Notes on Gamma Conversion to Muons in Geant4

NataliaToro

(equation numbers refer to the Geant4 10.2 physics reference manual)

These notes summarize my modifications to the `G4GammaConversionToMuons::PostStepDoIt()` method to include a more exact parametrization of the matrix element for muon production. The result is an almost complete rewrite of the phase-space sampling. Major conceptual changes are:

- Instead of trying to factorize the problem into sequential 1-dimensional rejection samplings of distributions in the x_+, t, ψ, and ρ kinematic variables, I sample these four variables uniformly (in the case of ρ, uniformly in $log(1 + \rho^2/c^2)$) and then rejection-sample the point in 4d phase space based on the full matrix element.
- This allows us to include the full q-dependence in the form factor, rather than just the $q_{\|}$ dependence included in the standard G4 sequential sampling.
- In addition, I have explicitly used the elastic form factor from (6.115), rather than the perplexing t-dependence in (5.60)-(5.61).
- The above also allows to sample the entire kinematically allowed range of ρ, rather than the ad-hoc range used in the standard G4 code.
- The matrix element (5.50) from Physics Reference Manual is evidently making small-angle approximations. This seems to under-populate the large-angle phase space. To model this region better, I have used a ME from the literature (Tsai 1974). This seems to *over-produce* the wide-angle tails by an O(1) factor, suggesting that I have made some mistake. But using the code in its current form for studies should be conservative.

Some things that are *not* included:

- Muon production off individual protons (and neutrons) in the nucleus. This should dominate over the elastic process at large q – in particular, at $q \gtrsim$GeV where the coherent elastic form factor is suppressed by a factor of $O(1/Z)$, there's no reason this incoherent piece wouldn't be comparable. At these large q, the recoil of the kicked nucleon is also not experimentally negligible, so we need to find a parametrization

that includes it; the recoil nucleon should go through some sort of intra-nuclear cascade. **We should think more about how to model and/or bound this process.**

- Atomic screening, relevant for $q \lesssim a_0/Z^{1/3} \sim 0.01\text{MeV}$, is not taken into account. Since the minimum q for muon production is of order $m_\mu^2/(2E_\gamma)$, this should only be relevant for $E_\gamma \gtrsim 600\text{GeV}$ and we can safely ignore it.

1 Kinematics: Variables, Transformations and Domains

We use the kinematic variables $x_\pm \equiv E_\mu^\pm/E_\gamma$ and the variables introduced in (5.51):

$$u_\pm = \gamma_\pm \theta_\pm, \qquad \gamma_\pm = \frac{E_\mu^\pm}{m_\mu}, \qquad q^2 = q_\parallel^2 + q_\perp^2, \tag{1}$$

and the variables t, ρ, and ψ defined implicitly above and below (5.54):

$$u_\pm = u \pm \xi/2, \qquad \beta = u\varphi; \qquad \xi = \rho\cos\psi, \qquad \beta = \rho\sin\psi; u^2 = 1/t - 1. \tag{2}$$

The kinematically allowed domains for these variables are:

- $x_\pm \in [\epsilon, 1-\epsilon]$ where $\epsilon \equiv m_\mu/E_\gamma$ (since $E_\mu^\pm \geq m_\mu$). Since $x_- = 1 - x_+$ by conservation of energy (under the assumption that the nucleus carries away no energy), an in-range x_+ guarantees in-range x_-.
- $\theta_\pm \in [0, \pi]$
- $\varphi \in [-\pi, \pi]$
- $u_\pm \in [0, x_\pm\pi/\epsilon]$ (follows from $\theta_\pm$ ranges).

In the high-energy limit,$u \in [0, \infty]$ so that $t \in [0, 1]$. ξ and β can have either sign, so that $\psi \in [0, 2\pi]$, but an upper limit on ρ comes from demanding that $|\varphi| < \pi$ ($\Rightarrow \rho < \pi u/|\sin\phi|$) and that $\theta_\pm$ are both positive ($\Rightarrow \rho < 2u/|cos\psi|$). The tighter of these limits always implies a cutoff $\rho \lesssim u$, with $O(1)$ dependence onψ.

For finite beam energy, small t and/or large ρ can still lead to unphysically large $u_\pm$, i.e. $\theta_\pm > \pi$. However, since this is rare (and there are several different sub-cases to check) it is easier to check for the pathology as part of the "rejection" step rather than in the initial sampling. **It's quite easy to impose a lower bound on t motivated by the u_{max} logic, and this may be worth doing.**

To summarize: to generate physical kinematics we

1. sample $x_+ \in [\epsilon, 1-\epsilon]$ (uniform sampling)

2. sample $t \in [0,1]$ (uniform sampling)
3. sample $\psi \in [0, 2\pi]$ (uniform sampling)
4. sample $\rho \in [0, \rho_{\max}]$ with $\rho_{\max} = \min(\pi u/|\sin\phi|, 2u/|cos\psi|)$ (uniformly sampling $a(\rho)$ defined below)
5. reject events with $u_\pm > x_\pm \pi/\epsilon$.

The x_+, t, and ψ phase-space distributions are approximately flat (when the form factor is neglected), so that uniformly sampling the above ranges is an efficient starting point for event generation. The ψ distribution, however, is not, motivating a further change of variables. As derived in (5.56), the ρ distribution from the matrix element (again ignoring the form factor) is approximately

$$d\sigma/d\rho \propto \frac{\rho^3}{(c_2^2+\rho^2)^2} \tag{3}$$

with $c_2 = \frac{q_\parallel}{m_\mu} = \frac{q_{\min}}{(tm_\mu)} = \frac{m_\mu}{2E_\gamma x_+ x_- t}$. A change of variables that turns this into a uniform integral is possible, but it is not easily (analytically) invertible. A practical way to match the scaling at large and small ρ is to uniformly sample $a(\rho) = \log(\rho^4 + c_2^4) \in [a(0), a(\rho_{\max})]$ — $da(\rho) \propto \frac{\rho^3}{\rho^4+c_2^4}$, which approaches (3) at large and small ρ, is strictly greater than (3), and at most a factor of 2 larger than (3).

2 Assigning Event Weights: Measure and Matrix Element Factors

Having defined a procedure for sampling phase space, we can now define an event weighting (rejection) scheme so that generated events are distributed according to the physical matrix element (5.50), up to an extra form-factor.

The above procedure will sample

$$\frac{dx_+}{1-2\epsilon}\frac{dt}{1}\frac{d\psi}{2\pi}\frac{da}{a(\rho_{\max})-a(0)}. \tag{4}$$

We wish to sample the differential cross-section

$$\frac{d\sigma}{dx_+ du_+ du_- d\varphi} \propto f(x_+, u_+, u_-, \varphi) F_{exp}(q^2), \tag{5}$$

where

$$f(x_+,u_+,u_-,\varphi)=\frac{m_\mu^4}{q^4}u_+u_-\Bigg\{\ \frac{u_+^2+u_-^2}{(1+u_+^2)(1+u_-^2)}-2x_+x_-\left[\frac{u_+^2}{(1+u_+^2)^2}+\frac{u_-^2}{(1+u_-^2)^2}\right] -\frac{2u_+u_-(1-2x_+x_-)cos\varphi}{(1+u_+^2)(1+u_-^2)}\Bigg\}. \tag{6}$$

To do so, we should weight events by

$$g(x_+,t,\psi,\rho)=(1-2\epsilon)\frac{\log\left[1+(\rho/c_2)^4\right]}{\Delta a_{ref}}fF_{exp}(q^2)\left|\frac{dx_+du_-du_-d\varphi}{dx_+dtd\psi d\rho}\right|\left(\frac{da}{d\rho}\right)^{-1}. \tag{7}$$

The normalization a_{ref} must be independent of the final-state kinematics, but should account for the overall scaling of the matrix element and of $\log\left(\frac{\rho^4+c_2^4}{c_2^4}\right)=\log(1+(\rho/c_2)^4)$ with incident photon energy.

Note that $\rho_{\max}$ is bounded from above by $\sqrt{\pi^2+2^2}u=\sqrt{\pi^2+2^2}\sqrt{1/t-1}$ for given ρ. An absolute upper bound on ρ/c_2 is therefore:

$$(\rho/c_2)_{MAX}=\sqrt{\pi^2+2^2}\cdot 2\gamma_0(x_+x_-)\sqrt{t(1-t)})\leq 2\sqrt{\pi^2+2^2}\gamma_0\tfrac{1}{4}\tfrac{1}{2}=\frac{\sqrt{\pi^2+2^2}}{4}\gamma_0, \tag{8}$$

where $\gamma_0\equiv E_\gamma/m_\mu$. Therefore, if we take $\Delta a_{ref}=\log\left[1+(\sqrt{\pi^2+2^2}\gamma_0/4)^4\right]$ then $(a(\rho_{\max})-a(0))/\Delta a_{ref}$ is always strictly less than unity.

To use g as a weight, we need to divide by its maximum possible value. The G4 physics reference manual formulas imply that the matrix element factor (including Jacobians – i.e. g above with the form-factor F and $\Delta a/\Delta a_{ref}$ factors removed) is <1 in the limit $\rho\ll u$, but in the opposite limit this is no longer the case. For example, if I take the limit of $E_\gamma\to\infty, u\to 0$ with $\varphi=\pi,\xi=u$ I find a matrix element weight proportional to $3(\pi^2+1)/16.(1-2x_{+-}\approx 2$ for x_+ near zero or 1.

(The version of the code used for our studies actually used a different, more ad-hoc Δa_{ref}: based on the fact that the form factor suppresses events with $\rho>\rho_{ff}\equiv(0.20A^{0.27}/m_\mu)^{-1}$, we defined

$$\Delta a_{ref}=\log\left[1+\left(\frac{\rho_{ff}}{c_{2,min}}\right)^4\right]=\log\left[1+(\gamma_0\rho_{ff}/2)^4\right]. \tag{9}$$

For Tungsten, these happen to be within 10-25% of one another for multi-GeV beam energies — **but a better solution would be to use the Δa_{ref} from** (8).

2.1 Form Factor

The form factor first appears in (5.60) and (5.61). It seems these are trying to implement a standard form factor (c.f. (6.115))

$$F_{exp}(q) = \left[1 + \frac{1}{12}\left(\frac{qr_n}{\hbar}\right)^2\right]^{-2} = \left[1 + (0.20A^{0.27}q/m_\mu)^2\right]^{-2} \tag{10}$$

where in the last expression we have used (c.f. (6.116)) $r_n/\hbar = 1.27A^{0.27}\,\mathrm{fm}/\hbar = 6.45/\mathrm{GeV}$, $\frac{1}{\sqrt{12}} r_n \cdot m_\mu = 0.20$.

It should be noted that even the "full" formulas for $q_\parallel$ and $q_\perp$ in the Physics Reference Manual (5.52) rely on small-angle approximations, and are inaccurate for muons at large polar angles. I have used the (5.52) approximation for q in both the form-factor calculation and the $1/q^4$ factor in the matrix element.

A Notes on Current Geant4 Formulas for sampling "t" (an "opening angle"/"invariant mass" variable)

My calculations suggest that eqns. (5.60) and (5.61), and the corresponding lines of code, have implemented the nuclear form factor incorrectly. The discrepancies in the formulas look like small "typos", but I estimate that they impact the tails of the muon distribution relevant for LDMX by 2–3 orders of magnitude, as shown in Tim's slides.

At first, I tried modifying the Geant4 implementation to take only this form factor revision into account. I found that the distribution actually did not *become more consistent with a full matrix element generator (MadGraph/MadEvent4), because the approximation that $q \approx q_\parallel$ was very inaccurate.*

Based on the formulas in §1 and (5.58), I would think the appropriate t distribution in (5.60) would be

$$\begin{aligned} f_1(t)dt &= (1 - 2x_+x_- + 4x_+x_-t(1-t))F_{exp}(q_\parallel(t))dt \\ &= \frac{1-2x_+x_-+4x_+x_-t(1-t)}{(1+(0.20A^{0.27}q_{\min}/(t\cdot m_\mu))^2)^2}dt \\ &= \frac{1-2x_+x_-+4x_+x_-t(1-t)}{(1+C_1'/t^2)^2}dt \end{aligned} \tag{11}$$

where

$$C_1' = \left(0.20A^{0.27}q_{\min}/m_\mu\right)^2 = \frac{(0.20A^{0.27})^2}{(2x_+x_-E_\gamma/m_\mu)^2}. \tag{12}$$

In the expressions (11) and (12), I have highlighted in red the differences between my formulas and (5.60) and (5.61) in the physics reference manual.

The implementation in the G4GammaConversionToMuons class seems consistent with (5.60) and (5.61), but as written these seem rather unreasonable — in particular, the effective "form factor" for any given nucleus isn't just a function of $q_{\|}$ but of $q_{\min}/t^2 \sim q_{\|}/t$, which has no physical significance that I can see.

A second issue is that f_1 should also include a term proportional to the integral of the ρ distribution. This is *not* t-independent; the approximation given in (5.56) fails badly at $t \to 0$ or $t \to 1$. While we don't care much about $t \to 1$ for LDMX (this is $u \to 0$ — forward pairs which are easier to veto), and $t \to 0$ is much *more* heavily suppressed by the form factor, it's useful to lay things out correctly so we can compare more reliably to MG.

A.1 Sampling "ρ" – and moving away from sequential sampling

The next step of generation aims to sample the ρ distribution from (5.55) and (5.56). One might think of sampling the following function:

$$f_3(\rho)d\rho = \frac{\rho^3}{(c_2^2+\rho^2)^2}\frac{F(\sqrt{q_{\|}^2+m_\mu^2\rho^2})}{F(q_{\|})}, \tag{13}$$

with $c_2 = \frac{q_{\|}}{m_\mu} = \frac{q_{\min}}{(tm_\mu)} = \frac{m_\mu}{2E_\gamma x_+ x_- t}$.

Note that this is different from the f_3 quoted in (5.64), in two ways: (i) the denominator is $(c_2^2+\rho^2)^2$ rather than $C_2+\rho^4$, and (ii) I have included the "ratio of F's" factor to account for the form-factor penalty of going to large ρ.

This approach is incorrect, though: a rejection sampling on ρ (where rejection leads to resampling ρ but *not* previously sampled kinematic variables x_+, t, and ψ), this will sample a distribution $f_3/\int f_3$ of unit norm, which is definitely not what we want! That was part of the motivation for using the multivariate rejection approach, as described elsewhere in these notes.

B Approaches to Factorizing the Matrix Element

The above approach is wrong, see first bullet below – we have a few options:

1. Sequential sampling (as done in G4) — here, the function we sample for t should include the t-dependence of the *integrals* of the ψ and ρ distributions. In practice, this doesn't seem tractable because the ρ-integral with the form factor is quite messy.

2. Multivariate rejection: if we can define the domains independently for sampling t, ψ, and ρ, then we can just sample all three and *then* reject if the *multivariate* matrix element is less than a suitably normalized uniform random number. This will be quite inefficient unless we can parametrize the distributions such that the matrix elements are pretty flat.

3. A "combination" approach: write the matrix element as a product of two functions: $A(t, \psi, \rho)$ is evaluated by sequential sampling and B by a multivariate rejection. In particular, it seems reasonable to include the form factor in B and try to do sequential sampling for the kinematic parts of the matrix element.

C Basics of Multi-Variate Sampling

Selecting random $x \in [0, x_{\max}]$ and $y \in [0, y_{\max}]$ samples

$$\frac{dx\big|_0^{x_{\max}}}{\int_0^{x_{\max}} dx} \frac{dy\big|_0^{y_{\max}}}{\int_0^{y_{\max}} dy} = \frac{dxdy\big|_{\mathcal{D}}}{x_{\max} y_{\max}} \sim dxdy\big|_{\mathcal{D}} \tag{14}$$

(the sampling must, by definition, have an integral of one – hence the division. In the above, $\mathcal{D}$ is short for the previously specified domain, and $\sim$ denotes equivalence up to overall normalization. This is equivalent to sampling $dxdy$ over the desired domain D, which differs only by normalization from the above. But when the limits of y integration depend on x, or I start doing rejection-sampling in y by a function that depends on x, I need to be more careful.

C.1 Variable-dependent Domains of Integration

As a first example: If I replace the constant $y_{\max}$ above by an x-dependent $y_{\max}(x)$, then the y selection by itself is really sampling

$$\frac{dy\big|_0^{y_{\max}}}{y_{\max}(x)} \tag{15}$$

and so the sequential x and y-selection samples

$$\frac{dx\big|_0^{x_{\max}} dy\big|_0^{y_{\max}} / y_{\max}(x)}{\int_0^{x_{\max}} dx} \sim dxdy/y_{\max}(x)\big|_{\mathcal{D}}. \tag{16}$$

This can be compensated by selecting $x \in [0, x_{\max}]$, but rather than a uniform distribution for x we select with weight $y_{\max}(x)/y_{MAX}$ ($y_{MAX} \equiv \max_{x\in[0,x_{\max}]} y_{\max}(x)$).

C.2 Rejection Sampling

To sample $dxf(x)$, where $f(x) < 1$ is a nontrivial weight, we can do the following: first, pick a uniform x, then pick a uniform random $r_x \in [0,1]$ and keep x if $r_x < f(x)$. If we only care about sampling the right distribution up to an overall proportionality factor, then we can repeat the sampling — but when we do multi-variate sampling, we need to be careful about the loop logic. What do we do when we "reject"?

Let $(x \in [0, x_{\max}])_{f(x)}$ denote the "loop" of picking x repeatedly *until* the associated $r_x < f(x)$ is satisfied. What this is really sampling (keeping track of normalization explicitly) is $\frac{dx|_0^{x_{\max}} f(x)}{\int_0^{x_{\max}} dxf(x)}$. So the following loops do different things:

$$\text{Loop A:} \quad (x \in [0, x_{\max}])_{f(x)} (y \in [0, y_{\max}])_{g(x,y)} \tag{17}$$

$$\text{Loop B:} \quad (x \in [0, x_{\max}] y \in [0, y_{\max}])_{f(x)\, g(x,y)} \tag{18}$$

Loop A samples:

$$\frac{dx|_0^{x_{\max}} f(x)}{\int_0^{x_{\max}} dxf(x)} \frac{dy|_0^{y_{\max}} g(x,y)}{\int_0^{y_{\max}} dyg(x,y)} \tag{19}$$

while Loop B samples:

$$\frac{dx|_0^{x_{\max}} f(x) dy|_0^{y_{\max}} g(x,y)}{\int_0^{x_{\max}} dx \int_0^{y_{\max}} dyf(x)g(x,y)}. \tag{20}$$

Only Loop B is equivalent (up to normalization) to the quantity we presumably wanted to sample: $dxdyf(x)g(x,y)$. We could also sample this quantity by taking

$$\text{Loop A':} \quad (x \in [0, x_{\max}])_{f(x)G(x)} (y \in [0, y_{\max}])_{g(x,y)} \tag{21}$$

$$\text{Loop C:} \quad ((x \in [0, x_{\max}]) y \in [0, y_{\max}])_{f(x)\, g(x,y)}\,, \tag{22}$$

where $G(x) \equiv \int_0^{y_{\max}} dyg(x,y)$. The crucial difference between loop A and loop C is that in the latter case, "failing" g leads to re-sampling *both* x and y, while in the former case it only resamples y. Loop A' may be intractable if the integral $G(x)$ is complicated. Loop C is slightly more efficient than B, because we abort the calculation sooner for x that fail f, but B lends itself more simply to using an exact matrix element. Moreover, if $y_{\max}$ is a function of x then, irrespective of how we do the loop, we still need to include an explicit $y_{\max}/y_{MAX}$ weight factor.