

Propagation of simulated millicharged particles to the Milliquan detector

C. Campagnari, B. Marsh (UCSB), F. Golf (UNL)

September 10, 2019

1 Introduction

In this note we describe the procedure used to propagate simulated millicharged particles (mCPs) through the CMS environment to the Milliquan detector face. The goals of this are to (1) calculate the acceptance of the Milliquan detector (and thus, when combined with $\sigma \times \mathcal{B}$, the rate of incidence of mCPs), and (2) generate “hits” on the Milliquan detector face (position and momentum of incoming mCPs) that can be fed into more sophisticated Geant4 simulations of the detector.

There are four main components to this simulation, listed here and then described in more detail in the following sections:

1. Propagation through the CMS magnetic field
2. Multiple scattering while traveling through matter
3. Energy loss while propagating through matter
4. Computation of intersection with Milliquan detector face

All of the code for the propagation is contained in the [MilliquanSim](#) GitHub repository.

2 Propagation through the CMS magnetic field

At its core the simulation uses a fourth-order Runge-Kutta integrator to step a charged particle through a magnetic field. The basic equations of motion are

$$\frac{d\vec{x}}{dt} = \vec{v} = \frac{\vec{p}c}{E} = \frac{\vec{p}c^2}{\sqrt{(\vec{p}c)^2 + (mc^2)^2}}, \quad (1)$$

$$\frac{d(\vec{p}c)}{dt} = qc \vec{v} \times \vec{B}. \quad (2)$$

If B is in Tesla, t in ns, velocity in m/ns, charge in units of e , and $\vec{p}c$ in MeV, the latter becomes

$$\frac{d(\vec{p}c)}{dt} = 89.8755 q \vec{v} \times \vec{B}. \quad (3)$$

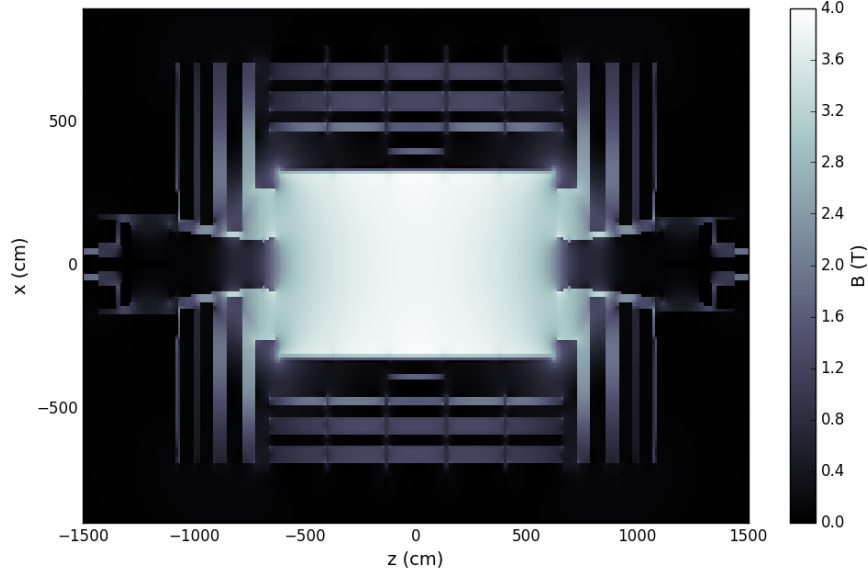


Figure 1: Magnitude of the CMS magnetic field, in the $y = 0$ plane.

The magnetic field for CMS has been extracted from CMSSW in a 3-dimensional cylindrical grid in increments of $\Delta r = 10$ cm, $\Delta z = 10$ cm, $\Delta\theta = 5^\circ$, out to $r = 10$ m and $z = \pm 15$ m. A map of the magnitude in the $y = 0$ plane is shown in Fig. 1. *[Note: we extracted this B-field map from CMSSW, which isn't public. But there are CMS publications (e.g. [1]) that measure this, so I guess we can plausibly say we got it from those.]*

At each timestep, the magnetic field is determined from this pre-loaded map and used in the equation of motion (3). The field is interpolated between the two nearest points in r (not z or θ , since for the region of interest the field is essentially flat over these variables).

For now, the simulation uses a timestep of $dt = 0.1$ ns, corresponding to a distance $dx \approx 3$ cm for a particle traveling near the speed of light. This can probably be increased without meaningfully changing the results, in order to speed up computation time.

3 Material model of CMS and surroundings

In order to simulate the interactions of particles with matter, a model of CMS has been extracted from a public ROOT model [2]. At each point in space, it is possible to get the material as well as its properties. A plot of $\rho Z/A$ as a function of radius is shown in Fig. 2. This is used to construct a simplified model of CMS for the purposes of particle propagation. For simplicity, all material lengths are converted into “equivalent distance of iron” by scaling by the $\rho Z/A$ ratio. Then concentric cylinders of iron are placed between the following radii:

- $1.80 \leq r < 2.80$ m (tracker + HCAL)
- $3.15 \leq r < 3.50$ m (magnet)
- $3.85 \leq r < 4.05$ m (return yoke 1)
- $4.60 \leq r < 4.95$ m (return yoke 2)

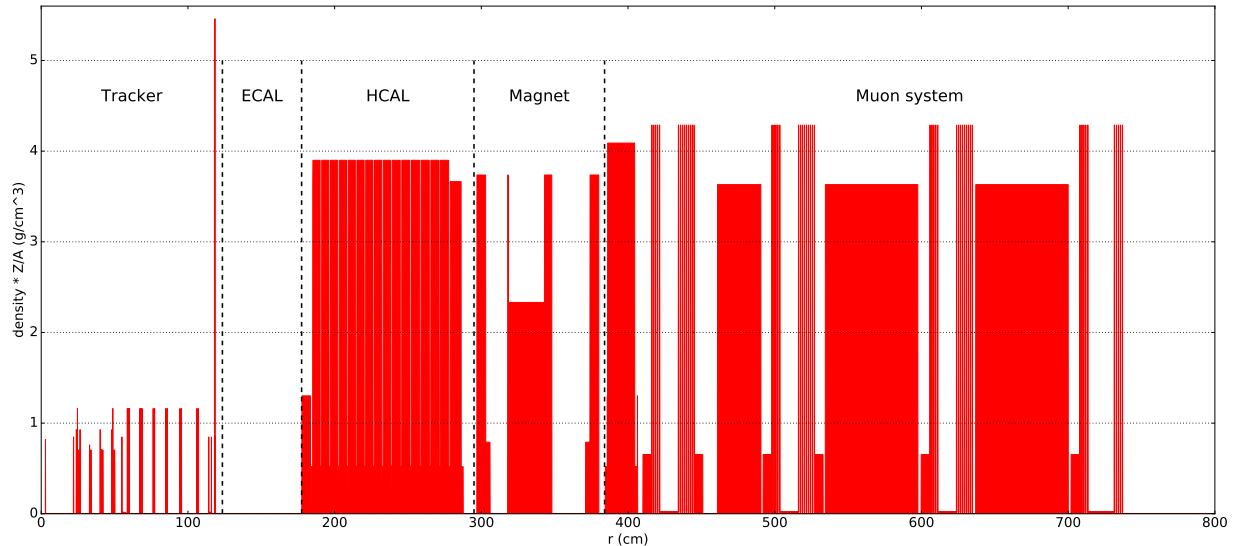


Figure 2: A map of material $\rho Z/A$ as a function of radius for CMS, as extracted from a public ROOT model of CMS [2]. Note that ECAL is missing, so we add 25 cm of PbWO_4 between radii of 1.25 and 1.50 m.

- $5.35 \leq r < 5.95$ m (return yoke 3)
- $6.35 \leq r < 7.15$ m (return yoke 4)

The total amount of iron is 3.30 m. In addition, 25 cm of PbWO_4 is placed between $1.25 < r < 1.50$ m to account for the ECAL crystals.

Rock is placed between $15.8 < R < 32.8$ m to account for the 17 meters of rock between the IP and Milliquan. All space not filled by iron, PbWO_4 , or rock by the above rules is set to air.

4 Multiple scattering through matter

Two methods of implementing multiple scattering have been studied. The first assumes purely small-angle gaussian scattering, and is described in the PDG chapter on the passage of particles through matter [3]. The second is a bit more complicated but more realistically simulates the larger, non-gaussian tails present in true multiple scattering. It is outlined in a paper by Kuhn and Dodge [5]. The prediction using the Kuhn algorithm agrees better with measured muon rates at Milliquan (the predicted slab rate is 10% lower compared to the prediction using the PDG algorithm), but uncertainties are relatively large so it is difficult to say definitively that it is more accurate. The following sections briefly summarize the implementations of both of these algorithms.

4.1 “PDG algorithm” for multiple scattering

The simpler implementation of multiple scattering is taken from the PDG chapter on the passage of particles through matter [3]. This assumes a small-angle gaussian scattering, and does not take into account non-gaussian tails from rare hard scatters. The algorithm consists of picking two random scattering angles independently in orthogonal directions, as well as correlated transverse displacements.

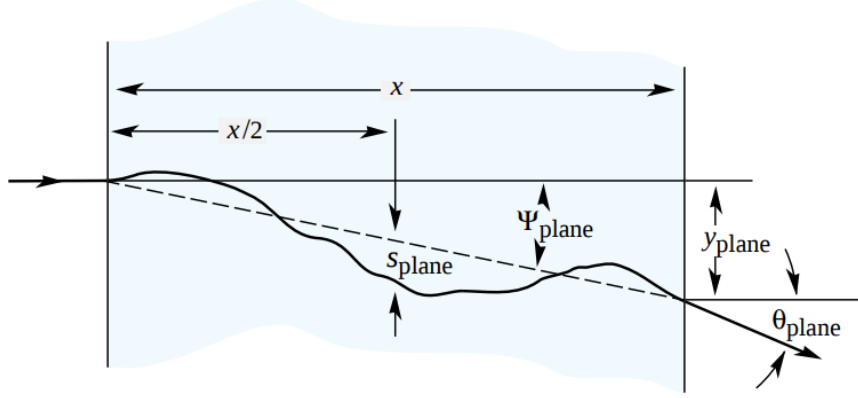


Figure 3: Diagram showing the defined θ_{plane} and y_{plane} variables used to describe the PDG algorithm for multiple scattering.

The RMS of the scattering angle distribution is given by

$$\theta_0 = \frac{13.6 \text{ MeV}}{\beta c p} z \sqrt{\frac{x}{X_0}} \left[1 + 0.088 \log_{10} \left(\frac{x z^2}{X_0 \beta^2} \right) \right], \quad (4)$$

where p , βc , and z are the momentum, velocity, and charge number of the incident particle, x is the distance traversed, and X_0 is the radiation length of the material.

In addition to an angular deflection θ_{plane} , there is a correlated transverse deviation y_{plane} (see Fig. 3). The correlation coefficient turns out to be $\sqrt{3}/2 \approx 0.87$, and it is sufficient to generate two independent gaussian random variables z_1, z_2 and compute

$$y_{\text{plane}} = z_1 x \theta_0 / \sqrt{12} + z_2 x \theta_0 / 2 \quad (5)$$

$$\theta_{\text{plane}} = z_2 \theta_0.$$

This is done independently in both directions orthogonal to the direction of travel.

4.2 “Kuhn algorithm” for multiple scattering

The second algorithm more accurately models the larger, non-gaussian tails of the scattering angle distribution at each timestep. It essentially piecewise-joins a gaussian core with non-gaussian tails modeled by the Rutherford scattering law. Details of the derivation are given in [5]. Here we describe the technical implementation.

First, constants are defined:

$$\chi_c^2 = 0.1569 \frac{z^2 Z(Z+1) \rho \Delta s}{(pc)^2 \beta^2 A} \quad (6)$$

and

$$B_0 - \log(B_0) = \log \left(\frac{6700 z^2 Z^{1/3} (Z+1) \rho \Delta s / A}{\beta^2 + (1.77 \times 10^{-4}) z^2 Z (Z+1)} \right), \quad (7)$$

where z is the charge of the projectile (in units of e), Z and A are the effective atomic number and atomic weight of the target material, respectively, $\beta = v/c$ is the velocity of the projectile, pc is its momentum (in MeV), and ρ is the density of the target material in g/cm^3 . These are taken from

the original analytical formulation of multiple scattering by Molière. The replacement of Z^2 by $Z(Z+1)$ and $Z^{4/3}$ by $Z^{1/3}(Z+1)$ are due to the correction for highly relativistic particles derived by Bethe. The constant B_0 is defined as a part of a transcendental equation, so it is solved for numerically using the Newton-Raphson method. For the following, we use a modified $B = B_0 + 1$ rather than B_0 straight from Eq. 7. This was found by Kuhn to fix an empirically too-narrow final distribution.

The core of the scattering angle distribution is described by a gaussian of width

$$\theta_{1/e} = \chi_c \sqrt{B - 1.25}. \quad (8)$$

Using this, we can compute

$$P(\pi) = 1 - 0.827 \frac{1}{B} + \frac{\chi_c^2}{4} \left(\frac{1}{\sin^2(\theta_{1/e}/\sqrt{2})} - 1 \right). \quad (9)$$

Then the actual algorithm is as follows:

1. Generate a random number R in $[0, P(\pi)]$.

2. If $R \geq 1 - 0.827/B$, then

$$R' = R - 1 + 0.827/B$$

$$\theta = 2 \sin^{-1} \left(\frac{\chi_c \sin(\theta_{1/e}/\sqrt{2})}{\sqrt{\chi_c^2 - 4R' \sin^2(\theta_{1/e}/\sqrt{2})}} \right). \quad (10)$$

3. Otherwise, if $R < 1 - 0.827/B$, then

$$\theta = \theta_{1/e} \left[\log \left(\frac{1 - 0.827/B}{1 - 0.827/B - R} \right) \right]^{1/2}. \quad (11)$$

4. Generate a random ϕ in $[0, 2\pi]$. Then the particle's new direction of propagation is given by (θ, ϕ) in spherical coordinates centered on its original direction of propagation.

The Kuhn algorithm does not provide for any transverse displacement on top of the angular deflection at each timestep. However, the paper shows that the final spatial distribution of a particle is well described by this algorithm using only angular deflections at each timestep.

4.3 Comparison of algorithms

Figure 4 shows a comparison of projected angular distributions for muons incident on the Millikan detector face, using either the PDG algorithm or the Kuhn algorithm. The width of the central gaussian core does not change much (1.07° vs. 1.14°). However, there is a small but noticable shift in the Kuhn algorithm to a smaller core and larger tails, as might be expected. The overall fraction of events in the fitted gaussian core goes from 74% to 70%, but the decrease in the central-most region (less than $\sim 1^\circ$) most relevant for through-going particles is closer to 10%. This corresponds well with an observed decrease in predicted four-slab muon rate of 10% when switching to the Kuhn algorithm.

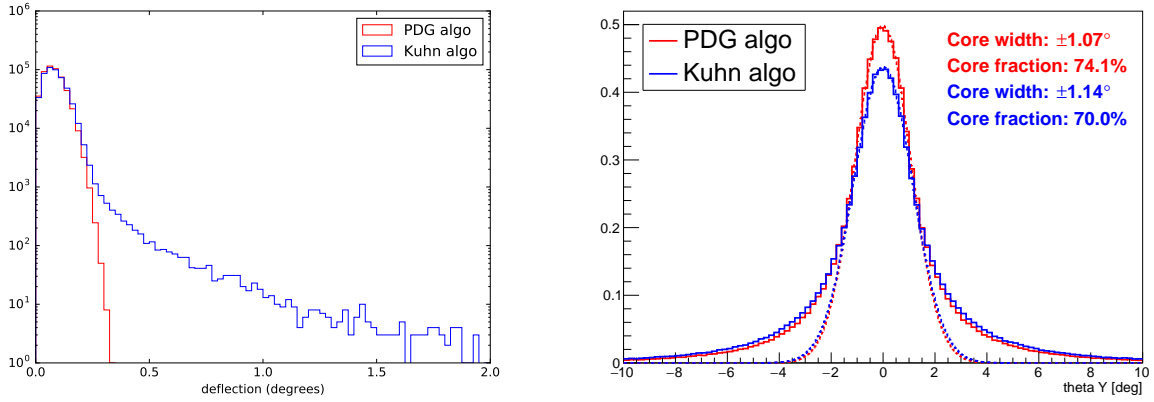


Figure 4: (left) The scattering angle distribution of a 16 GeV muon through 3 cm of iron, as simulated with the PDG algorithm (blue) or the Kuhn algorithm (red). Note the much larger high-angle tail when using the Kuhn algorithm. (right) The distribution of incidence angles (projected into the horizontal plane) of muons on the Millikan detector face for each algorithm. Dotted lines are gaussian fits to the cores of the distributions. The Kuhn algorithm slightly enlarges the tails of the distribution, and reduces the probability near $\theta = 0^\circ$ by $\sim 10\%$.

5 Energy loss

In addition to multiple scattering, particles lose energy as they propagate through matter. This is implemented using the Bethe formula

$$\left\langle -\frac{dE}{dx} \right\rangle = K z^2 \frac{Z}{A} \frac{1}{\beta^2} \left[\frac{1}{2} \log \frac{2m_e c^2 \beta^2 \gamma^2 W_{\max}}{I^2} - \beta^2 - \frac{\delta(\beta\gamma)}{2} \right]. \quad (12)$$

For definitions of all terms and parameters, see [3]. All material constants are taken from [4]. Note that energy loss is a stochastic process, and at each timestep the actual energy loss follows a highly skewed Landau distribution. However, the amount of material traversed in this application is quite large, so it should be sufficient to use the mean value at each timestep. At any rate, a systematic is assessed on the interaction with matter that should cover for any inaccuracies in the modeling of energy loss.

6 Putting it all together

At each timestep, the following steps are taken:

1. Evaluate $d\vec{x}/dt$ and $d\vec{p}/dt$ using the current position/momentum \vec{x}_i/\vec{p}_i inserted into Eqs. 1 and 3 (actually, evaluate four times as per the Runge-Kutta method). Use the Runge-Kutta method to compute $\Delta\vec{x}_{B,i}$ and $\Delta\vec{p}_{B,i}$ for the timestep.
2. Look up the material at \vec{x}_i , and use this to perform multiple scattering. This is done with either the PDG (Sec. 4.1) or Kuhn (Sec. 4.2) algorithms. Use the computed angular deflection and transverse displacement (PDG only) to get $\Delta\vec{x}_{MS,i}$ and $\Delta\vec{p}_{MS,i}$ due to multiple scattering.
3. Use material constants to evaluate dE/dx with Eq. 12. Multiply by $\delta x = v\delta t$ to get δE , and use this to compute $\Delta\vec{p}_{EL,i}$.

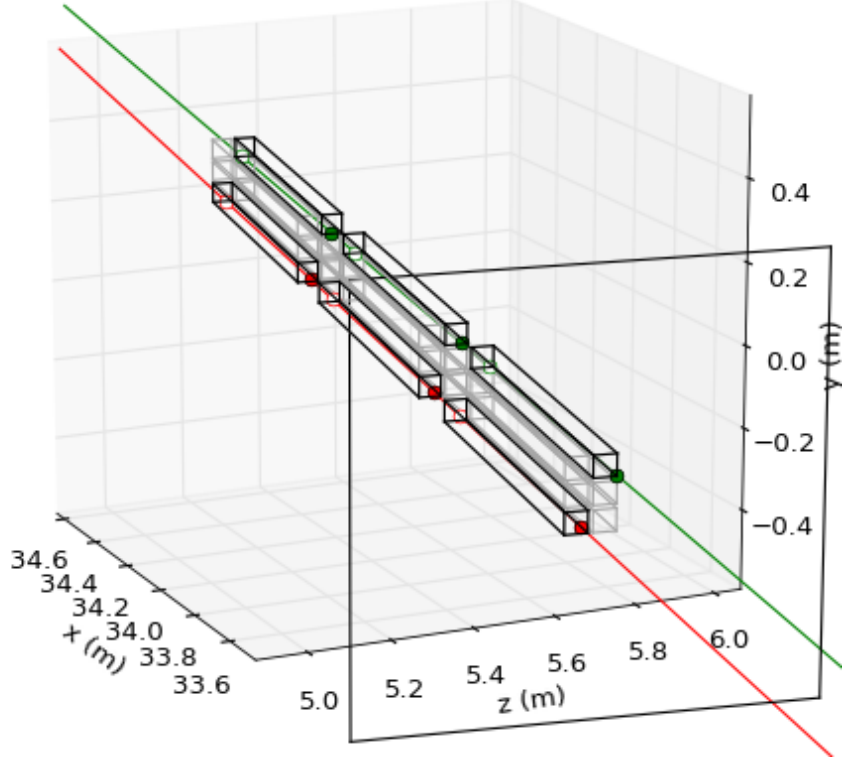


Figure 5: Visualization of two mCP trajectories passing through the Millikan demonstrator.

4. The total changes in position and momentum for the timestep are then $\Delta\vec{x}_i = \Delta\vec{x}_{B,i} + \Delta\vec{x}_{MS,i}$ and $\Delta\vec{p}_i = \Delta\vec{p}_{B,i} + \Delta\vec{p}_{MS,i} + \Delta\vec{p}_{EL,i}$.

This is done for N_{steps} timesteps (or until a specified distance cutoff is reached). At the end we get an array of $(t, x, y, z, p_x, p_y, p_z)$ values at each timestep.

7 Intersection with the Millikan Detector

To facilitate acceptance and rate calculations, tools have been added to the simulation to compute intersections with various detector models.

For simple applications, one can find intersections with an external plane and retrieve the position and momentum at the intersection. This is useful for gathering four-vectors to feed into an external simulation of the detector.

The ability to construct a “Millikan-type” detector with arbitrary numbers of bars has also been added. The entry/exit points for each individual bar can be computed, so that for each trajectory one can determine the number of bars hit, whether those bars are in a line, the length of the path traversed within the bar, etc. An example of two mCPs hitting the Millikan demonstrator is shown in Fig. 5.

References

- [1] CMS Collaboration, “Precise mapping of the magnetic field in the CMS barrel yoke using cosmic rays”. *Journal of Instrumentation* **5** (2010) no. 3, T03021. doi:10.1088/1748-0221/5/03/t03021.
- [2] ROOT, <https://root.cern/files/cms.root>
- [3] PDG, chapter 33. “Passage of particles through matter”. [link]
- [4] PDG, “Atomic and Nuclear Properties of Materials”. [link]
- [5] Kuhn, S. and Dodge, G. “A fast algorithm for Monte Carlo simulations of multiple Coulomb scattering”. *Nucl. Instrum. Methods Phys.* **A322** (1992), iss. 2, 88. doi:10.1016/0168-9002(92)90361-7.