HCMUS

Viet Nam National University
Ho Chi Minh City
University of Science

24th IEEE Real Time Conference, ICISE, Quy Nhon, Vietnam, April 22-26, 2024

ID #168

Virtualizing Experimental Setups Based on GATE/GEANT4 Monte Carlo Simulation Results

Hoang Thi Kieu Trang[1,2], Martin Grossmann[3], Huynh Anh Dung, Lam Chanh Khoa, Van Thi Thu Trang[1,2], Phan Le Hoang Sang[1,2], Nguyen Tri Toan Phuc[1,2], Chau Thanh Tai[1,2], Tran Thien Thanh[1,2], Chau Van Tao[1,2]

1. *Department of Nuclear Physics, Faculty of Physics and Engineering Physics, University of Science, Ho Chi Minh City, Vietnam*
2. *Vietnam National University, Ho Chi Minh City, Vietnam,*
3. *Paul Scherrer Institute - Center for Proton Therapy, Switzerland*

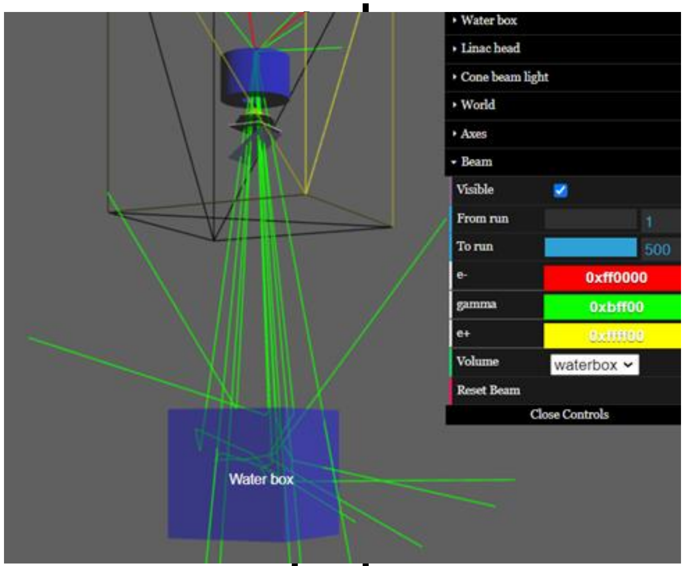
Email: htktrang@hcmus.edu.vn, martin.grossmann@psi.ch

Virtualizing Experimental Setups Based on GATE/GEANT4 Monte Carlo Simulation Results

GATE/GEANT4 Monte Carlo simulation
- Linux, MacOS, OpenGL visualization
- Window, Qt visualization is not yet available

GEANT4 or Gate scripted commands (macro)
definitions of geometry, systems, digitizer, physics, source, output

ASCII output
information of events (tracks and steps)

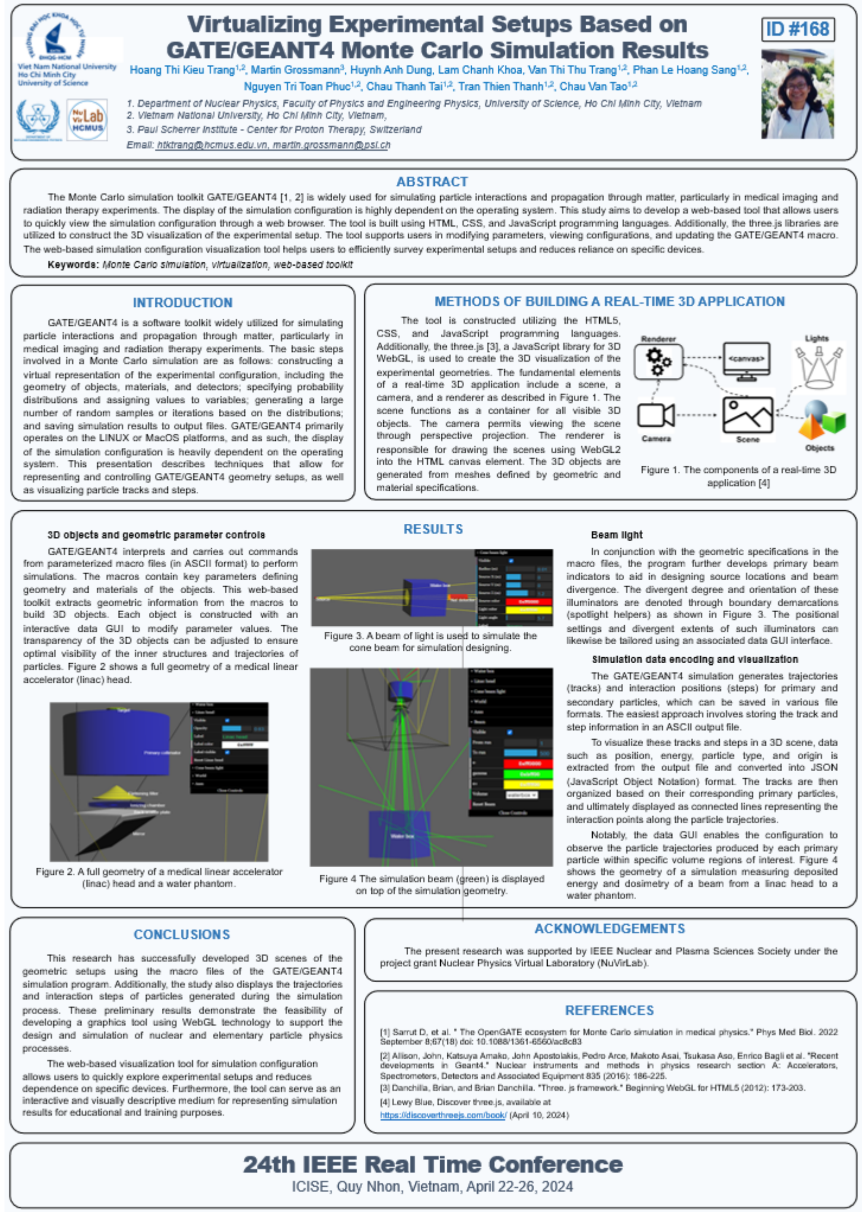


Extract geometry and beam parameters

Encode track and step information

Web-based 3D visualization
- Framework: Nodejs, HTML5, JavaScript, threejs library
- Goals: works across multiple devices

Virtualizing Experimental Setups Based on GATE/GEANT4 Monte Carlo Simulation Results



Virtualizing Experimental Setups Based on GATE/GEANT4 Monte Carlo Simulation Results

ID #168

Viet Nam National University Ho Chi Minh City University of Science

Hoang Thi Kieu Trang[1,2], Martin Grossmann[3], Huynh Anh Dung, Lam Chanh Khoa, Van Thi Thu Trang[1,2], Phan Le Hoang Sang[1,2], Nguyen Tri Toan Phuc[1,2], Chau Thanh Tai[1,2], Tran Thien Thanh[1,2], Chau Van Tao[1,2]

1. Department of Nuclear Physics, Faculty of Physics and Engineering Physics, University of Science, Ho Chi Minh City, Vietnam
2. Vietnam National University, Ho Chi Minh City, Vietnam,
3. Paul Scherrer Institute - Center for Proton Therapy, Switzerland

Email: htktrang@hcmus.edu.vn, martin.grossmann@psi.ch

ABSTRACT

The Monte Carlo simulation toolkit GATE/GEANT4 [1, 2] is widely used for simulating particle interactions and propagation through matter, particularly in medical imaging and radiation therapy experiments. The display of the simulation configuration is highly dependent on the operating system. This study aims to develop a web-based tool that allows users to quickly view the simulation configuration through a web browser. The tool is built using HTML, CSS, and JavaScript programming languages. Additionally, the three.js libraries are utilized to construct the 3D visualization of the experimental setup. The tool supports users in modifying parameters, viewing configurations, and updating the GATE/GEANT4 macro. The web-based simulation configuration visualization tool helps users to efficiently survey experimental setups and reduces reliance on specific devices.

Keywords: *Monte Carlo simulation, virtualization, web-based toolkit*

INTRODUCTION

GATE/GEANT4 is a software toolkit widely utilized for simulating particle interactions and propagation through matter, particularly in medical imaging and radiation therapy experiments. The basic steps involved in a Monte Carlo simulation are as follows: constructing a virtual representation of the experimental configuration, including the geometry of objects, materials, and detectors; specifying probability distributions and assigning values to variables; generating a large number of random samples or iterations based on the distributions; and saving simulation results to output files. GATE/GEANT4 primarily operates on the LINUX or MacOS platforms, and as such, the display of the simulation configuration is heavily dependent on the operating system. This presentation describes techniques that allow for representing and controlling GATE/GEANT4 geometry setups, as well as visualizing particle tracks and steps.

METHODS OF BUILDING A REAL-TIME 3D APPLICATION

The tool is constructed utilizing the HTML5, CSS, and JavaScript programming languages. Additionally, the three.js [3], a JavaScript library for 3D WebGL, is used to create the 3D visualization of the experimental geometries. The fundamental elements of a real-time 3D application include a scene, a camera, and a renderer as described in Figure 1. The scene functions as a container for all visible 3D objects. The camera permits viewing the scene through perspective projection. The renderer is responsible for drawing the scenes using WebGL2 into the HTML canvas element. The 3D objects are generated from meshes defined by geometric and material specifications.

Figure 1. The components of a real-time 3D application [4]

RESULTS

3D objects and geometric parameter controls

GATE/GEANT4 interprets and carries out commands from parameterized macro files (in ASCII format) to perform simulations. The macros contain key parameters defining geometry and materials of the objects. This web-based toolkit extracts geometric information from the macros to build 3D objects. Each object is constructed with an interactive data GUI to modify parameter values. The transparency of the 3D objects can be adjusted to ensure optimal visibility of the inner structures and trajectories of particles. Figure 2 shows a full geometry of a medical linear accelerator (linac) head.

Figure 2. A full geometry of a medical linear accelerator (linac) head and a water phantom.

Figure 3. A beam of light is used to simulate the cone beam for simulation designing.

Figure 4 The simulation beam (green) is displayed on top of the simulation geometry.

Beam light

In conjunction with the geometric specifications in the macro files, the program further develops primary beam indicators to aid in designing source locations and beam divergence. The divergent degree and orientation of these illuminators are denoted through boundary demarcations (spotlight helpers) as shown in Figure 3. The positional settings and divergent extents of such illuminators can likewise be tailored using an associated data GUI interface.

Simulation data encoding and visualization

The GATE/GEANT4 simulation generates trajectories (tracks) and interaction positions (steps) for primary and secondary particles, which can be saved in various file formats. The easiest approach involves storing the track and step information in an ASCII output file.

To visualize these tracks and steps in a 3D scene, data such as position, energy, particle type, and origin is extracted from the output file and converted into JSON (JavaScript Object Notation) format. The tracks are then organized based on their corresponding primary particles, and ultimately displayed as connected lines representing the interaction points along the particle trajectories.

Notably, the data GUI enables the configuration to observe the particle trajectories produced by each primary particle within specific volume regions of interest. Figure 4 shows the geometry of a simulation measuring deposited energy and dosimetry of a beam from a linac head to a water phantom.

CONCLUSIONS

This research has successfully developed 3D scenes of the geometric setups using the macro files of the GATE/GEANT4 simulation program. Additionally, the study also displays the trajectories and interaction steps of particles generated during the simulation process. These preliminary results demonstrate the feasibility of developing a graphics tool using WebGL technology to support the design and simulation of nuclear and elementary particle physics processes.

The web-based visualization tool for simulation configuration allows users to quickly explore experimental setups and reduces dependence on specific devices. Furthermore, the tool can serve as an interactive and visually descriptive medium for representing simulation results for educational and training purposes.

ACKNOWLEDGEMENTS

The present research was supported by IEEE Nuclear and Plasma Sciences Society under the project grant Nuclear Physics Virtual Laboratory (NuVirLab).

REFERENCES

[1] Sarrut D, et al. " The OpenGATE ecosystem for Monte Carlo simulation in medical physics." Phys Med Biol. 2022 September 8;67(18) doi: 10.1088/1361-6560/ac8c83

[2] Allison, John, Katsuya Amako, John Apostolakis, Pedro Arce, Makoto Asai, Tsukasa Aso, Enrico Bagli et al. "Recent developments in Geant4." Nuclear instruments and methods in physics research section A: Accelerators, Spectrometers, Detectors and Associated Equipment 835 (2016): 186-225.

[3] Danchilla, Brian, and Brian Danchilla. "Three. js framework." Beginning WebGL for HTML5 (2012): 173-203.

[4] Lewy Blue, Discover three.js, available at https://discoverthreejs.com/book/ (April 10, 2024)

24th IEEE Real Time Conference

ICISE, Quy Nhon, Vietnam, April 22-26, 2024

ID #168

Fundamental components of a 3D real-time application

Important features
- 3D objects with data GUI controls
- Additional beam light for beam design
- Simulation data encoding and visualization

This research is supported by IEEE NPSS under the project grant Nuclear Physics Virtual Laboratory (NuVirLab)