flair.