EPJ Web of Conferences **337**, 01213 (2025) https://doi.org/10.1051/epjconf/202533701213
CHEP 2024

Simulation Comparison for the mSTS Geometry based on ROOT Primitive Solids and Tessellated Solids

Mehulkumar Shiroya[1,2,3,*], *Omveer* Singh[1], *Florian* Uhlig[2], *Volker* Friese[2], *Eoin* Clerkin[4], *Christoph* Blume[1], and *Alberica* Toia[1,2,3]

[1]Goethe-Universität Frankfurt, 60438 Frankfurt am Main, Germany
[2]GSI - Helmholtzzentrum für Schwerionenforschung, 64291 Darmstadt, Germany.
[3]Helmholtz Forschungsakademie Hessen für FAIR (HFHF), Max-von-Laue-Stra ße 12, 60438 Frankfurt am Main, Germany.
[4]FAIR - Facility for Antiproton and Ion Research in Europe, 64291 Darmstadt, Germany.

Abstract. The Compressed Baryonic Matter (CBM) is a heavy-ion experiment, currently under construction, at the Facility for Anti-Proton and Ion Research (FAIR) in Darmstadt, Germany. It aims to explore the QCD phase diagram at high baryon density (μ_B) using the SIS-100 accelerator at FAIR. The Silicon Tracking System (STS) is the main detector for tracking and momentum determination. A scaled-down prototype of various detector systems, including mini STS (mSTS), is being meticulously tested in the mini CBM (mCBM) experiment at the existing SIS-18 accelerator at GSI, Helmholtzzentrum für Schwerionenforschung in Darmstadt. This experiment seeks to comprehensively assess both hardware and software components, ensuring their efficacy in online readout, processing, and analyzing the intricate topological data generated by real events detected by the detector subsystems.
The recent development provides a facility to convert the Computer-Aided Design (CAD) based geometry model to Geometry Description Markup Language (GDML), an XML-based format. The representation of the solids extracted from a CAD toolkit typically consists of triangular or quadrilateral facets. The GDML file is then used in ROOT and GEANT4 using `TGDMLParser` and `G4GDMLParser` respectively to read the information of geometry volumes and to create different volume assemblies to prepare the simulation geometry to be used in the simulation.
This report presents a comparative analysis of simulation studies using two distinct representations of the mSTS geometry: one employing simplified primitive solids and the other utilizing tessellated solid-based geometry, including the run-time, secondary particle production, and z-vertex of the secondary particles.

1 Introduction

The Compressed Baryonic Matter (CBM) [1] experiment is a fixed-target heavy-ion physics experiment designed to explore the QCD phase diagram at moderate temperatures (T) and high baryonic chemical potentials (μ_B). Currently under construction at the Facility for Antiproton and Ion Research (FAIR) in Darmstadt, Germany, the CBM experiment is expected

*e-mail: M.Shiroya@gsi.de

to begin data collection in early 2028. It will operate at the SIS-100 accelerator to achieve Center-Of-Mass energy ranging from $\sqrt{s_{NN}}$ = 2.9 GeV - 4.9 GeV [1]. Figure 1 illustrates the proposed experimental setup of the CBM experiment, including its various sub-detector systems. Among these, the Silicon Tracking System (STS) is a key detector, utilizing microstrip silicon sensors to track particles and accurately determine their momentum.

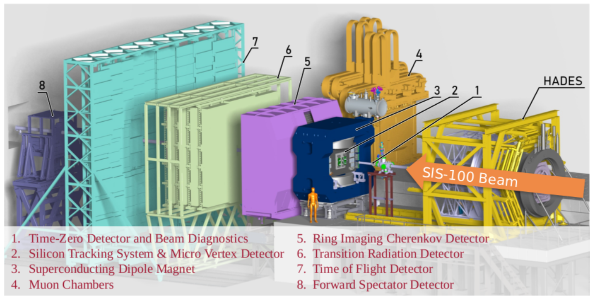


Figure 1. CBM experimental setup along with all sub-detector systems.

Currently, scaled-down prototypes of all sub-detector systems are undergoing rigorous testing to validate the detector performance for the full-scale CBM experiment. The scaled-down prototype, called mini-CBM (mCBM)[2], is operational at the SIS-18 accelerator facility at GSI. The tests of different hardware, readout data acquisition and online reconstruction software during several beam-time campaigns provide an opportunity to evaluate functionality under realistic experimental conditions. The test ensures the reliability and readiness of the full-scale CBM experiment. In this manuscript, we present a comparative analysis of simulation studies with two different representations of the mSTS geometry. This system was prepared using primitive ROOT solids and tessellated solids.

2 Procedure: CAD geometry model to ROOT framework

The detector design plays a crucial role in any physics experiment for evaluating the detector's performance and reliable data collection. It involves different key aspects such as

- Detector geometry modeling.
- Evaluation of the detector response using different particle transport models such as GEANT4[4], FLUKA[5] etc.
- Re-optimization of the detector geometry based on the simulation results.

Traditionally, the detector geometries for physics simulation are constructed using the primitive shapes provided by software toolkits like ROOT[3] and GEANT4[4]. However, over time, modern approaches have evolved to enable more precise and detailed physics modelling. The progress results from directly importing the CAD model, providing its more accurate descriptions. Figure 2 shows the steps to convert the CAD-based geometry model to the ROOT framework. The first step involves the design and modeling of the detector geometries using Computer-Aided Design (CAD) toolkits e.g. FreeCAD[6], CATIA[7]. The CAD designs are saved in Standard for the Exchange of Product Data (STEP) file format, which can be converted into Geometry Description Markup Language (GDML)[8], a XML-based format. The conversion to a GDML file can be performed using in-built functionality within those CAD toolkits or by external software such as Matter-RADiation Interac-

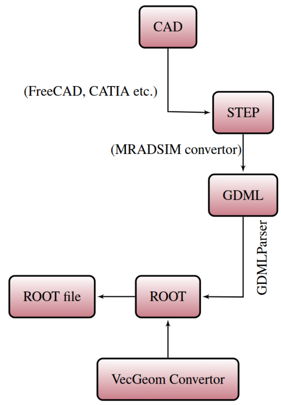


Figure 2. Flow chart for the inclusion of CAD-based geometry model into ROOT framework.

tions SIMulations (MRADSIM)[1]. After converting the STEP file of the CAD Model into a GDML file, the resulting output typically consists of triangular or quadrilateral facets, referred to as tessellated solids, which represent the meshing of various geometry volumes. The GDML file can be processed in ROOT using the `TGDMLParser`[2] or in GEANT4[4] using the `G4GDMLParser`[3], allowing for reading the different geometry volumes and creation of geometry assemblies using the functionality of such software toolkits.

In our case, the ROOT framework was used to implement the detector geometry for the simulation. Currently, ROOT's default TGeoNavigator does not support direct navigation functionality for tessellated solids[4]. Also, previous versions of ROOT contained bugs that also affected this functionality, which has been solved and added to the most recent release of ROOT (version: ≥ 6-32-00 patch). In our case, ROOT version `6.33.01` tag branch, which includes these bug fixes, compiled with the Vectorized Geometry (VecGeom)[5], a geometry modeller library that was developed as part of the GEANTV R&D project [9], to support the navigation of particles for tessellated solids within ROOT software package[10]. The VecGeom converter is added to the ROOT macro to convert geometry shapes into VGShapes and save the transformed shapes in ROOT file format. This converter allows all TGeoVolume shapes, including Tessellated ones, to be transformed into VGShapes as defined by the ROOT TGeoVGShape class[6]. The geometry saved in ROOT file format that can be used further for the particle transport for evaluating the different physics cases using different transport model based such as GEANT4[4], FLUKA[5] etc.

As shown in Figure 3, the C-frame support structure of the STS detector is approximated using primitive shapes within the ROOT geometry framework and on the other side, the direct import of CAD geometry into the ROOT framework. The direct inclusion of a geometry model from the CAD-to-ROOT method provides a detailed representation of the geometry compared to the traditional approach.

[1] https://www.mradsim.com/
[2] https://root.cern.ch/doc/master/classTGDMLParse.html
[3] https://gitlab.cern.ch/geant4/geant4/tree/75c7fd177dc4853b8a93aaedc78d6037aa36dba6/examples/extended/gdml
[4] https://root.cern/doc/master/classTGeoTessellated.html
[5] https://gitlab.cern.ch/VecGeom/VecGeom
[6] https://root.cern.ch/doc/master/classTGeoVGShape.html

Figure 3. Support structure for ladder called C-frame prepared using primitive solids (Left) and directly from a CAD into ROOT (Right).

3 mCBM Experiment: mSTS Detector Geometry

Small-scale prototypes of various detector subsystems that will be used in the CBM experiment are currently being tested, including all hardware and software at the mini-CBM experiment at the SIS-18 accelerator facility at GSI during beam-time campaigns using beams of different heavy-ion collisions such as Ni-Ni, U-U, etc.[2]. Among the detector subsystems,

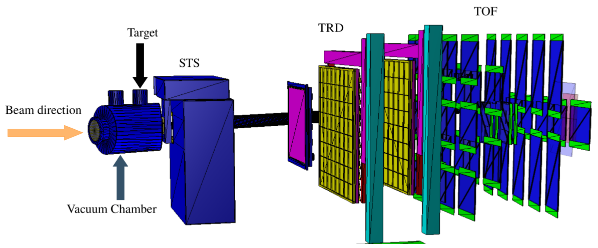


Figure 4. Experimental setup of the mCBM experiment with sub-detector system.

Figure 5. mSTS simulation geometry based on primitive solids (Left) and tessellated solid (right) used in the simulation.

the mini-Silicon Tracking System (mSTS) plays a key role in tracking charged particles. The detector systems are placed 25° angle from the nominal beam-axis, as shown in Figure 4.

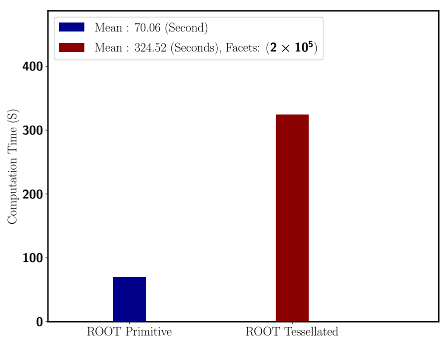
Figure 5 depicts the simplified simulation geometry implemented using different primitive shapes within the ROOT framework (Left). On the other hand, the same geometry is prepared after importing the CAD model into the ROOT framework (Right) using the procedure described in section 2. The tessellated solid-based geometry consists of a higher level of detailed description compared to the primitive shape, ensuring higher accuracy and fidelity in simulations.

4 Results

Two distinct geometries have been prepared, one using primitive ROOT shapes and the other using tessellated shapes imported from CAD. A comparative analysis of simulations is performed, including run-time comparison and secondary particle production (physics process and production vertex). The simulation performance was performed using minimum bias events generated using the UrQMD model [11] for the Ni-Ni collision system, with a beam momentum of 1.93 AGeV/c, using GEANT4 as the transport model. The simulation setup configuration included only the beam pipe, the vacuum chamber, the target, and the mSTS detector system during the run-time comparison, while for other cases, the full mCBM setup was used.

4.1 Simulation Run-time

Run-time refers to the total time required to complete all transport events in a simulation. Figure 6 (Left) compares the mean run-time for 10 jobs, each containing 100 events, performed on the GSI Virgo cluster[7] using a system equipped with an Intel Xeon Gold 6248R CPU (3.0 GHz). Notably, the tessellated solid-based geometry (Figure 5), despite the same geometry volumes as the simplified primitive solid geometry, requires approximately 4.6 times more run-time to complete the event-processing tasks. The run-time is also influenced by various



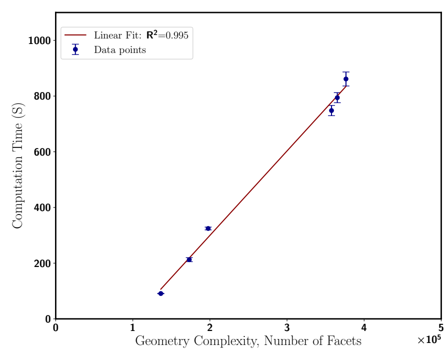


Figure 6. Run time comparison of ROOT primitive solids and Tessellated solid geometry using the ROOT TGeoNavigator (Left) and Tessellated solid-based geometry run-time, increasing the complexity of volumes (Right).

GEANT4 settings, including the number of steps after which the event transport stops, different physics process cuts, etc. These parameters also play a crucial role in physics performance and in determining computational efficiency. However, the current run-time comparison was done with the default settings of those parameters that we have in the CBM-ROOT.

In addition, including additional tessellated solids volumes in the geometry increases the complexity of the geometrical description, meaning increasing the number of facets, which in turn leads to an increase in the computational time. Figure 6 (Right) shows the average run-time for tessellated solid-based geometry as a function of the number of tessellated facets used. The run-time increases significantly, following a linear trend.

[7] https://hpc.gsi.de/virgo/

4.2 Primary and Secondary Particle Distribution

To compare the production of secondary particles, a larger statistics sample of 1000 jobs with 100 events each was used. Figure 7 presents a comparison of simulated data of the primary particle distribution based on their PDG ID for two distinct geometries of the mSTS. Since the primary particles originate directly from the input source of the UrQMD events, it is verified that no difference is observed. Figure 8 shows the distribution of secondary

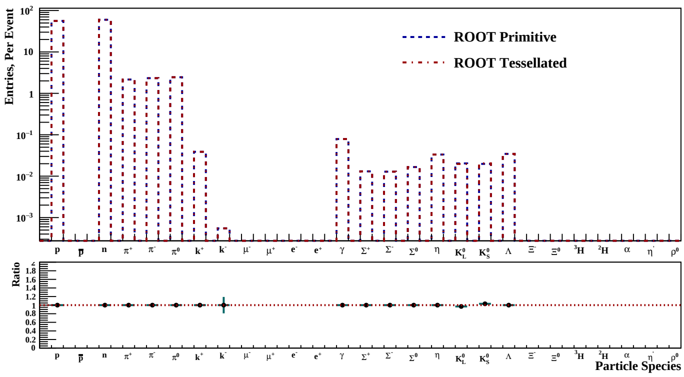


Figure 7. Primary particle distribution based on their PDG Id.

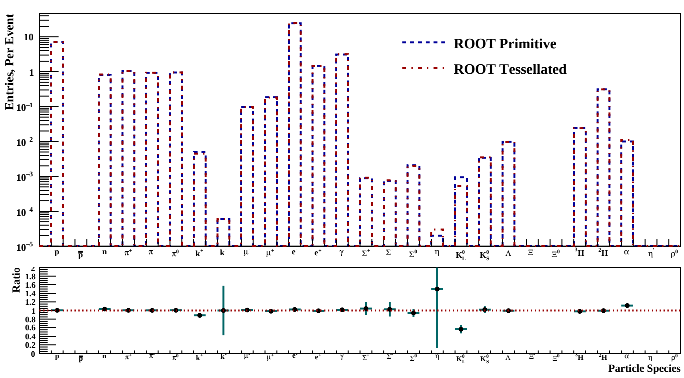


Figure 8. Secondary particle distribution based on their PDG Id.

particles, according to the PDG ID, for primitive and tessellated shape geometries, together with their ratio. The distributions agree within their statistical accuracy, except few cases of low statistics species.

4.3 GEANT4 Process

Figure 9 represents the various physics processes, including Electromagnetic (EM) and Hadronic processes, that occurred during the transport of the UrQMD events. We can observe that a few processes, such as Compton scattering, Hadronic interaction, and Coulomb scattering, exhibit a variation in the ratio ranging between 2 - 5%. Apart from these, the ratio for the remaining process shows no differences.

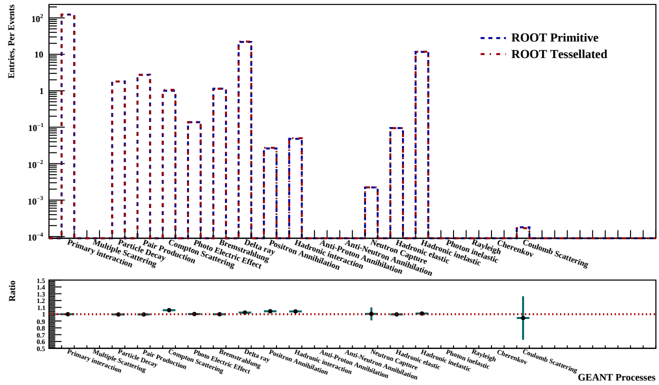


Figure 9. distribution of various physics processes occurred during the UrQMD model-based events transported using GEANT4.

4.4 Z-vertex of Secondary Particles

Figure 10 provides insights into the origin of secondary particles and shows the Z distribution of the secondary vertexes for primitive and tessellated geometries. Notably, the region

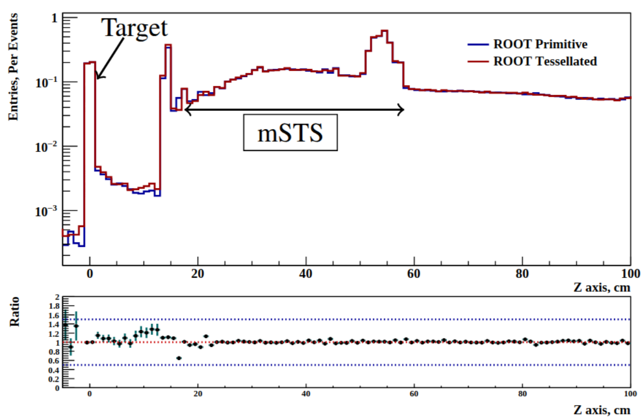


Figure 10. Z-vertex of the secondary particles MC tracks along the Z-axis.

between 17 and 58 cm, covered by the mSTS detector, does not show any significant difference. Minor differences can be observed upstream of the mSTS detector, where however the amount of secondaries produced is very much suppressed compared to the target or the mSTS and needs some investigation.

5 Summary & Conclusion

We explored the alternative CAD-to-ROOT method to represent the complex geometry for simulation within the CBM collaboration by integrating the VecGeom library package with ROOT. As discussed in section 2, the CAD-to-ROOT method simplifies the process to integrate the more complex CAD model into the simulation geometry. Our performance analysis, as discussed in the section 4.1, shows that tessellated solid-based geometry requires a significantly higher computational run-time compared to the geometry prepared using primitive solids. The increased run-time for the geometry based on tessellated solids is attributed to the algorithm iterating over all the surfaces through which the particle traversed. However, there is ongoing development to make the simulation faster for tessellated solids by optimizing algorithms such as the Bounding Volume Hierarchy (BVH) at the CERN[8] [12], which can only make the simulation faster up to some extent. Furthermore, it was found that careful preparation of geometry using primitive solids available in ROOT and with meticulous optimization of the volume representations and rigorous approximation of the material properties can lead to results consistent with those obtained from the detailed tessellated geometry. This observation suggests that there is no significant advantage from the CAD-to-ROOT tessellated solid geometry in terms of physics accuracy, while it imposes a substantial computational expense. Therefore, well-optimized primitive solid-based geometries remain the ideal choice to be used in physics simulation study. However, detailed CAD models can serve as precise references, and the CAD-to-ROOT workflow still provides significant value as a powerful tool for preparing simplified geometry and improving the physics accuracy. Additionally, there are a few aspects that still need to be understood well, which include the study of geometry load time in the simulation with increasing the number of tessellated solid facets.

References

[1] CBM Collaboration, Technical Design Report for the CBM Online Systems - Part I: DAQ and FLES Entry Stage, July 2023. https://doi.org/10.15120/GSI-2023-00739

[2] CBM Collaboration, Beamtime Application, mCBM@SIS18, A CBM full system test-setup for high-rate nucleus-nucleus collisions at GSI/FAIR, (2017). https://doi.org/10.15120/GSI-2019-00977

[3] Rene Brun and Fons Rademakers, ROOT - An Object Oriented Data Analysis Framework, Proceedings AIHENP'96 Workshop, Lausanne, Sep. 1996, Nucl. Inst. & Meth. in Phys. Res. A 389 (1997) 81-86.

[4] J. Allison et al., Recent developments in GEANT4, Nucl. Instrum. Meth. A 835 (2016) 186-225. https://geant4.web.cern.ch/

[5] F. Ballarini et al., The FLUKA code: Overview and new developments, EPJ Nuclear Sci. Technol. 10, 16 (2024). https://doi.org/10.1051/epjn/2024015

[6] Juergen Riegel, Werner Mayer, Yorik van Havre (2001-2017). FreeCAD (Version 0.16.6712)

[7] S. Belogurov et al. 2011 J. Phys.: Conf. Ser. 331 032035.

[8] R. Chytracek, J. McCormick, W. Pokorski, G. Santin, Geometry Description Markup Language for Physics Simulation and Analysis Applications, IEEE Trans. Nucl. Sci., Vol. 53, Issue: 5, Part 2, 2892-2896.

[9] Amadio G., Ananya, A., Apostolakis, J. et al. GeantV. Comput Softw Big Sci 5, 3 (2021). https://doi.org/10.1007/s41781-020-00048-6

[10] O. Singh, M. Shiroya et al. Modelling of simulation geometries using Tessellated Shapes with the Vectorized Geometry (VecGeom) package, CBM Progress Report 2023 pg. 174-175. https://doi.org/10.15120/GSI-2024-00765

[11] S. A. Bass et al. Microscopic Models for Ultrarelativistic Heavy Ion Collisions Prog. Part. Nucl. Phys. 41 (1998) 225-370.

[12] Sandro Wenzel et al, Accelerating navigation in the VecGeom geometry modeller 2017 J. Phys.: Conf. Ser. 898 072032.

[8] https://root.cern/doc/master/classbvh_1_1v2_1_1MiniTreeBuilder.html