PROCEEDINGS OF SCIENCE

# A new Monte Carlo muon generator for cosmic-ray muon applications

**Nicola Zurlo,**[a,b,*] **Germano Bonomi,**[b,c] **Antonietta Donzella,**[b,c] **Davide Pagano,**[b,c] **Aldo Zenoni**[b,c] **and Gianni Zumerle**[d,e]

[a] *Department of Civil, Environmental, Architectural Engineering and Mathematics, University of Brescia, via Branze 43, 25123 Brescia, Italy*

[b] *INFN Sezione di Pavia, via Bassi 6, 27100 Pavia, Italy*

[c] *Department of Mechanical and Industrial Engineering, University of Brescia, via Branze 38, 25123 Brescia, Italy*

[d] *Department of Physics and Astronomy, University of Padova, via Marzolo 8, 35131 Padova Italy*

[e] *INFN Sezione di Padova, via Marzolo 8, 35131 Padova, Italy*

*E-mail:* nicola.zurlo@unibs.it, zurlo@cern.ch

[*] Speaker





PoS(CompTools2021)019

Cosmic rays, thanks to their ubiquity and high penetration capability, have been successfully used in scientific research ever since their discovery. As soon as their knowledge improved, applications in the civil/environmental field were also developed: muon radiography (or muography, based on the flux attenuation) and muon tomography (based on the scattering angle) have been used to study the inner structure of volcanoes, to seek hidden rooms in Egyptian pyramids, to search for heavy metals in containers, and so on. And besides these imaging techniques, cosmic ray muons are also widely used for detector testing and alignment practically in every Nuclear Physics or Particle Physics experiment.

Since most of these applications are sensitive to the angular and momentum distribution of cosmic muons, an accurate modelling of these distributions is a key feature for any generation tool conceived to simulate the cosmic muon flux. This can make the generator quite time-consuming, which is a strong limit when one needs to reach high statistics or to study large structures.

A new Monte Carlo generator for cosmic-ray muons, named Efficient COsmic MUon Generator (EcoMug for short), especially designed to be fast ($\gtrsim 10^5$ muons generated per second on a standard machine) without losing accuracy, is presented here. It is written as a header-only C++11 library, ready to be integrated into whatever C++ code, in particular C++ code based on Geant4 simulation tool. By default, EcoMug relies on a simple and effective parametrisation of the experimental data of cosmic ray differential flux at sea level, taken from the literature, but the library is written in such a way that every user can easily replace it with his own user-defined parametrisation.

Unlike other tools, EcoMug is able to generate muons from different kind of surfaces (plane, cylinder and half-sphere), while keeping the correct angular and momentum distribution of generated tracks inside a fiducial volume. This allows to optimise the generation surface according to the system under study, and leads to a further improvement of the overall simulation efficiency.

In this contribution we will present the main features of EcoMug, starting from its mathematical foundation, and eventually showing some interesting applications.

*Computational Tools for High Energy Physics and Cosmology (CompTools2021)*
*22-26 November 2021*
*Institut de Physique des 2 Infinis (IP2I), Lyon, France*

## 1. Introduction

Cosmic rays are high energy particles, mostly protons and helium nuclei, that create muons (usually called *secondary* cosmic rays) on collisions at the top of the atmosphere. In fact, the so-called *primary* cosmic rays, when approaching the Earth, interact with atomic nuclei of the air molecules, giving birth to cascade processes and extended showers. These result in a number of particles among which the main components, able to reach the Earth's surface because of their relativistic velocity and consequent time dilatation, are muons.

As a rule of thumb, at the sea level the flux is so that $\sim$ 1 muon /(minute·cm$^2$) hits the ground. This flux is roughly independent of the azimuthal angle $\phi$, but strongly depends on the zenith angle $\theta$: it scales approximately as $(\cos\theta)^2$, so it is maximum at the zenith and minimum near the horizontal direction. In first approximation, it only slightly depends either on the time of the day, or on the season of the year. The kinetic energy distribution is continuous, and the energy distribution actually depends on the $\theta$ angle; the average kinetic energy is typically around some GeV (see e.g. the experimental spectra in Figure 1, taken from [1]).

Finally, an excess of positively charged muons over negative muons is observed: this is generally considered a consequence of the primary cosmic rays being positively charged. The ratio $N_{\mu_+}/N_{\mu_-}$ has been measured to be around 1.28 (see e.g. [2]).

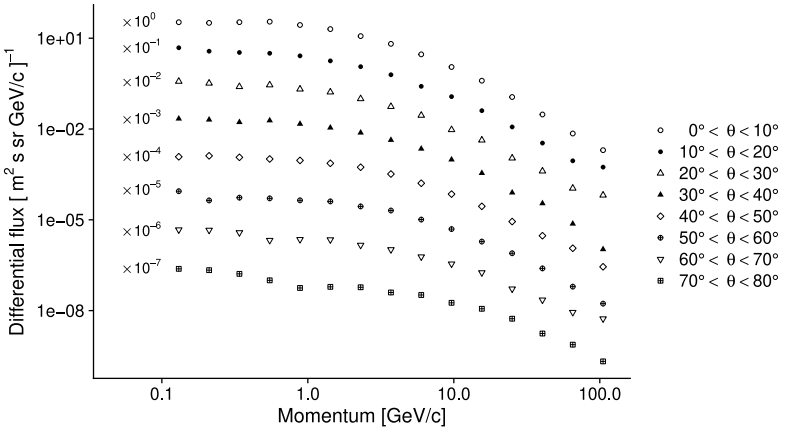


**Figure 1:** Experimental differential flux, as a function of the momentum and for eight zenith angle intervals [1]. The knee at low momenta and large $\theta$ is due to the electronic component of the secondary cosmic rays.

## 2. Cosmic ray muon applications

Originally discovered in 1936 at Caltech by Anderson and Neddermeyer [3] (although there is no agreement on this being the "real" discovery), the cosmic ray muons have been exploited in the last decades to develop applications in various fields.

In particular, some of these applications are based on the muon absorption/transmission (which in turn depends on the energy loss inside the matter), and are essentially implemented by counting the flux upstream and downstream the structure under analysis (since the upstream flux is the reference

cosmic ray flux, only the downstream measurement is usually performed); these applications are generally called muon radiography, muon absorption radiography, muon imaging, or even muography. Among them, we would like to mention the pioneering work of E.P. George, that in 1955 estimated the thickness of rock above an underground tunnel through muon flux attenuation [4], as well as the one of Luis W. Alvarez, that in 1970 tried to investigate the interior of the Khafre's pyramid at Giza [5], without good results unfortunately. By the way, further improvements in this field recently brought to the discovery of a big void in Khufu's pyramid [6], and studies on the pyramids with this technique are currently in full bloom, as well as in other fields (volcanoes [7], nuclear plants [8], etc.: see [9] for a general review on this topic).

More recent applications rely also on the deflection of the muons trajectory in their path inside the matter, due to multiple Coulomb scattering: the scattering angles have a Gaussian distribution with variance proportional to the traversed thickness and depending on the material crossed by the muons; these applications are referred to as muon tomography or muon scattering tomography, and are more complicated since they need detectors at both sides of the object under study. The first idea was proposed in Los Alamos Laboratory in 2003 [10] and a more detailed description of this application may be found e.g. in [11] (see also [12], where the same technique was put forward to monitor historical buildings).

Another field of application concerns the use of the cosmic ray muons to align structures (in particular, Particle Physics tracking detectors [13]) or even to calibrate the response of Particle Physics detectors (some of the co-authors of the present paper successfully performed the calibration of detectors built with photomultipliers and bent scintillator slabs, in the framework of the AEgiS collaboration at the CERN AD [14, 15]),

For a recent and comprehensive review of all these applications, see [16].

## 3. Available simulation tools compared with our new tool

For all these applications, it is essential to make a detailed comparison between the experimental data and the expected results obtained via Monte Carlo simulations. In fact, muon radiography applications are sensitive to the angular distribution of cosmic ray muons, while muon tomography are also often sensitive to their momentum distribution. For these reasons, an accurate simulation of the dependence of the muon flux on momentum and direction is a key requirement for every generation tool intended for such applications.

Quite a few simulation tools are currently available for the generation of the cosmic-ray muons as well as for their tracking. While, in most cases, the tracking toolkit is either Geant4 [17] or FLUKA [18], for the generation of the cosmic-ray muons there is a variety of tools already at hand.

Some research groups use their own code, based on specific parametrisations and approximations of the flux of cosmic ray muons, while others rely on dedicated packages. Among them, we can perform the following categorization:

1. cosmic-ray air shower (CRAS) generators, simulating the full cascade of secondary particles started by primary cosmic rays; examples are CORSIKA [19], CRY [20] and MCEq [21];

2. parametric generators, using a parametrisation of the flux of muons, based either on experimental data or on the results from CRAS simulations; examples are GEMC [22],

CMSCGEN [23] and the generator presented in [24], which is indeed very interesting since it shares with our new tool the emphasis on the *sampling method* itself;

3. special generators, specifically designed for underground, high altitude or underwater experiments; examples are muTeV [25] and MUPAGE [26].

In this context, we decided to develop and share a new simulation tool, called EcoMug, aimed at the generation of the cosmic-ray muons, focusing on the following features:

1. *efficiency*, being able to generate the muons on different kind of surfaces but keeping the same properties for the muons inside a fiducial volume, so that the user can choose the more suitable source geometry for his own application;
2. *speed*, being able to generate $\sim 10^6$ muons per second on a standard core, so that the time spent in generating the muons is really negligible compared to the subsequent tracking;
3. *flexibility*, being able to generate a muon differential flux with some most general parametrisation (see section 4) that can be quickly and easily replaced by other customised kind of parametrisation;
4. *portability* on different platform, since it is written as a header-only C++11 library.

In addition, EcoMug is easy to use, since the user has to implement just a few instructions to set up his customised generator, smoothly interfaced with Geant4, the widely diffused tracking tool mentioned before, and doesn't need the installation of special libraries or other tools.

## 4. Data Parametrisation

We started from the cosmic ray muon spectrum measured in [1], that defined a parametrisation approximating their experimental data (see Figure 1). According to their work, the differential flux of muons at the Earth's surface may be written as:

$$J \equiv J(t, p, \theta, \phi) = \frac{dN}{dt \cdot dp \cdot d\Omega \cdot dS_n} = f(p) \cdot (\cos\theta)^{n(p)} \cdot \frac{1}{\mathrm{m}^2 \cdot \mathrm{s} \cdot \mathrm{sr} \cdot \mathrm{GeV/c}} \tag{1}$$

where $f(p)$ and $n(p)$ are functions of $p$ so that:

$$f(p) = \left[1600 \cdot \left(\frac{p}{p_0} + 2.68\right)^{-3.175} \cdot \left(\frac{p}{p_0}\right)^{0.279}\right] \tag{2}$$

$$n(p) = \max\left[0.1,\ 2.856 - 0.655 \cdot \ln\left(\frac{p}{p_0}\right)\right]$$

with $p_0 = 1$ GeV/c and $p > 0.040$ GeV/c.

In other words, $dN$ is the number of particles, with momentum between $p$ and $p + dp$, crossing a surface element $dS_n$ orthogonal to the motion direction $(\theta, \phi)$, within the solid angle $d\Omega$ and in the time $dt$. $\theta$ and $\phi$ are respectively the zenith and azimuthal angles of the muon direction, so the solid angle is $d\Omega = \sin\theta d\theta d\phi$. Alternatively to $J$, also $J' = J \cdot \sin\theta$ may be defined.

Simulated muon spectra for different values of the zenith angle $\theta$ are plotted in Figure 2. It is worthwhile noting that for $\frac{p}{p_0} \gtrsim 67$ the value for $n$ is constrained by 0.1. As a consequence, at large momenta the momentum and the azimuthal angle are uncoupled, i.e. the most energetic muons are substantially isotropic.

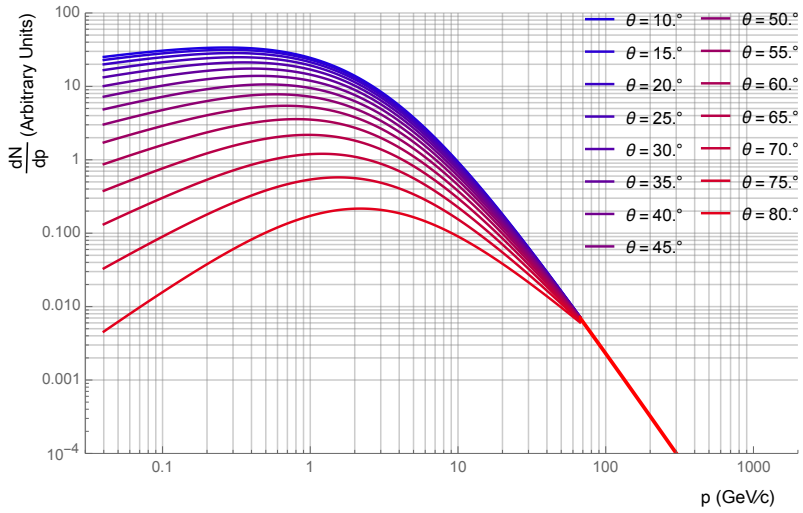


**Figure 2:** Differential muon flux across a surface perpendicular to the motion direction implemented in the generator. The bluish curves correspond to muons closer to the vertical direction, while the reddish curves indicate muons closer to the horizontal direction; for $p \gtrsim 67$ GeV/c all the spectra collapse.

## 5. Generation of muons on different surfaces

Established that the differential flux *(defined across a surface perpendicular to the motion direction)* is tied up to Equation 1, we have found that there are several ways to achieve this by the generation of muons on different surfaces (see also Figure 3, upper panel): a horizontal plane (actually, it is a square or a rectangle of finite size), a cylinder (or, to be more precise, a cylindrical surface, since the bases are not included), and a hemisphere.

The muon generation process consists in randomly generating each muon via its origin on the desired surface, its direction and its momentum, with appropriate joint probability distribution, so that the all muons inside the fiducial volume have the expected properties (uniform flux and convenient differential flux across whatever surface, i.e. correct direction/momentum distributions).

### 5.1 Generation over a horizontal plane

The easiest muon source to implement is obviously the horizontal plane (the so-called "flat sky"): only a multiplicative factor $\cos\theta$ is needed to pass from the differential flux $J'$ defined across a surface perpendicular to the motion direction (Equation 1) to the differential flux *defined through a horizontal surface*, let's call it $J'_H$.

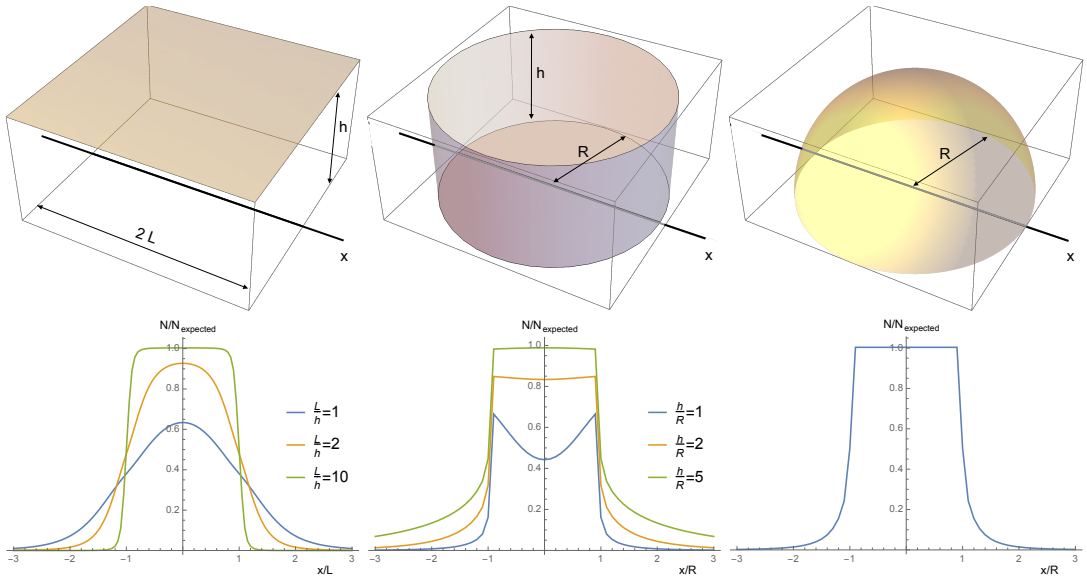


**Figure 3:** Upper panel: drawing of the three different surfaces where muons are generated. Left: horizontal plane, supposed a square, $L$ is half the side, $h$ the distance from the ground. Center: cylinder, $R$ is the base radius and $h$ the height. Right: hemisphere, $R$ is the radius. Lower panel: respective muon flux measured "on the ground", i.e. the overall number of muons measured across a horizontal surface element centred in points with coordinates ($x$,0,0), highlighted in the upper figures with the black straight line. In order to have a meaningful and universal plot, the muon flux is divided by the "expected" muon flux (i.e., the flux in case of no losses), while $x$ is plotted as either $x/L$ or $x/R$ depending on the geometry. For the first column (planar surface) the flux is plotted for planes with 3 different aspect ratios $L/h$. For the second column (cylindrical surface) the flux is plotted for cylinders with 3 different aspect ratios $h/R$.

In consequence, the number of particles with momentum between $p$ and $p + dp$, zenith angle between $\theta$ and $\theta + d\theta$, and azimuth angle between $\phi$ and $\phi + d\phi$, crossing a horizontal surface element $dS_H = dx_0 dy_0$ (subscript "0" refers to the origin point) in the time $dt$ is:

$$J'_H \equiv \frac{dN}{dt \cdot dp \cdot d\theta \cdot d\phi \cdot dS_H} = f(p) \cdot (\cos\theta)^{n(p)+1} \cdot \sin\theta \cdot \frac{1}{\mathrm{m}^2 \cdot \mathrm{s} \cdot \mathrm{sr} \cdot \mathrm{GeV/c}}$$

In this case, the generation must be uniform on the plane and isotropic in $\phi$ (for symmetry). So, the position of the muon origin, i.e. $(x_0, y_0)$, as well the angle $\phi$, are uniformly distributed without any coupling with either the angle $\theta$ or the momentum $p$, which are indeed coupled up through $n(p)$, and that must be consequently generated together .

It is clear that in principle, if we were able to generate muons on whichever horizontal and unlimited plane with a flux having these properties, the same flux would be conserved on whatever plane below the generation plane. The only practical issue is that the generation plane should be infinite size in order to generate all the muons that travel in every direction inside a fiducial volume below it. Since the simulation plane has (necessarily) finite size, there will be some edge effects, and muons coming from the almost-horizontal directions will be missing: see Figure 3, lower panel, left. But this drawback can be overcome, and the simulation results will not be affected, if there are some more constraints due to the detector geometry (like for muon telescopes, see e.g. [27]).

### 5.2 Generation over a cylindrical surface with vertical axis

Alternatively, we can generate the muons on a cylindrical surface. In this case, the differential flux $J'_V$ across a vertical surface $dS_V$ becomes:

$$J'_V \equiv \frac{dN}{dt \cdot dp \cdot d\theta \cdot d\phi \cdot dS_V} = f(p) \cdot (\cos\theta)^{n(p)} (\sin\theta)^2 \cos\phi_R \cdot \frac{1}{\mathrm{m}^2 \cdot \mathrm{s} \cdot \mathrm{sr} \cdot \mathrm{GeV/c}}$$

where now $\phi_R$ is assumed to be the azimuthal muon angle measured with respect to the local (outer) radial direction, $\phi_R = \phi - \phi_0$, being $\phi_0$ the azimuthal angle of the point where the muon originates on the cylinder ($z_0$ will be the other cylindrical coordinate and $R$ the radius, so that the vertical surface element is $dS_V = Rd\phi_0 dz_0$). In order to improve the efficiency by a factor of 2, only muons directed inside the cylinder are generated, so that $\frac{\pi}{2} < \phi_R < \frac{3\pi}{2}$.

Now, again for symmetry, the density of sources on the cylindrical surface must be uniform: $\phi_0$, $z_0$ are simply uniformly distributed (and are both independent of the others variables). On the other side, $\phi_R$ is not uniformly distributed but still independent, while the angle $\theta$ and the momentum $p$ are coupled through $n(p)$ as usual.

As for the planar surface, also for the cylindrical surface some muons are missing in the fiducial volume (obviously taken inside the cylinder). In this case, we are missing some muons closer to the vertical direction, which are not generated. Again, this is not a problem when, for other reasons, these muons are not detected by the system under study or we are not interested in. For example, in muon tomography applications, where two vertical detectors are placed around the structure to be investigated.

If one wants to get most of the muons and to minimise the edge effect for this second method, it is advisable to use a cylinder of larger height (see Figure 3, lower panel, center), more or less as in the planar case a larger plane size was recommended.

### 5.3 Generation over a hemisphere

The last case consists in generating the muons on a hemisphere or half-sphere (that is actually a hemispherical surface). If $dS_S$ is a generic surface element on a half-sphere, orthogonal to the direction $(\phi_0, \theta_0)$, $dS_S = R^2 \sin\theta_0 d\phi_0 d\theta_0$ and the differential flux $J'_S$ across this surface becomes:

$$\begin{aligned} J'_S &\equiv \frac{dN}{dt \cdot dp \cdot d\theta \cdot d\phi \cdot dS_S} = \\ &= f(p) \cdot (\cos\theta)^{n(p)} \sin\theta \cdot \Big( \sin\theta_0 \sin\theta \cos\phi_R + \cos\theta_0 \cos\theta \Big) \cdot \frac{1}{\mathrm{m}^2 \cdot \mathrm{s} \cdot \mathrm{sr} \cdot \mathrm{GeV/c}} \end{aligned}$$

As for the cylinder, $\phi_R$ is assumed to be the azimuthal muon angle measured with respect to the local radial direction, i.e. $\phi_R = \phi - \phi_0$. In order to maximize the efficiency, only muons going inside the hemisphere can be generated, although the gain here is less relevant than for the cylinder.

Generating a muon in this case means again to assign the values of five parameters: now they are $(\phi_0, \theta_0)$, defining the position of the source, $(\phi_R, \theta)$, defining the muon direction, and $p$, its momentum. These variables are currently all coupled except $\phi_0$ (which is uniformly distributed). As a consequence, both the density of muon sources on the hemisphere and the inward total flux per unit area (measured perpendicularly to the elementary surface, i.e. in the radial direction) are

not uniform but vary with $\theta_0$: it is maximum at the zenith ($\theta_0 = 0$) and minimum near the horizon $\theta_0 = \pi/2$ (see Figure 4).

The main advantage of this third method is that the fiducial volume can be extended to the whole volume inside the hemisphere, since no muon directions are lost (see Figure 3, lower panel, right), so it can be used also when we have not an a priori knowledge about muon selection by the detector. On the other hand, the coupling between four variables makes it computationally more time consuming (see next paragraph), although it gains a factor around 1.72 over the flat sky generator (for the same area and number of generated muons). This gain is because the "equivalent" flux measured perpendicularly to the hemisphere should be measured at the zenith and it is $\sim 72\%$ larger than the average flux on the whole hemisphere.

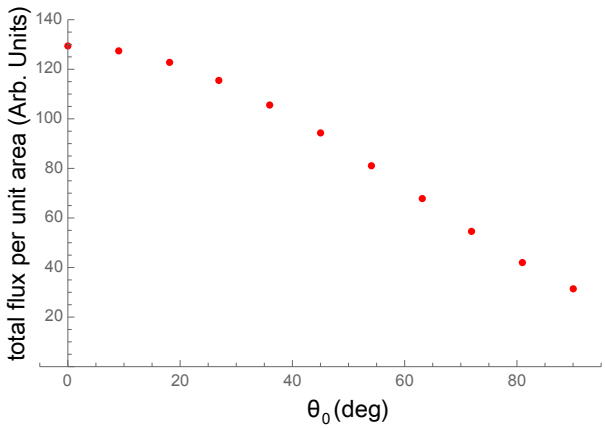


**Figure 4:** Total inward flux per unit area, integrated over the full range of $\phi$ and $\theta$ accessible for each value of $\theta_0$, plotted as function of $\theta_0$ for the hemispherical generator.

## 6. Comparison of the different generators

The three different generators produce comparable results, assumed that the fiducial volume is suitably selected (as shown in Figure 3).

To be more precise, the flat sky generator by construction loses almost-horizontal muons, which is actually not a big issue since they are indeed a small fraction of the total. In particular, the results are totally unaffected when the detector under study intrinsically selects muons in a limited range of $\theta$, like for muon telescopes, built with stacked scintillators or other sensitive elements (see e.g. [27]). In any case, choosing a large aspect ratio $L/h$ reduces the edge effects and improves the flux. This makes it extremely efficient for studying low-rise buildings or structures covering a large surface.

On the other hand, the cylindrical generator intrinsically loses almost-vertical muons, especially near it axis. Although the vertical direction is the most frequent for the muons, the generator can be successfully used when the detector disregards vertical muons and automatically selects muons with large $\theta$. Again, to reduce the losses it is advisable to increase the aspect ratio $h/R$, and this is ideally the case when tall and narrow structures are under analysis (see e.g. [28]).

As a third option, the half-spherical generator has no muon directions lost, although it is the slowest in terms of CPU time. Apart from this drawback, it is the most appropriate when muons from every directions count (like in detector calibration [14]) or when the structures have comparable size along the 3 axes.

Concerning the CPU time, i.e. the time needed to generate the muons (in terms of initial position position, direction and momentum - excluding obviously the subsequent tracking), the planar generator is the fastest, generating over 1 million muons per second on a standard CPU (see Table 1). The cylindrical generator is intermediate and the half-spherical generator is the slowest.

These times look strongly related to the number of attempts needed to find one "acceptable" set of parameters out of the generator section based on the acceptance-rejection method. The application we have made of the acceptance-rejection method will be described in the next paragraph.

| generator surface | plane | cylinder | hemisphere |
|---|---|---|---|
| calculation time per $10^6$ muons on a 2,2 GHz Intel Core i7 (2015) | 0.7 s | 2.0 s | 3.2 s |
| number of attempts (±5% ) for the acceptance-rejection module | 21 | 47 | 65 |

**Table 1**

### 6.1 Acceptance-rejection method

As we have already remarked, the generation of a single muon consists in the generation of 5 random variables ($x_0$,$y_0$, $\theta$, $\phi$, $p$ for the flat sky; $\phi_0$, $z_0$, $\theta$, $\phi$, $p$ for the cylinder; $\theta_0$,$\phi_0$, $\theta$, $\phi$, $p$ for the hemisphere) with a convenient joint probability distribution. Since some of the concerned variables are uncoupled (namely independent), it is quite easy to apply the inverse transform sampling method to them. Some of them are even uniformly distributed, and this is the easiest and fastest case. But unfortunately this method cannot be applied to all the variables, and some of them must be generated with the acceptance-rejection method, that is to say starting from a uniform distribution on a multidimensional rectangular area (see e.g. [29]).

This is essentially the reason why the CPU time increases moving from the planar case (with 2 not-independent variables, we have to sample in a 2D space and reject outside of the acceptance region) to the hemispherical case (with 4 not-independent variables, we have to sample in a 4D space and reject consistently).

The acceptance-rejection method may become very time-consuming indeed, but in our case the number of attempts to get one simulated muon is kept to an acceptable level with the following approach.

Let's start with the momentum marginal distribution (2): it can be shown that $p$ is the most critical variable to be handled with the acceptance-rejection method since $f(p)$ spans over several order of magnitudes and has small average, so a lot of attempts would be needed to find one good event; moreover, $p$ conditions $n(p)$ and then other variables' marginal distribution, like $\theta$.

Although we cannot apply the inverse transform sampling method to $f(p)$ as a whole, since we cannot invert its primitive, it is possible to split $f(p)$ as the product of 2 functions:

$$f(p) = f_1(p) \cdot f_2(p)$$

$$\text{with} \quad f_1(p) = 1600 \cdot \left(\frac{p}{p_0} + 2.68\right)^{-3.175} \quad \text{and} \quad f_2(p) = \left(\frac{p}{p_0}\right)^{0.279}$$

Since the most steep function is $f_1(p)$, while $f_2(p)$ is nearly constant, we generate a first version of $p$ by applying the inverse transform sampling method to $f_1(p)$, which is feasible. Then, we apply the acceptance-rejection method in order to take into account $f_2(p)$ and the other not-independent variables.

## 7. C++11 implementation

The code is available at https://github.com/dr4kan/EcoMug under the GPL-3.0 license, and it is possible to find an exhaustive and more detailed description in [30].

EcoMug has been designed as a header-only C++11 library: in other words, the full definitions of classes and functions are contained in a single header file. The full code is less than 1000 lines. The integration of EcoMug is thus extremely easy in any C++11 project. You just need to include the header file into the source code and point the compiler to the location of the EcoMug.h file.

If you want to use the planar generator, you have to include in the main code something like the following lines:

```
EcoMug gen; // initialization of the class
gen.SetUseSky(); // flat sky  generation
gen.SetSkySize({{10., 10.}}); // x and y size of the plane
// (x,y,z) position of the center of the plane
gen.SetSkyCenterPosition({{0., 0., 20.}});
```

In order to use the cylindrical generator, the main code shall contain:

```
EcoMug gen; // initialization of the class
gen.SetUseCylinder(); // cylindrical surface generation
gen.SetCylinderRadius(10.); // cylinder radius
gen.SetCylinderHeight(30.); // cylinder height
// (x,y,z) position of the center of the cylinder
gen.SetCylinderCenterPosition({{0., 0., 15.}});
```

Finally, the use of the hemispherical generator needs in the main code these settings:

```
EcoMug gen; // initialization of the class
gen.SetUseHSphere(); // half-spherical surface generation
gen.SetHSphereRadius(30.); // half-sphere radius
// (x,y,z) position of the center of the half-sphere
gen.SetHSphereCenterPosition({{0., 0., 0.}});
```

Once the setup of the instance of the *EcoMug* class is done, the generation of a cosmic-ray muon can be implemented as shown here:

```
// Setup of the instance of the EcoMug class e.g as in the flat sky case:
....
// The array storing muon generation position
std::array<double, 3> muon_position;
// Loop to generate 1000 cosmic-ray muons
for (auto event = 0; event < 1000; ++event) {
        gen.Generate(); // generate a cosmic-ray muons
        muon_position = gen.GetGenerationPosition();
        double muon_p = gen.GetGenerationMomentum();
        double muon_theta = gen.GetGenerationTheta();
        double muon_phi = gen.GetGenerationPhi();
        double muon_charge = gen.GetCharge();
        ...
}
```

EcoMug internally uses a class called EMRandom, which is based on the xoroshiro128+ algorithm [31], the fastest PRNG at the present time.

A more detailed description on how to integrate EcoMug in a Geant4 application is presented in appendix A of [30] as well as in [32]. Moreover, EcoMug provides the possibility to use a custom function for $J$, in case the users consider that the default parametrisation (1) does not provide an accurate description of the cosmic-ray muons differential flux in the conditions of their experiments. How to do so is explained in detail in the mentioned reference as well.

## 8. Conclusions

We have developed and presented a new, fast and efficient Monte Carlo generator of secondary cosmic-ray muons, called EcoMug.

EcoMug gives the possibility of originating the cosmic-ray muons from different surfaces (plane, cylinder and half-sphere) keeping the correct angular and momentum distribution of the generated tracks inside the fiducial volume. The generation is based on a real data parametrisation of the ADAMO detector data [1], that can be also easily overwritten by any user that needs a custom-defined parametrisation. If necessary, it is also possible to restrict the angular variables and the muon momentum at the generation level.

EcoMug is a header-only C++11 library, consisting in a single EcoMug.h file containing the whole optimised code, suitable for applications that require huge statistics, as transmission muography and tomography.

EcoMug is available at https://github.com/dr4kan/EcoMug under the GPL-3.0 license. The current version (April 2022) is EcoMug 1.3.

## References

[1] L. Bonechi et al., *Development of the ADAMO detector: test with cosmic rays at different zenith angles*, 29th International Cosmic Ray Conference, Vol. 9, pp. 283-286 (2005).

[2] V. Khachatryan et al. (CMS Collaboration), *Measurement of the charge ratio of atmospheric muons with the CMS detector* Physics Letters B, Vol. 692, Issue 2, pp. 83-104 (2010). doi: 10.1016/j.physletb.2010.07.033

[3] S.H. Neddermeyer and C.D. Anderson, *Note on the Nature of Cosmic-Ray Particles*, Physical Review, Vol. 51 Issue 10, pp. 884-886 (1937). doi:10.1103/PhysRev.51.884

[4] E. George, *Cosmic rays measure overburden of tunnel*, Commonwealth Engineer, pp. 455-457 (1955).

[5] L.W. Alvarez et al., *Search for Hidden Chambers in the Pyramids*, Science, Vol. 167, Issue 3919, pp. 832-839 (1970).

[6] K. Morishima et al., *Discovery of a big void in Khufu's Pyramid by observation of cosmic-ray muons*, Nature, Vol. 552, pp. 386-400 (2017).

[7] A. Anastasio et al., *The MU-RAY detector for muon radiography of volcanoes*, Nucl. Instrum. Methods Phys. Res. A, Vol. 732, pp. 423-426 (2013).

[8] N. Kume et al., *Muon trackers for imaging a nuclear reactor*, JINST, Vol. 11, P09008 (2016).

[9] S. Procureur, *Muon imaging: Principles, technologies and applications*, Nucl. Instrum. Methods Phys. Res. A, Vol. 878, pp.169-179 (2018).

[10] K.R. Borozdin et al., *Radiographic imaging with cosmic-ray muons*, Nature, Vol 422, pp. 277-279 (2003).

[11] C.L. Morris et al., *Tomographic Imaging with Cosmic Ray Muons*, Science and Global Security, Vol 16, pp. 37-53 (2008). doi: 10.1080/08929880802335758

[12] G. Bonomi et al., *Cosmic ray tracking to monitor the stability of historical buildings: a feasibility study*, Meas. Sci. Technol., Vol. 30, 045901 (2019).

[13] The CMS collaboration, *Alignment of the CMS tracker with LHC and cosmic ray data*, JINST, Vol. 9, P06009 (2014).

[14] N. Zurlo et al. (AEgIS Collaboration), *Calibration and equalisation of plastic scintillator detectors for antiproton annihilation identification over positron/positronium background*, Acta Phys. Polon. B, Vol. 51 (1), pp. 213-223 (2020).

[15] C. Amsler et al. (AEgIS Collaboration), *Pulsed production of antihydrogen*, Commun Phys., Vol. 4, Article 19 (2021). https://doi.org/10.1038/s42005-020-00494-z

[16] G. Bonomi et al., *Applications of cosmic-ray muons*, Progress in Particle and Nuclear Physics, Vol. 112, 103768 (2020).

[17] S. Agostinelli et al., *Geant4 simulation toolkit*, Nucl. Instrum. Methods Phys. Res. A, Vol. 506 (3), pp. 250-303 (2003).

[18] T. Böhlen, F. Cerutti et al., *The FLUKA code: Developments and challenges for high energy and medical applications*, Nucl. Data Sheets, Vol. 120, pp. 211-214 (2014).

[19] D. Heck, T. Pierog and J. Knapp, *CORSIKA: An air shower simulation program*, Astrophysics Source Code Library, record ascl:1202.006 (2012). URL https://www.iap.kit.edu/corsika/

[20] C. Hagmann, D. Lange and D. Wright, *Cosmic-ray shower generator (CRY) for Monte Carlo transport codes*, 2007 IEEE Nuclear Science Symposium Conference Record, IEEE (2007).

[21] A. Fedynitch, R. Engel, T.K. Gaisser, F. Riehn and T. Stanev, *Calculation of conventional and prompt lepton fluxes at very high energy*, EPJ Web of Conferences, Vol. 99, EDP Sciences, 08001 (2015).

[22] GEMC: GEant4 Monte-Carlo, URL https://gemc.jlab.org/gemc/html/index.html

[23] P. Biallass and T. Hebbeker, *Parametrization of the cosmic muon flux for the generator CMSCGEN*, arXiv preprint arXiv:0907.5514 (2009).

[24] S. Chatzidakis, S. Chrysikopoulou and L.H. Tsoukalas, *Developing a cosmic ray muon sampling capability for muon tomography and monitoring applications*, Nucl. Instrum. Methods Phys. Res. A, Vol. 804, pp. 33-42 (2015).

[25] G. Battistoni, A. Margiotta, S. Muraro and M. Sioli, *FLUKA as a new high energy cosmic ray generator*, Nucl. Instrum. Methods Phys. Res. A, Vol. 626, S191-S192 (2011). doi: 10.1016/j.nima.2010.05.019

[26] G. Carminati, M. Bazzotti, A. Margiotta and M. Spurio, *Atmospheric muons from parametric formulas: A fast generator for neutrino telescopes (MUPAGE)*, Comput. Phys. Comm., Vol.179 (12), pp. 915-923 (2008).

[27] M. Abbrescia et al. (EEE Collaboration), *The cosmic muon and detector simulation framework of the extreme energy events (EEE) experiment*, Eur. Phys. J. C (2021) Vol. 81, Article 464, pp. 1-15 (2021). doi: 10.1140/epjc/s10052-021-09237-y

[28] X. Hu et al., *Exploring the Capability of Muon Scattering Tomography for Imaging the Components in the Blast Furnace*, ISIJ International, Vol. 58 (1), pp. 35-42 (2018). doi:10.2355/isijinternational.ISIJINT-2017-384

[29] B.D. Flury, *Acceptance-Rejection Sampling Made Easy*, SIAM Review, Vol. 32, No. 3, pp. 474-476 (1990).

[30] D. Pagano, G. Bonomi, A. Donzella, A. Zenoni, G. Zumerle and N. Zurlo, *EcoMug: An Efficient COsmic MUon Generator for cosmic-ray muon applications*, Nucl. Instrum. Methods Phys. Res. A, Vol. 1014, 165732 (2021).

[31] D. Blackman and S. Vigna, *Scrambled Linear Pseudorandom Number Generators*, ACM Trans. Math. Softw., Vol. 47 (4), Article 36, pp. 1-32 (2021). doi: 10.1145/3460772

[32] G. Bonomi et al., *A Monte Carlo Muon Generator for Cosmic-Ray Muon Applications*, Journal for Advanced Instrumentation in Science, Vol. 2022, No. 1, (2022).