Proceedings of ICRC 2001: 1 © Copernicus Gesellschaft 2001

ICRC 2001

Muon Monte Carlo: a new high-precision tool for muon propagation through matter

Dmitry Chirkin[1,2] **and Wolfgang Rhode**[2]

[1]University of California at Berkeley, USA
[2]Bergische Universität Gesamthochschule Wuppertal, Germany

Abstract. Propagation of muons through large amounts of matter is a crucial necessity for analysis of data produced by muon/neutrino underground experiments. A muon may sustain hundreds of interactions before it is seen by the experiment. Since a small uncertainty, introduced hundreds of times may lead to sizable errors, requirements on the precision of the muon propagation code are very stringent. A new tool for propagating muon and tau leptons through matter that is believed to meet these requirements is presented here. The latest formulae available for the cross sections were used and the reduction of the calculational errors to a minimum was our top priority. The tool is a very versatile program written in an object-oriented language environment (Java). It supports many different optimization (parametrization) levels. The fully parametrized version is as fast or even faster than the competition. On the other hand, the slowest version of the program that does not make use of parametrizations, is fast enough for many tasks if queuing or SYMPHONY environments with large number of connected computers are used. An overview of the program is given and the results of its application to two muon detectors (AMANDA and Fréjus) are discussed.

1 Introduction

In order to observe atmospheric and cosmic neutrinos with a large underground detector (e.g. AMANDA (Andres et al., 2001)), one needs to isolate their signal from the 3-5 orders of magnitude larger signal from the background of atmospheric muons. Methods that do that have been designed and proven viable (DeYoung, 2001). In order to prove that the methods work and to derive indirect results such as the spectral index of atmospheric muons, one needs to compare data to the results of the computer simulation. Such a simulation normally contains three parts: propagation of the measured flux of the cosmic particles from the top of the atmosphere down to the surface of the ground (ice, water); propagation of the atmospheric muons from the surface down to and through the detector; generation of the Cerenkov photons emanating from the muon tracks in the vicinity of the detector and their interaction with the detector components. The first part is normally called *generator*, since it generates muon flux at the ground surface; the second is *propagator*; and the third simulates the detector interaction with the passing muons. Mainly two generators were used so far (by AMANDA): basiev and CORSIKA (Heck et al., 1998). Results and method of using CORSIKA as a generator in a neutrino detector (AMANDA) were discussed in our previous contribution (Chirkin and Rhode, 1999). Several muon propagation Monte Carlo programs were used with different degrees of success as propagators. Some are not suited for applications which require the code to propagate muons in a large energy range (e.g. mudedx), the others are optimized in only some of the interesting energy range ($E > 1$ TeV, propmu) (Desiati and Rhode, 2001). Most of the programs use cross section formulae, whose precision has been improved since their writing. In addition, one would like to use the code for the propagation of muons that contain 100 - 1000 interactions along their track, so the precision of each step should be sufficiently high and the computational errors should accumulate as slowly as possible. Here we present a new tool (Muon Monte Carlo: MMC), specifically designed to meet these criteria.

Correspondence to: D. Chirkin (chirkin@physics.berkeley.edu)

2 Description of the code

The primary design goals of MMC were uncompromising computational precision and code clarity. It was decided that the program should be written in JAVA, since JAVA is an object-oriented programming language (for best code readability) and has consistent behavior across many platforms. MMC consists of pieces of code (classes), each contained in a separate file. These pieces fulfill their separate tasks and are combined in a structured way (Fig. 1). This simplifies

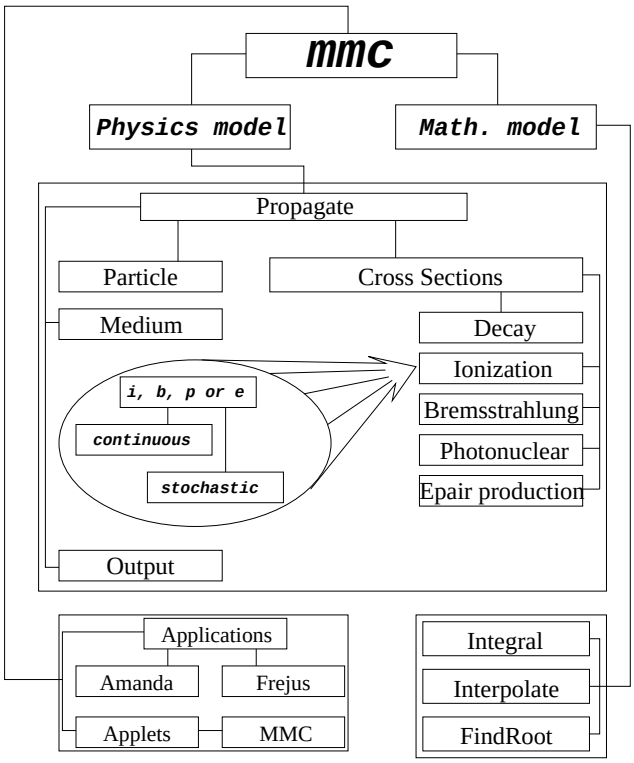

Fig. 1. MMC structure

code maintenance and introduction of changes/corrections to the cross section formulae. It is also very straightforward to even "plug in" new cross sections, if necessary.

The code evaluates many cross-section integrals, as well as two tracking integrals. All integral evaluations are done by the Romberg method of the 5th order (by default) (Numerical Recipes, 1986) with a variable substitution (mostly log-exp). If an upper limit of an integral is an unknown (that depends on a random number), an approximation to that limit is found during normalization integral evaluation, and then refined by Newton-Raphson method combined with bisection (Numerical Recipes, 1986).

Originally, the program was designed to be used in the Massively Parallel Network Computing (SYMPHONY) (Winterer, 1999) framework, therefore computational speed was considered only a secondary issue. However, parametrization and interpolation routines were implemented for all integrals. These are both polynomial and rational function interpolation routines spanned over varying number of points (5 by default) (Numerical Recipes, 1986). Inverse interpolation is implemented for root finding (i.e. when $x(f)$ is interpolated to solve $f(x) = y$). Two dimensional interpolations are implemented as two consecutive one-dimensional ones. It is possible to turn parametrizations on or off for each integral separately at program initialization. With full optimization (parametrizations) this code is at least as fast or even faster than the competition.

Generally, as a muon travels through matter, it loses energy due to ionization losses, bremsstrahlung, photo-nuclear

Fig. 2. breaking a large step into small

interaction and pair production. Formulae for the cross sections were taken from the recent contribution (Rhode and Cârloganu, 1999). These formulae are claimed to be valid to within about 1%. All of the energy losses have continuous and stochastic components, the division between which is superficial and is chosen in the program by selecting an energy cut ($ecut$) or a relative energy loss cut ($vcut$). Ideally, all losses should be treated stochastically. However, that would bring the number of separate energy loss events to infinity, since the probability of such events to occur diverges as $1/E_{lost}$ for the bremsstrahlung losses, as the lost energy approaches zero, and even faster than that for the other losses. A good choice of $vcut$ should lie in the range (0.05 - 0.1) (Bugaev et al., 2000).

Let the continuous part of the energy losses (energy losses, integrated from zero to $ecut$) be described by a function f(E):

$$-\frac{dE}{dx} = f(E).$$

The stochastic part of the losses is described by a function $\sigma(E)$, which is a probability for any energy loss event (with lost energy $> ecut$) to occur along a path of 1 cm. Consider the particle path from one interaction to the next consisting of small intervals (Fig. 2). On each of these small intervals probability of interaction is $dP(E(x_i)) = \sigma(E(x_i))dx$. It is now easy to derive an expression for the final energy on this step as a function of the random number ξ. Probability to completely avoid stochastic processes on an interval $(x_i;x_f)$ and then suffer a catastrophic loss on dx at x_f is

$$(1 - dP(E(x_i))) \cdot ... \cdot (1 - dP(E(x_f))) \cdot dP(E(x_f))$$

$$= \exp(-dP(E(x_i))) \cdot ... \cdot \exp(-dP(E(x_f))) \cdot dP(E(x_f))$$

$$= \exp\left(-\int_{E_i}^{E_f} dP(E(x))\right) \cdot dP(E(x_f))$$

$$= d_f\left(-\exp(-\int_{E_i}^{E_f} \frac{\sigma(E)}{-f(E)} \cdot dE)\right) = d(-\xi), \quad \xi \in (0;1]$$

To find the final energy on each step the above equation is solved for E_f:

$$\int_{E_i}^{E_f} \frac{\sigma(E)}{-f(E)} \cdot dE = -\log(\xi) \quad \text{(energy integral)},$$

and then the corresponding displacement is found:

$$x_f = x_i - \int_{E_i}^{E_f} \frac{dE}{f(E)} \quad \text{(tracking integral)}.$$

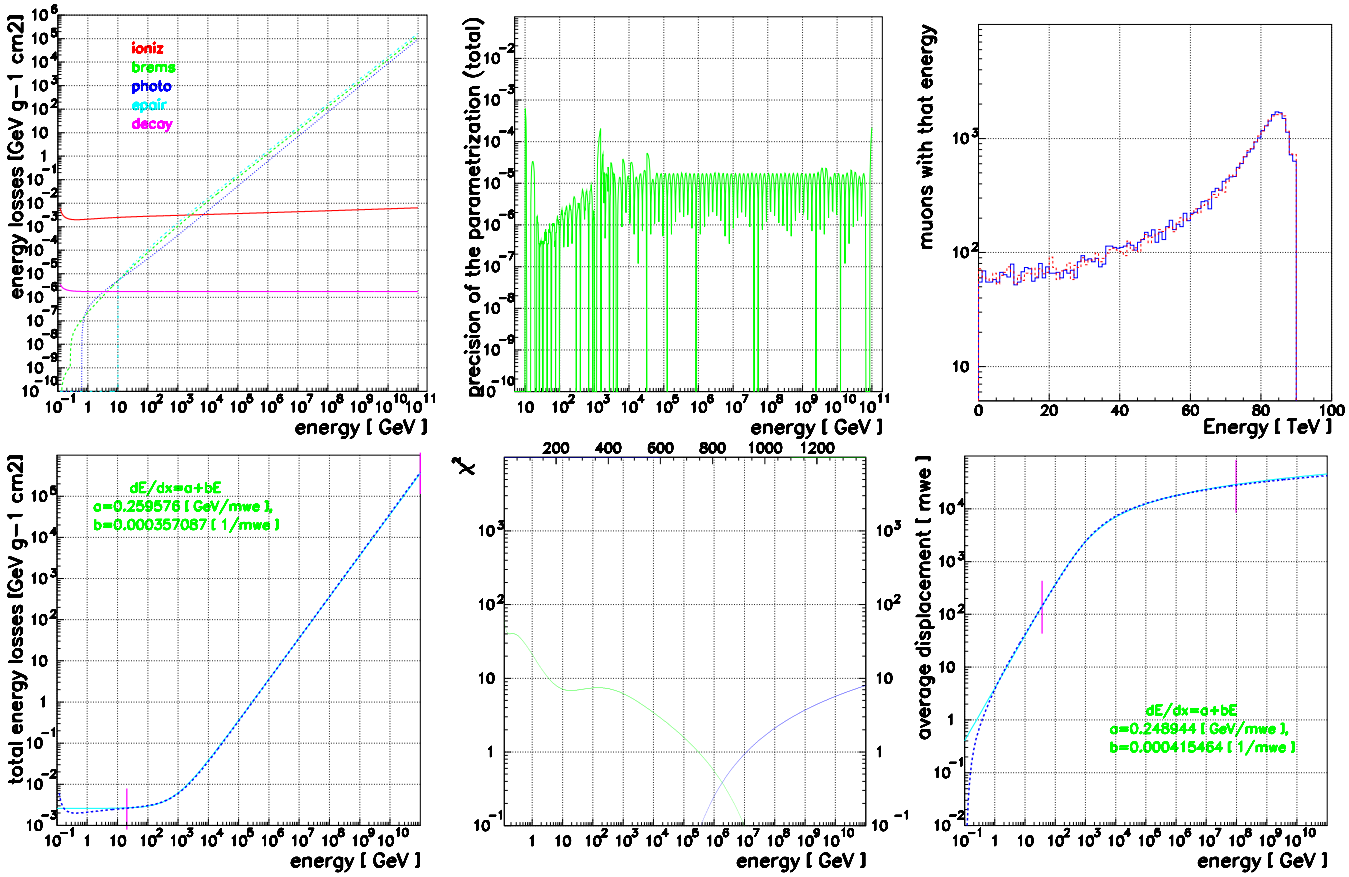

Fig. 3. from left to right, top to bottom: 1. ioniz (upper solid curve), brems (dashed), photo (dotted), epair (dashed-dotted) and decay (lower solid curve) losses; 2. Interpolation precision $(e_{pa} - e_{np})/e_{pa}$; 3. Distribution of the fi nal energy of the muons that crossed 300 m of Fréjus Rock with initial energy 100 TeV; 4. Fit to the energy losses in ice and 5. χ^2 plot; 6. Fit to the stochastic energy losses in ice

3 Errors

All cross-section integrals are evaluated to the relative precision of 10^{-6}, the tracking integrals are functions of these, so their precision was set to a smaller value of 10^{-4}. To check the precision of interpolation routines, results of running with parametrizations enabled were compared to those with parametrizations disabled. The first plot in Fig. 3 shows relative energy losses due to different mechanisms. Decay energy loss is shown here only for comparison and is evaluated by multiplying the probability of decay by the energy of the particle. In the region below 10 GeV pair production cross section is set to zero. Since on the four parametrization grid points located right below 10 GeV interpolation routines give non-zero values, whereas the exact values are zero, relative error of parametrized vs. non-parametrized runs ($(e_{pa} - e_{np})/e_{pa}$) is equal to 1. But since the value of pair production energy loss in this region is small in comparison to the sum (mostly ionization energy loss), this big relative error results in a much smaller increase of the relative error of the total energy losses (the second plot). Because of that, parametrization errors never exceed 10^{-4} - 10^{-3}, as one can estimate from the plot. These errors are much smaller than the uncertainties in the formulae for the cross sections. Moreover, from the discussion above it is clear that it is due these uncertainties (and artificial cutoffs) that the errors of parametrization may get as high as 10^{-3}. Now the question arises whether this precision is sufficient to propagate muons with hundreds of interactions along their way. The third plot is one of the examples that demonstrate that it is sufficient: the final energy distribution did not change after enabling parametrizations.

4 Results

The code was incorporated into the Monte Carlo chains of at least two different detectors: Fréjus (Schröder et al., 2001) and AMANDA (Desiati and Rhode, 2001). In this section some general results are presented.

The energy losses plot was fitted to the function $dE/dx = a + bE$ (Fig. 3, fourth plot). In order to choose low and high energy limits correctly (to cover the maximum possible range of energies that could be comfortably fitted with a line), a χ^2 plot was generated and analysed (Fig. 3, fifth plot). It can be seen that χ^2 plot at the low energies goes down sharply, then levels out. This corresponds to the point where linear approximation starts to work. At high energies

χ^2 rises monotonically. This means that a linear approximation, though valid, has to describe a growing energy range. An interval of energies from 20 GeV to 10^{11} GeV is chosen for the fit. The following table summarizes the found fits to a and b:

medium	a, $\frac{GeV}{mwe}$	b, $\frac{10^{-3}}{mwe}$	av. dev.	max. dev.
ice	0.25958	0.35709	3.7%	6.6%
fr. rock	0.23131	0.42927	3.0%	5.1%

The errors in the evaluation of a and b are in the last digit of the given number. However, if the lower energy boundary of the fitted region is raised and/or the upper energy boundary is lowered, each by an order of magnitude, a and b change by about 1%.

To investigate the effect of stochastic processes, muons with energies 120 MeV - 10^{11} GeV were propagated to the point of their disappearance. Average final distance (range) for each energy was fitted to the solution of the energy loss equation $dE/dx = a + bE$:

$$x_f = \log(1 + E_i \cdot b/a)/b$$

(Fig. 3, sixth plot). The same analysis of the χ^2 plot as above was done in this case. A region of initial energies from 40 GeV to 10^8 GeV was chosen for the fit. The following table summarizes the results of these fits:

medium	a, $\frac{GeV}{mwe}$	b, $\frac{10^{-3}}{mwe}$	av. dev.
ice	0.2489	0.4155	2.4%
fréjus rock	0.2209	0.5051	2.4%

As the energy of the muon increases, it suffers more interactions before it is lost and the range distribution becomes more Gaussian-like (Fig. 4). It is obvious that the inclusion of stochastic processes into consideration leads in general to larger energy losses than with only continuous processes and the center of gravity of the muon beam travels to a smaller distance.

5 Conclusions

A very versatile, clear-coded and easy-to-use Muon propagation Monte Carlo program (MMC) is presented. It is capable of propagating muon and tau leptons of energies from 120 MeV (just above the muon rest mass, higher for taus) to 10^{11} GeV, which should be sufficient for the use as propagator in the simulations of the modern neutrino detectors. A very straightforward error control model is implemented, which results in computational errors being much smaller than uncertainties in the formulae used for evaluation of cross sections. It should be very easy to "plug in" cross sections, modify them, or test their performance. The program was extended on many occasions to include newly introduced formulae or effects. Even though it does not change the direction of the propagated particle (as of the moment of writing), MMC is "3d-ready", since it does all calculations and checks

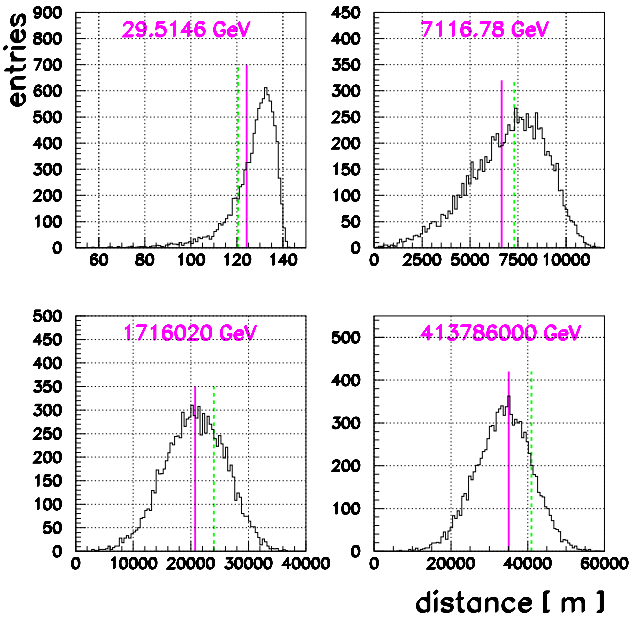

Fig. 4. Final distance distributions: on all plots solid line designates the value of the final energy evaluated with the first table (continuous losses only) and the broken line shows the final energy evaluated with the second table (continuous and stochastic losses).

in three dimensions. The correct angular dependence of the cross sections will be inserted at a later date, when necessary.

The MMC program has already been successfully incorporated into and used in the Monte Carlo chains of AMANDA and Fréjus. We hope that the combination of precision, code clarity, speed and stability will make this program a useful tool in the research connected with high energy particles propagating through matter.

Also, a calculation of coefficients in the energy loss formula $dE/dx = a + bE$ is presented for both continuous and full (continuous and stochastic) energy loss treatments. The calculated coefficients apply in the energy range from 20 GeV to 10^{11} GeV with an average deviation from the linear formula of 3.7% and maximum of 6.6%.

References

Andres, E. et al. (AMANDA Collaboration), Nature (2001) 441
Bugaev, E., Sokalski, I., Klimushin, S., hep-ph/0010323, 2000
Chirkin, D., Rhode, W., 26th ICRC, HE.3.1.07 Salt Lake City, 1999
Desiati, P., Rhode, W., these proceedings, HE.2.2
DeYoung, Tyce R., Dissertation, U. of Wisconsin-Madison, 2001
Heck, D. et al., Report FKZA 6019, 1998
Numerical Recipes (W. H. Press et al.)
Rhode, W., Cârloganu, C., DESY-PROC-1999-01, 1999
Schr¨oder, F., Rhode, W., Meyer, H., these proceedings, HE.2.2
Winterer, V.-H., SYMPHONY (talk)