Status of JUNO Simulation Software

Ziyan Deng[*] On behalf of the JUNO Collaboration

Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, China

Abstract. The JUNO (Jiangmen Underground Neutrino Observatory) experiment is a multi-purpose neutrino experiment designed to determine the neutrino mass hierarchy and precisely measure oscillation parameters. It will be composed of a 20k ton liquid scintillator (LS) central detector equipped with about 18000 20-inch photon-multipliers (PMTs) and 25000 3-inch PMTs, a water Cherenkov detector with about 2000 20-inch PMTs, and a top tracker. Monte-Carlo simulation is a fundamental tool for optimizing the detector design, tuning reconstruction algorithms, and performing physics study. The status of JUNO simulation software will be presented, including generator interface, detector geometry, physics processes, MC truth, pull-mode electronic simulation.

1 JUNO introduction

The Jiangmen Underground Neutrino Observatory (JUNO) [1] is a multi-purpose neutrino observatory under construction in the southeast of China. It will be located at a distance of 53km from two major nuclear power plants, Yangjiang and Taishan. The JUNO detector consists of a central detector, a water Cherenkov detector and a muon tracker. The central detector is a liquid scintillator (LS) detector of 20 kton mass with a designed energy resolution of 3% at 1MeV, equipped with about 18000 20-inch Photomultiplier Tubes (PMTs) and 25000 3-inch PMTs. The central detector is submerged in a water pool to be shielded from natural radioactivity from the surrounding rock and air. The water pool is equipped with about 2000 20-inch PMTs to detect the Cherenkov light from cosmic muons, acting as a veto detector. On top of the water pool, there is another muon detector to accurately measure the muon tracks. A schematic view of the JUNO detector is shown in Fig.1.

To achieve the unprecedented energy resolution is very challenging. A Geant4 [2][3] based Monte Carlo simulation software has been developed based on the Monte Carlo of the Daya Bay experiment, which has been carefully tuned to match observed detector distributions, such as the liquid scintillator light yield, charge response, and energy non-linearity. The JUNO simulation software is used to study the detector response and optimize the detector design. It consists of three components: kinematic generator, detector simulation and electronics simulation.

[*] Corresponding author: dengzy@ihep.ac.cn

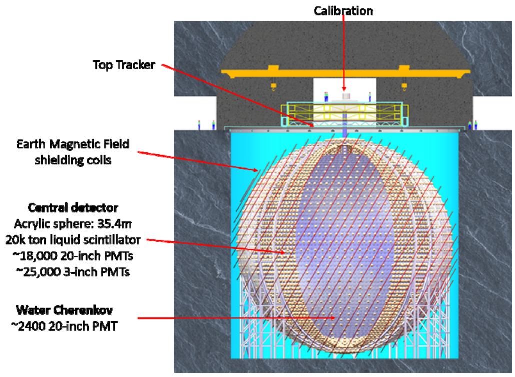

Fig. 1. Schematic of some of the main parts of the JUNO detector, consisting of a 35.4m diameter sphere filled with 20 ktons of liquid scintillator surrounded by around 43,000 PMTs and immersed in ultrapure water.

2 Offline software framework

JUNO offline software is developed based on the SNiPER [4] framework, which is written in C++ and python, inspired by Gaudi [5]. The basic components in SNiPER include Task, Algorithm, Service and Tool, as shown in Fig.2. Task is for application management which can load algorithms, manage and execute algorithms. Users' algorithms get data from the event buffer, execute calculations and put results back into the buffer. Services are generally sizeable components that are setup and initialized once at the beginning of the job by the framework and used by many algorithms as often as they are needed. Tools are light weight objects whose purpose is to help other components perform their work. Algorithms and Services can declare themselves as Tools parents.

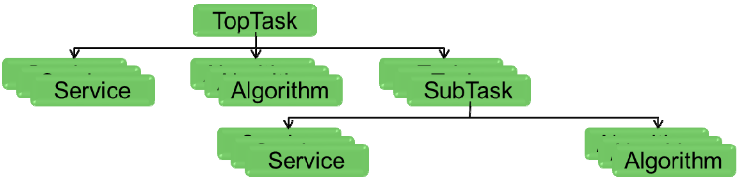

Fig.2. Components in SNiPER framework.

The event buffer management is designed to manage event data [6], send data to algorithms and get results from algorithms. The event buffer management in SNiPER is optimized for neutrino experiments, to enable retrieving the events in a specific time window. The event data model also builds relations of event data in different processing stages, from GenEvent to SimEvent, ElecEvent, CalibEvent, RecEvent, as shown in Fig.3. It helps users to do navigation inside an event, for example: navigating from a reconstructed event back to MC particles. It also helps users to associate signals from a simulated IBD event.

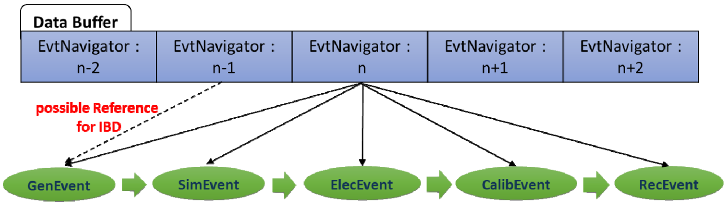

Fig.3. Event buffer management in SNiPER.

3 Simulation Software

Detector simulation of JUNO is developed based on Geant4 and the SNiPER framework. All components in simulation have been designed and implemented, including generator interface, geometry construction, physics processes, optical model of LS and PMT, and MC observables, as shown in Fig.4.

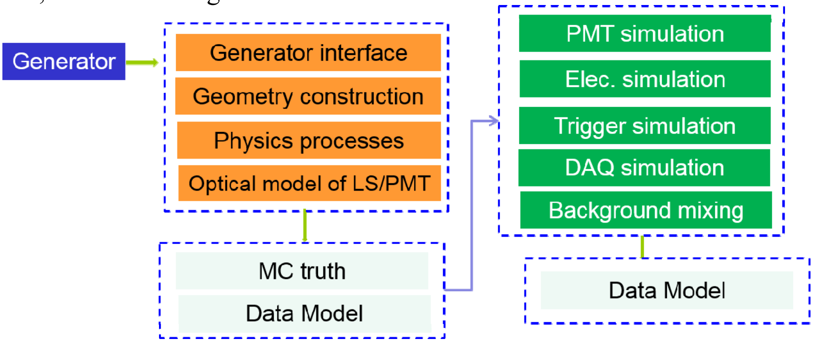

Fig.4. Components in simulation software.

3.1 Generator interface

A flexible generator interface is implemented to handle different types of generators (C/C++, Fortran, Python, ...). Any kind of generator output can be converted by a parser to HepMC::GenEvent Object, which will be converted to G4Event for detector simulation. For example, HepEvt2HepMC is a parser which converts HepEvt to HepMC. A Positioner tool is designed to set the generated position for each event. The position can be randomly inside a specific volume or a specific material.

3.2 Geometry management

Fig.5 shows GDML-based geometry management in offline software. Text files serve as input of Geant4 detector construction and Geant4 converts the geometry to GDML [7] file and ROOT [8] geometry objects, which are output together with simulated data to ensure consistency between the geometry data and simulated event data. The ROOT geometry objects are input of the geometry service, which provides consistent detector description for different applications: simulation, reconstruction, event display, and data analysis. The geometry service also provides convenient functions to identify different types of PMTs, locate the boundary of each PMT, and search for the neighbouring PMTs.

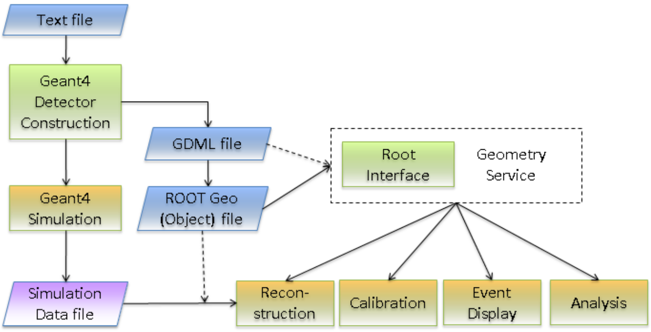

Fig.5. GDML-based geometry management in offline software.

3.3 Physics processes

The original Geant4 version used to develop JUNO detector simulation software was Geant4 9.4. The physics processes together with LS and PMT optical model were validated with Daya Bay experiment data. The latest progress achieved in the last year is updating the Geant4 version from 9.4 to 10.4, to study the parallelized simulation based on Geant4 10.4. In the latest detector simulation software, physics constructors from Geant4 10.4 are added into JUNO physics list.

- G4EmLivermorePhysics, G4EmExPhysics
- G4DecayPhysics, G4RadioactiveDecayPhysics
- G4HadronElasticPhysicsHP, G4HadronPhysicsQGSP_BERT_HP
- G4StoppingPhysics
- OpticalPhoton Processes
  - G4Scintillation, G4Cerenkov, G4OpAbsorption
  - G4OpRayleigh, G4OpBoundaryProcess

For the G4RadioactiveDecayPhysics, Li9/He8 decay is modified [9] based on the available nuclear measurements. The ^{9}Be excited states should break into a neutron and two alphas, and ^{8}Li excited states emit a neutron and possibly an alpha and a triton depending on the decay chain, whereas in Geant4 they both reach the ground state by emitting a gamma. Re-emission is added in the G4Scintillation process. The neutron capture process is also modified. Detailed validation has been performed to compare between Geant4 versions 9.4 and 10.4. Low energy and high energy particles are generated in liquid scintillator to validate different physics processes. Alpha particles and protons with several MeV kinematic energy have different simulation results in these two Geant4 versions. In Geant4 10.4, alpha particles and protons propagate for far fewer steps (3-10 in 10.4 compared with 60-120 in 9.4), shorter step length and deposit more energy at each step.

3.4 Fast simulation

Muon can produce millions of optical photons in the JUNO central detector. It is very challenging to perform a full optical simulation, since it occupies a huge amount of CPU memory and takes very long time (several hours) to simulate one event. Different kinds of fast simulation methods are studied and developed, for example, CPU-based voxel methods [10], the GPU-based Opticks [11] method. A unified deferred optical propagation method proposed to integrate different optical photons simulators, as shown in Fig.6. GenSteps are collected during the detector simulation with optical photon generation and propagation disabled. GenSteps can be sent to different optical photon simulators which can run on CPU or GPU. The simulators are responsible for photon generation and propagation, then output Hits information.

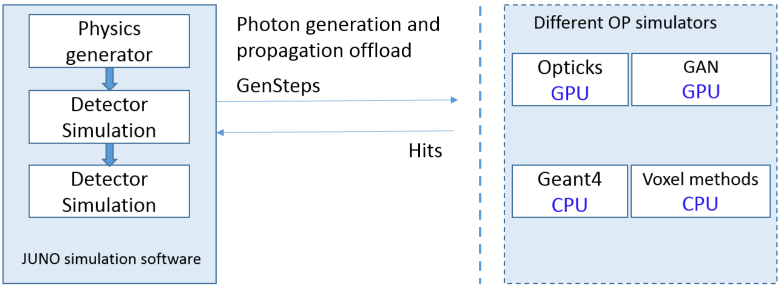

Fig. 6. Deferred optical propagation method to integrate different optical photon simulators.

3.5 Electronics simulation

The Geant4 based tracking code simulates optical photons until they are detected by the PMTs. The Electronics simulation code reproduces the electronics chain, including the PMT pulses, trigger system and readout system. A "Back-driven" strategy is designed and developed for electronics simulation, which can well handle the time correlation and well manage the memory. It works in pipe-line mode and enable hit-level background mixing. Results from PMT mass testing at Pan-Asia are used in PMT simulation. Detailed implementations for electronics and trigger is based on the real design.

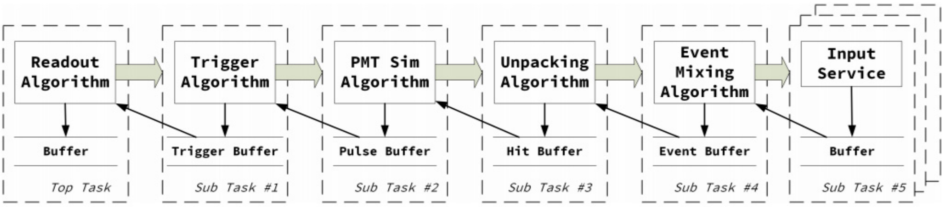

Fig.7. Workflow of electronics simulation.

3.6 MC truth

For the MC truth in detector simulation, different kinds of analysis elements are designed and developed to save the MC truth information. The analysis element is an independent Tool in the SNiPER framework, each kind of analysis element is designed for one dedicated kind of MC truth information, and can record MC truth during Geant4 Run/Event/Tracking/Stepping stage.

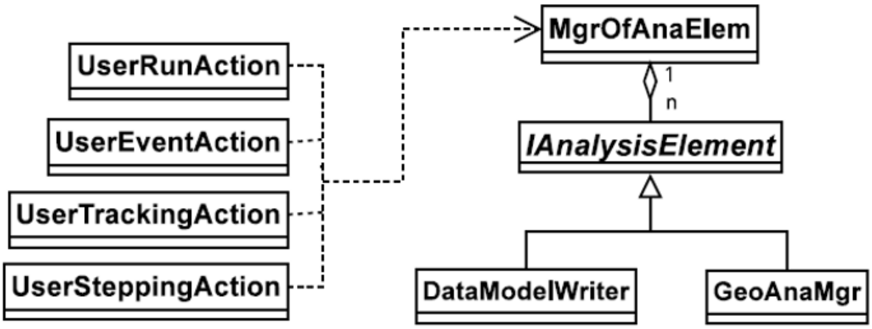

Fig. 8. Design of MC truth in detector simulation software.

For the truth information in electronics simulation, ElecTruth Header added to match DetSim hits and ElecSim pulses. It can help to understand the waveform reconstruction results.

3 Computing performance

The simulation software has been used for the tuning of reconstruction algorithms, study of calibration strategy and various kinds of physics studies. The computing performance of simulation software is shown in Table 1.

Table 1. Computing performance of simulation software

Calibration source, 60Co, 1.33MeV + 1.17MeV	
Detector Simulation	
Size/event	0.09 MB
CPU time/event	3.7 s
Electronics Simulation	
Size/event	4.9MB
CPU time/event	1.2 s

4 Summary and outlook

All components in simulation have been designed and implemented in the SNiPER framework. The latest detector simulation software was released based on Geant4 10.4, with some physics processes modified for the JUNO experiment. Electronics simulation supports a "PULL" workflow and allows hit level mixing. The full data processing chain for data production is ready. Muon simulation is a big challenge. Several fast simulation methods have been studied. A unified deferred optical propagation is under development, to integrate different optical photon simulators.

This work is supported by the National Natural Science Foundation of China (11575224), the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant No. XDA10010900, Youth Innovation Promotion Association, CAS.

References

1. F. An et al. (JUNO Collaboration), J. Phys. G 43, no. 3, 030401 (2016)
2. S. Agostinelli et al. (GEANT4 Collaboration), Nucl. Instrum. Meth. A506 250-303 (2003)
3. https://geant4.web.cern.ch/
4. J. H. Zou, X.T. Huang, W.D. Li, T. Lin, K. Zhang, Z.Y. Deng and G.F. Cao, J. Phys. Conf. Ser. 664 072053 (2015)
5. G. Barrand et al. Comput. Phys. Commun. 140 45-55 (2001)
6. T. Li, X. Xia, X.T. Huang, J.H. Zou, W.D. Li, T. Lin, K. Zhang and Z.Y. Deng Chin.Phys. C41 (2017) no.6, 066201
7. Chytracek R, McCormick J, Pokorski W and Santin G, IEEE Trans. Nucl. Sci. 53 2892 (2006)
8. Brun R and Rademakers F, Nucl. Instrum. Meth. A389 81-86 (1997)
9. C. Jollet, A. Meregaglia, Nucl. Instrum. Meth. A 949 162904 (2020)
10. T. Lin, Z.Y. Deng, W.D. Li, G.F. Cao, Z.Y. You, X.Y. Li, Chinese Physics C Vol. 40, N0. 8 (2016) 086201
11. Simon Blyth, EPJ Web Conf. 214 02027 (2019)