The ALICE Geant4 Simulation

I Hřivnáčová[1], O Datskova[2], A Gheata[3], A Morsch[3], E Sicking [3,4]
For the ALICE Collaboration

[1] Institut de Physique Nucléaire (IPNO), Université Paris-Sud, CNRS-IN2P3, 91406 Orsay Cedex, France
[2] Joint Institute for Nuclear Research (JINR), Joliot-Curie 6, 141980 Dubna, Russia
[3] European Organization for Nuclear Research (CERN), CH-1211 Genève 23, Switzerland
[4] University of Münster, Schlossplatz 2, 48149 Münster, Deutschland

E-mail: ivana@ipno.in2p3.fr

Abstract. ALICE adopted a strategy to develop a virtual interface to the detector simulation codes, the Virtual Monte Carlo [1], with which the transport of particles can be performed with three different detector simulation codes: GEANT 3.21 [2], Geant4 [3], and FLUKA [4]. The Root geometrical modeller, TGeo [5], was adopted by ALICE as the unique geometry description in the simulation and reconstruction framework. This implied the integration of the TGeo geometrical modeller with all the transport codes used.

GEANT3 was the most frequently used detector transport codes in past years, however, the recent LHC data production has created a greater interest in other transport codes. In this paper we will present our experience with Geant4. We will give an overview and the present status of the tools used in the Geant4 simulation: the implementation of the Virtual Monte Carlo interface, Geant4 VMC [6], and the implementation of Geant4 geometry navigation using directly the TGeo geometry, G4Root [7]. We will also present the deployment of these tools on the Grid, the results obtained, as well as their comparison with GEANT3 and with real data.

1. Introduction

An overview of the present status of the ALICE simulation using Geant4 is given in this paper. To begin with, the tools developed and used by the ALICE collaboration are presented, they consist of: the Virtual Monte Carlo (VMC) and its specific implementation for Geant4 (Geant4 VMC); the ROOT geometrical modeller, TGeo, and finally, the G4Root navigation interface. The Geant4 distribution build and its deployment on the ALICE GRID is discussed in detail, and is followed by a short presentation and discussion of the first results obtained using these tools.

2. The ALICE simulation framework

ALICE is using a single framework for simulation, reconstruction and analysis, called AliRoot [8]. It uses the ROOT system [9] as a foundation, it is based on the Object Oriented programming paradigm, and is written in C++, except for large existing libraries, such as PYTHIA6 [10], HIJING [11], and some remaining legacy code. At the time of the CHEP 2010 conference, it included 57 packages, 178 libraries and more than 6,780,000 lines of code.

The AliRoot layout with its external dependencies is shown in Fig. 1. The ROOT package provides a foundation for the software. The core of AliRoot is represented by the steering package

(STEER). Then there are 13 central services packages, 18 event generator packages, 16 packages with detector specific code, and 8 packages for data analysis. The external packages used for simulation are the packages for the support of VMC: while the Geant3 VMC package includes the GEANT3 code itself, the Geant4 VMC and Fluka VMC packages are distributed independently from their baseline Monte Carlo. The AliEn (Alice Environment) package provides the interface to the GRID services.

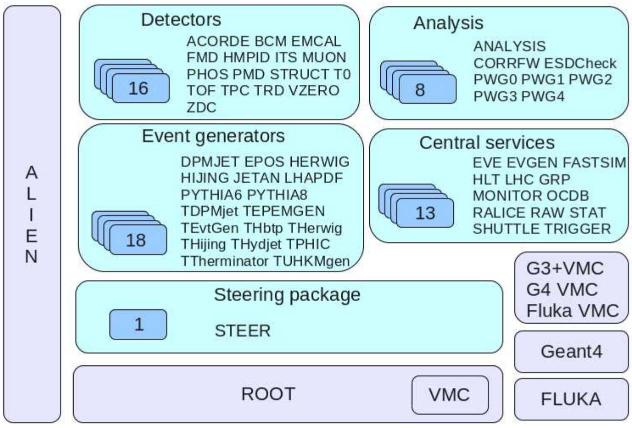

Figure 1. The AliRoot layout.

The simulation framework initiates the simulation of primary collisions and the generation of emerging particles (via event generator packages). These are transported through the detector via Monte Carlo packages that handle the simulation of deposited energy, time and position information (hits) in the detector components. The detector response is then transformed in the so-called summable digits. These are finally summed-up into digits with the optional merging of underlying events and the creation of raw data (via detector packages controlled by STEER package).

3. Virtual Monte Carlo

ALICE has developed, in close collaboration with the ROOT team, the Virtual Monte Carlo interface. The VMC has been already described in detail in [1] and [12], here only the main ideas are briefly repeated.

3.1. The VMC Concept

The VMC defines an abstract layer between the detector simulation user code and the transport Monte Carlo code (MC). In this way the user code becomes independent from the specific MC and it can be used with different transport codes GEANT3 [2], Geant4 [3], FLUKA [4] within the same simulation application.

In VMC, there is introduced on one side the interface to the transport MC itself, `TVirtualMC`, and on the other side the interface to the user application, `TVirtualMCApplication`, see Fig. 2. In this way, the dependence between the user code and the concrete MC is decoupled. Furthermore `TVirtualMCStack`, the interface to a user defined particle stack, is defined. All these classes are available together with a few more utility classes in the `vmc` package in the ROOT framework [9].

3.2. Geant4 VMC

The Geant4 VMC package provides the implementation of Virtual Monte Carlo for Geant4. It has been already described in detail in [13], so here only the main developments and corrections prompted by the ALICE experience and feedback will be mentioned.

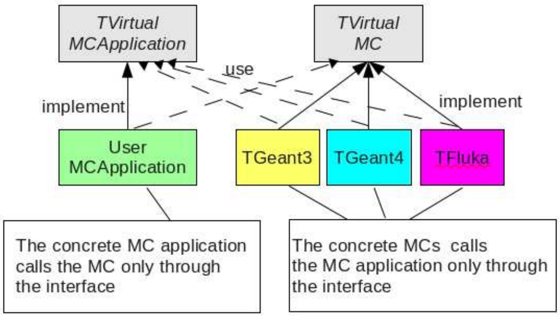

Figure 2. The VMC design.

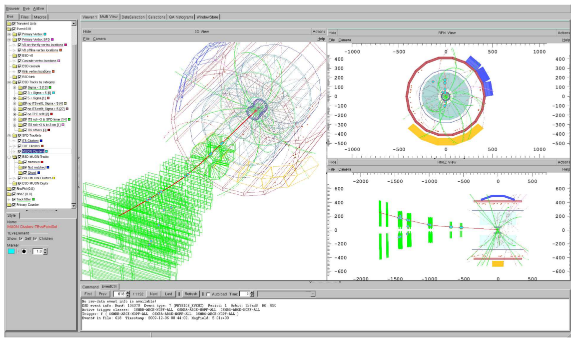

Figure 3. The integration of TGeo geometry description in the ALICE event display.

The most important new development was the implementation of a better strategy for the handling of VMC energy thresholds (commonly called "cuts"). This was done by using the G4Region objects in conjunction with G4UserLimits and applying a special algorithm. These new production thresholds limit the number of particles in the stack and consequently contribute to avoiding a severe memory problem that commonly occurred when running the standard ALICE benchmark for pp collisions. Amongst other new features it is worth noting the new VMC controls implementation. With these controls it is possible to activate, or deactivate, selected physics processes in different tracking mediums. Through new user commands it is now also possible to select other models for energy loss and/or energy fluctuations and to customize the options used in the optical processes.

Several fixes in the tool, made thanks to the ALICE feedback, contributed to its robustness and future stability.

4. ROOT geometry tools

4.1. TGeo

For geometry detector description, ALICE is using TGeo [5]. The TGeo geometry package is a general purpose HEP toolkit inside ROOT which was implemented following closely the needs of ALICE. It is used also by other experiments: PANDA, CBM, OPERA, STAR.

It provides the full geometry description of the ALICE detector and also the navigation functionality used in simulation. It is also able to store and directly apply misalignment within the geometry itself. It provides built-in checking tools, a geometry builder and a powerful visualization. While mainly used in for simulation, ALICE is using the built-in TGeo features in reconstruction and event display.

Fig. 3 is a snapshot that demonstrates the integration of the TGeo geometry description in the event display, which is run online in the ALICE control room.

4.2. G4Root

In order to allow navigation using the TGeo geometry modeller, the navigation interfaces had to be implemented for each Monte Carlo code. The G4Root package [7] implements the Root geometry navigation interface for Geant4. In a more detailed view, it implements the `G4VUserDetectorConstruction::Construct()` interface to build a Geant4 native geometry skeleton with links to TGeo geometry objects.

The implementation of the `G4Navigator` interface, using TGeo navigation methods as the back-end, is an important component of G4Root. It provides all navigation queries like `ComputeStep()`, `LocateGlobalPointAndSetup()`, `ComputeNormal()`.

It was tested and benchmarked against the Geant4 native geometry. The tests show comparable speed while giving the same results.

5. The GRID deployment

The ALICE production software packages currently deployed on the GRID consist of the following: ROOT, Geant3 and AliRoot. The Geant4 package has been tested with production during 2010. It was designed to be consistent and maintainable for users within the ALICE GRID. The complete GRID package contains all Geant4 related packages: CLHEP, Geant4, Geant4 VMC and data files needed by Geant4. Including Geant4 VMC inside the package makes it dependent on ROOT.

ALICE has developed the BITS [14] build system, which provides a systematic and uniform process for compiling and packaging distributions. The build process is designed to be consistent. On the first build, a package configuration (specific versions of included software and environment setting) is defined, built and packaged, ready for GRID installation. In subsequent rebuilds no user input is required, as the package and all relevant dependencies are installed automatically by the build system.

The distribution packages are built for multiple platforms and deployed on the ALICE GRID through the centralized AliEn package [15] manager. The package is then installed automatically on all sites requesting it. GRID users simply specify the distribution version they want to use in their job description. They can also run an automated script and install the compiled Geant4 distribution with relevant ROOT dependency locally on their machine. In this way the version synchronization between user and GRID installations is ensured.

6. First results

ALICE's central Geant4 test productions were run on GRID in July and October 2010. p+p collisions at $\sqrt{s} = 7$ TeV were simulated using the same detector configuration active during the data taking of one selected ALICE run in May 2010. 3,207,371 events were collected during this selected run. The simulations were performed with ROOT v5.26/00b, AliRoot v4-18-Rev-23 and two Geant4 versions: Geant4 9.3.p02 and Geant4 9.4.b01. For the same run, the official GRID simulation using GEANT3 was also performed. Both simulations with the two different transport Monte Carlos used the event generator PYTHIA6 with the so-called "Perugia-0" tune in use among the LHC collaborations [16].

Several Geant4 simulations, each with 100,000 events, were performed for several Geant4 physics lists. The resulting data are available on the GRID and on the ALICE Proof facilities (CAF, SKAF) and their analysis and comparison with real data and GEANT3 simulation are ongoing. The first results are shown in Figures 4 - 7.

In Fig. 4, properties of Monte Carlo secondary particles for several physics list options are compared to GEANT3. The distributions of kinetic energy and polar radius of generation of secondary particles are presented, as well as their azimuthal angle. The shapes of the distributions do not show big differences, except for the kinetic energy distribution where Geant4 produces more low energy particles. In general, the number of secondary particles is lower in Geant4 than in GEANT3.

In the next figures, the first results from detector sub-system validation are shown. In Fig. 5, the track matching efficiency between the TRD (Transition Radiation Detector) and the TPC (Time Projection Chamber) sub-systems is presented. Geant4 appears slightly higher than GEANT3 and data, this is under investigation. In Fig. 6, the track matching efficiency between the ITS (Inner Tracking System) and the TPC sub-systems shows that Geant4 and GEANT3 reproduce the data well. In Fig. 7, a validation plot from the HMPID (High Momentum Particle Identification) detector is presented. Geant4 fits data very well.

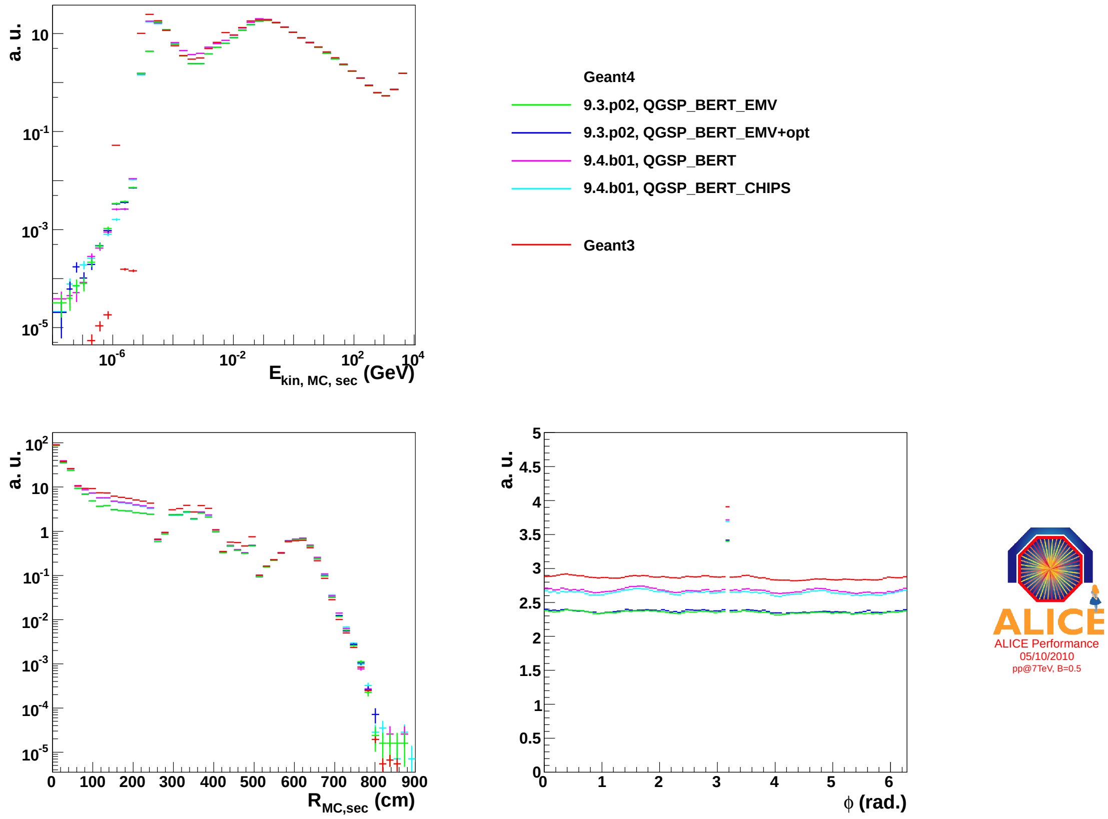

Figure 4. Properties of Monte Carlo secondary particles.

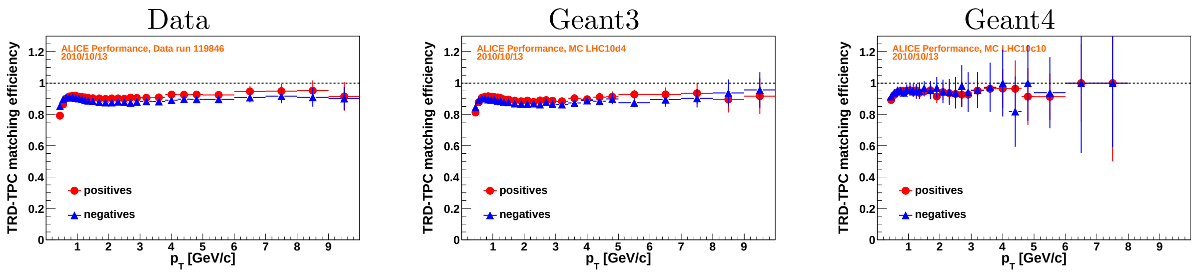

Figure 5. The TPC-TRD track matching efficiency.

7. Conclusions

This paper gives a brief overview of the ALICE simulation framework and the ROOT tools used in simulation: Virtual Monte Carlo, the Root geometry modeller TGeo, and the G4Root navigation interface. Geant4 is used by ALICE as a second Monte Carlo, after GEANT3. The first test productions with Geant4 were run on GRID in 2010. Produced data are available on the GRID and Proof facilities and their analysis and comparison with real data and GEANT3 simulation is ongoing. The ALICE collaboration continues to show an active interest in Geant4, in particular with the recent LHC data.

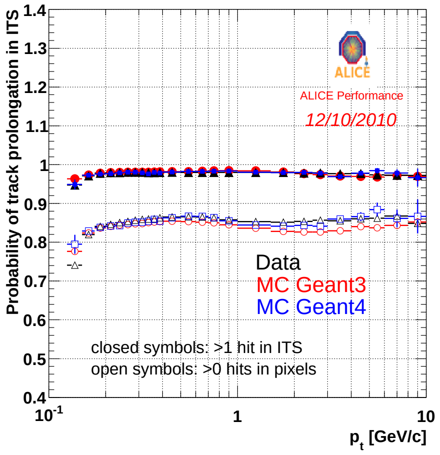

Figure 6. The TPC-ITS track matching efficiency.

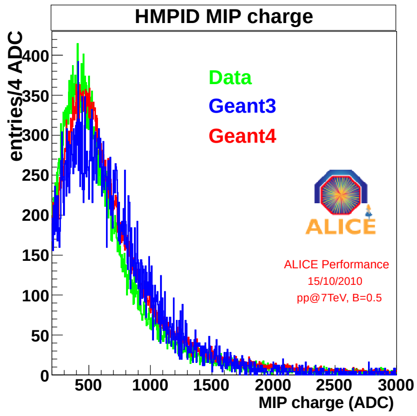

Figure 7. The validation plot from HMPID detector.

Acknowledgments

The authors would like to acknowledge all detector experts who performed their analysis on the Geant4 productions in a short time so that their results could be included in the conference presentation and this paper. Namely we would like to thank A. Dainese from *Laboratori Nazionali di Legnaro, INFN, Legnaro, Italy*, M. Fasel and A. Andronic from *Research Division and ExtreMe Matter Institute EMMI, GSI Helmholtzzentrum für Schwerionenforschung, Darmstadt, Germany* and G. Volpe from *Dipartimento Interateneo di Fisica Ḿ. Merlinánd Sezione INFN, Bari, Italy.*

References

[1] Hřivnáčová I *et al.* 2003 *Proc. of Computing in High Energy and Nuclear Physics* (La Jolla) pp THJT006, arXiv:cs/0306005
[2] Brun R *et al.* 1985 *GEANT3 User Guide* (CERN Data Handling Division, DD/EE/84-1)
[3] The GEANT4 Collaboration, Agostinelli S *et al.* 2003 *Nucl. Instrum, and Methods* **A506** 250-303
The GEANT4 Collaboration, Allison J *et al.* 2006 *IEEE Transactions on Nuclear Science* **53** **1** 270-278
[4] Fasso A *et al.* 2001 *Proc. of the MonteCarlo 2000 Conference* (Lisbon, Springer Verlag Berlin) 159-164 and 955-960.
[5] Brun R, Gheata A and Gheata M 2003 *Proc. of Computing in High Energy and Nuclear Physics* (La Jolla) pp THMT001, arXiv:physics/0306151
[6] http://root.cern.ch/drupal/content/geant4-vmc
[7] Gheata A and Gheata M 2008 *J. Phys: Conf. Series* **119** 042014
[8] http://aliceinfo.cern.ch/Offline
[9] http://root.cern.ch
[10] Bengtsson H U and Sjostrand T 1987 *Comput. Phys. Commun.* **46** 74
Sjostrand T 1994 *Comput. Phys. Commun.* **82** 74
http://home.thep.lu.se/ torbjorn/Pythia.html
[11] Wang X N and Gyulassy M 1991 *Phys. Rev.* **D44** 3501
Gyulassy M and Wang X N, 1994 *Comput. Phys. Commun.* **83** 307331
http://www-nsdth.lbl.gov/ xnwang/hijing/
[12] Carminati *et al.* 2004 *Proc. of Computing in High Energy and Nuclear Physics* (Interlaken) pp 433
[13] Hřivnáčová I 2008 *J. Phys: Conf. Series* **119** 032025
[14] http://alienbuild.cern.ch:8889
[15] http://alien2.cern.ch/
[16] Sjostrand T, Skands P 2005 *Eur. Phys. J.* **C39**, 129; Skands P 2008 *Multi-Parton Interaction Workshop* (Perugia, Italy) 2831; 2009 arXiv:0905.3418 [hep-ph] 482.