UDC 53.072

BENCHMARKING OF GDML IN COMPARISON WITH THE BUILT-IN GEOMETRY CONSTRUCTOR IN GEANT4

D.S. Kovalenko, V.V. Kobychev
Institute for Nuclear Research of NASU, Kyiv, Ukraine

Geant4 is a Monte-Carlo development toolkit which proposes multiple variants to set experiment geometry. We compare using Geometry Description Markup Language (GDML) against the traditional Geant4 geometry constructor in terms of calculation efficiency and of reproducibility and similarity of the energy spectra. The speed of calculations is equal in both cases whereas the obtained spectra are slightly different.

INTRODUCTION

In Monte-Carlo application development, based on the Geant4 toolkit [1], the geometry implementation is the one of the main concern. Geometry Description Markup Language (GDML) [2] is gaining popularity for complex detector setups due to its simplicity and customizability. The built-in GDML parsing allows reading any geometry and material setup, making it usable in multiple dedicated programs. Nevertheless, the calculation should be effective, accurate and precise, especially in complicated geometry when GDML is comfortable to use. We compare GDML with the standard Geant4 geometry constructor in terms of efficiency and result consistency.

SETUP

For evaluations of calculation efficiency, the geometry was chosen to contain multiple spherical volumes; this geometry should provide intensive trigonometric calculations for every step. We have a 21×21×21 cubic lattice of spheres of 4 cm diameter – 9261 objects in total inside the World volume which is visible as a border for traditional geometry implementation and invisible by default for GDML parsing (Fig. 1).

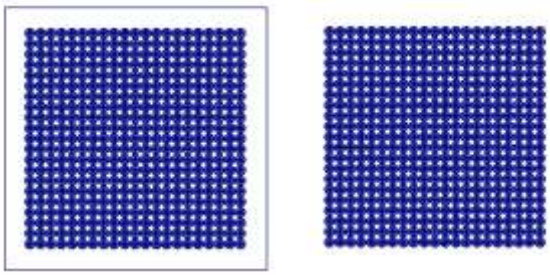


Fig. 1. Geometry setup. Left: build-in; right: GDML

In both cases the material of spheres is silicone oil with density 0.963 g/cm^3 and the chemical formula C_2H_6OSi. The elements for this material are implemented from the NIST database and mass fractions calculated as

$$\omega_i = \frac{N_i \cdot \mu_i}{N \cdot \mu_{\text{tot}}}$$

where ω_i – mass fraction of element; N_i – number of corresponding element in chemical formula; μ_i – molar mass of element; N – total number of elements; μ_{tot} – total molar mass.

The World material is air (G4_AIR) from the NIST database as well.

The particle generator is identical in both cases: it is an isotropic point source located at the center of the World volume (Fig. 2), emitting 1 MeV monoenergetic gamma quanta. Again, the tunings of the particle gun, the physics and the data collection are exactly the same; the difference might appear only at the level of geometry and materials.

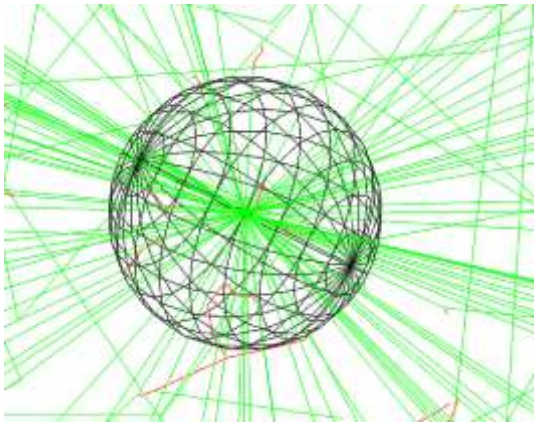


Fig. 2. Source

BENCHMARKING

As the first stage, the calculation efficiency was compared.

In the case of GDML, the solids, logical and physical volumes are parsed from a file with assignment of its parameters to the corresponding template geometrical objects represented by inherited Geant4 classes. On the other hand, one should manually inherit and assign parameters when working with default tools. So intuition suggests that we should expect the time gap appearing between these two scenarios.

Time measurements were conducted by recording the start and end times of program execution, as well as the time taken to process each event. The number of events corresponds to the number of generated gamma photons. The instance of typical calculation benchmarking is shown on Fig. 3.

This behavior (see Fig. 3) demonstrates instability of calculation process especially in multithreading mode – the same calculation may deviate in time duration. Modern systems have automatic priority leveling which can cause such sparks in calculation rate. The relative

deviation of the average event rate appears to be around 10%.

Fig. 4 demonstrates that the GDML-based simulation is faster, although it remains within a 10% deviation range.

Averaged across all threads, the processing time per event is reduced by $9.58 \cdot 10^{-7}$ s/event when using GDML.

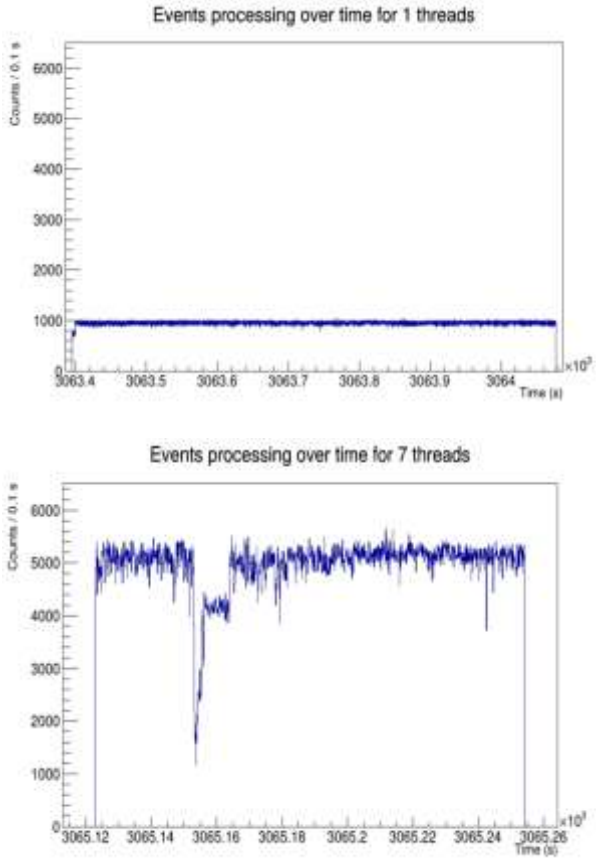


Fig. 3. The rate of events processed over system time. Top: 1 processor thread; Bottom: 7 processor threads. The bin width is 0.1 s

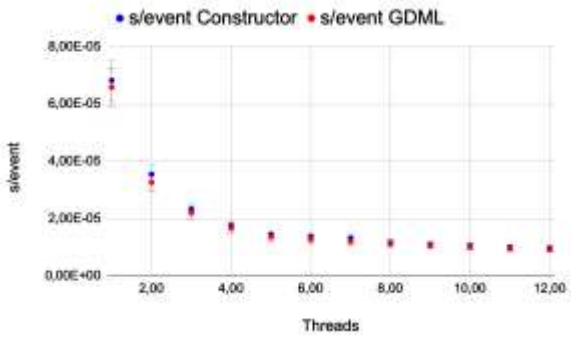


Fig. 4. Average event processing time vs. number of threads (blue dots: default constructor; red dots: GDML)

RESULTS OF SIMULATION

As shown above, the main advantage of GDML model is that it can be easily transferred to another project with different particle generators, physics lists, and data collection. Hence the main source of results deviation is materials and geometry. Fig. 5 shows that results are pretty similar.

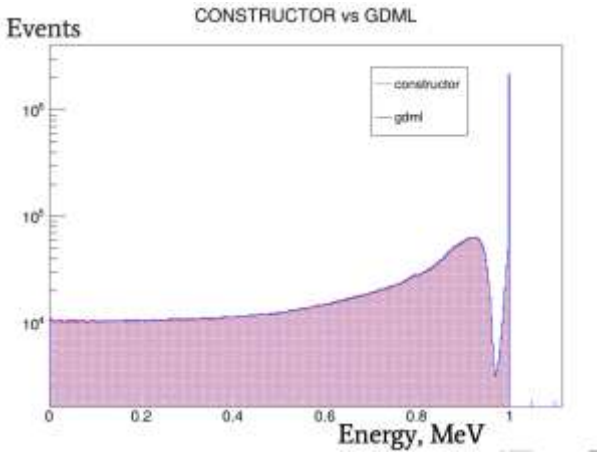


Fig. 5. The energy release spectra measured in all spheres (red: the default constructor; blue: GDML)

However, when subtracting the GDML spectrum from the default constructor spectrum (Fig. 6) or rescaling the plot (Fig. 7), one can see that the difference is 2-10 times greater than the simple statistical uncertainty ($\sqrt{N}$). It's well noticeable in the photopeak (see Fig. 7, d).

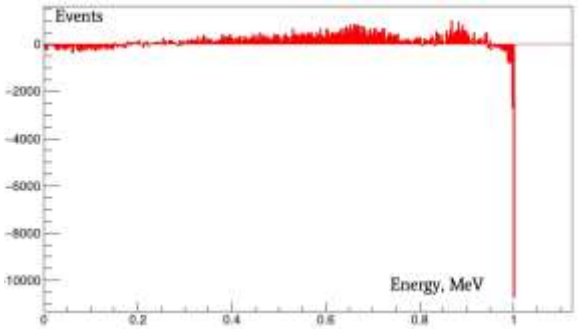
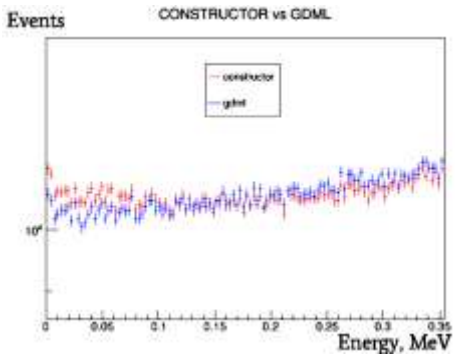
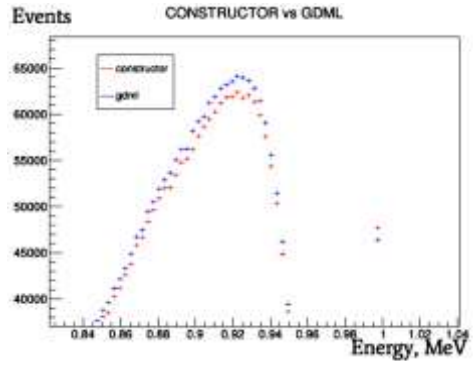


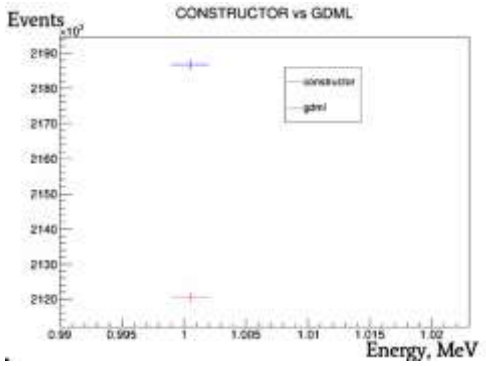
Fig. 6. The difference of two simulated energy spectra – Constructor minus GDML



a



b



c

Fig. 7. Rescaled parts of the spectra from figure 5: (a) the range of the Compton continuum; (b) the Compton edge; (c) the photopeak

CONCLUSIONS AND USAGE CONSIDERATION

The results lead to the conclusion that GDML implementation in Geant4 is calculation efficient. But one should take into account some nuances.

Complicated geometry trees and multicomponent materials can lead to differences in calculation results which are larger than any possible statistical uncertainties. Possible sources of systematic deviation may remain undetected without examining the underlying code of the GDML implementation in Geant4. Thus, for GDML, results should be carefully verified to ensure the required level of precision is achieved.

Nevertheless, GDML is perfectly optimized. The calculation rate is similar to the same application with the default geometry constructor method.

REFERENCES

1. S. Agostinelli, et al. Geant4–a simulation toolkit // *Nuclear Instruments and Methods in Physics Research Section A*. 2003, v. 506, issue 3, p. 250-303.

2. J. Allison, et al. Recent developments in Geant4 // *Nuclear Instruments and Methods in Physics Research Section A*. 2016, v. 835, p. 186-225.