

# ELTE MSc – Modelling Laboratory

## Simulating the NEBULA detector in Geant4 – Report 4.

Balázs Pál  
Eötvös Loránd University

April 20, 2021

### Abstract

On the MSc course "*Computer Modelling Laboratory*" at ELTE, I have worked on a project in nuclear physics, where I studied the behaviour of the Japanese NEBULA detector when it was bombarded by neutron beams. For the simulation and analysis I've used the Geant4 general-purpose software, which is capable of producing state-of-the-art simulations and results in almost any field in nuclear- or particle physics.

## 1. Introduction

On the third regular meeting I have presented my plan to explore the different, built-in, so-called „physics list” in Geant4 and discover some of the differences between them in my neutron simulation. Besides working with the different physics list, I have also implemented a new data analysis method for my simulation pipeline.

## 2. Particle tracking in Geant4

Geant4 simulations can be divided into 3 different levels. The simulation will behave differently or store different informations on different levels. We can give these levels the names „step level”, the „event level” and the „run level”.

### 2.1. Step level

Simulations in Geant4 work just like almost any other computer simulations. The time is discretized and the simulation advances in smaller  $dt$  long steps, updating the parameters and quantities constantly. This is why the bottom and highest resolution level of the Geant4 simulation is this step-wise stream of information.

This is the most (and only) versatile level in tracking quantities, which won't accumulate over time and/or would change value at basically every other step. Quantities like the polarization of a particle, its charge, the magnitude of the electric/magnetic field, the physical process happening in a simulation step, etc. are all step-dependent and can't be agglomerated.

### 2.2. Event level

An „event” in Geant4 is the set of all steps taken by a user-defined particle and all the steps the presence of the particle induced. What this means is, that when we shot a particle into a simulation and it happens to create a new particle through some arbitrary

physical process, the software has to start to track the newly created particle too, through a series of simulation steps. These are appended to the set consisting of the steps of the original particle. These together form an event in Geant4.

Disadvantage of using the event level to track time evolution of quantities is that on this level we can easily handle only accumulative quantities, such as energies. We can analyse eg. how much energy a neutron lost during its lifetime by broadcasting deposited energy values from the step level and accumulating them on the event level, resulting in the total energy deposited during the full event.

### 2.3. Run level

The uppermost one is the „run level”, which encompasses every event in the simulation. The `UserRunAction` class is the direct operator of the particle generation and it conducts the simulation of the individual Geant4 events, as well it handles data I/O.

## 3. Exploration of Geant4's „physics lists”

The so-called „physics list” in Geant4 are collections of physical processes. They explicitly determine what physics Geant4 should utilize when running a simulation. They define different effects, different interactions, which lists can be customized at will to suit our simulation. Certain physical effects only relevant at lower energies, some only at higher energies. And if we don't choose appropriate physics for our simulation, it will be simply inaccurate. Of course different physics means different results, and that's why it is an important question to ask, what physical framework should be used in a novel simulation.

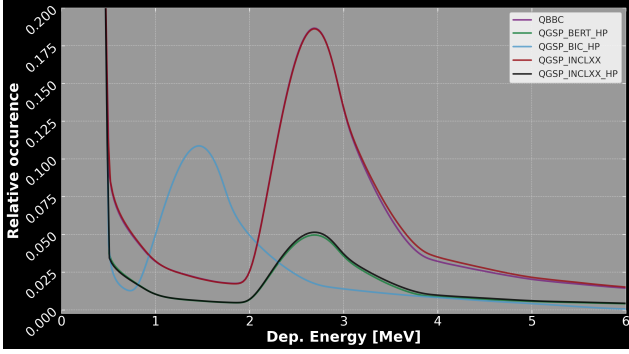


Figure 1

On Fig. 1 we can see the difference of the KDE of energy deposit for different physics lists in Geant4. It can be seen that there are obviously big differences between different physics types. One can say, that for neutrons with smaller energies, the physics list ending with the HP tag in their names should be used, which contain the so-called „data driven high precision neutron” package.

#### 4. Explored quantities

Besides the deposited energy I’ve also explored the occurrence of physical processes in the simulation. What you would see on this figure is that the three most frequent events are the energy deposit of hadron

ionization (**hIoni**), the elastic collision of hadrons (**hadElastic**) and the energy deposit of electron ionization (**eIoni**). Of course if the physics list changes, the processes also change. Some processes are dropped, new processes are introduced, proportions are changing around. The difference between two different physics lists can be seen on Fig. 2 and Fig. 3.

We can also plot the processes by their contribution to the total energy deposit. We can see, that for 100 MeV in case of this simplified NEBULA detector, the hadron ionization deposited the most energy during the simulation, with ion ionization and electron ionization following it on the second and third places. This is so robust, that this distribution was the very same for other processes too. These quantities can be seen on Fig. 4 and 5.

We can also explore the distribution of all the different particle steps during the simulation as seen on Fig. 6 and Fig. 7.

#### 5. Plans for next week

For the final lecture I’ll plan to examine the detection efficiency of the NEBULA detector for different energy levels. This is an important measure of the detector and will serve as a good and important task to end the project with.

6. Appendix: Figures

6.1. Process distribution in two different physics lists

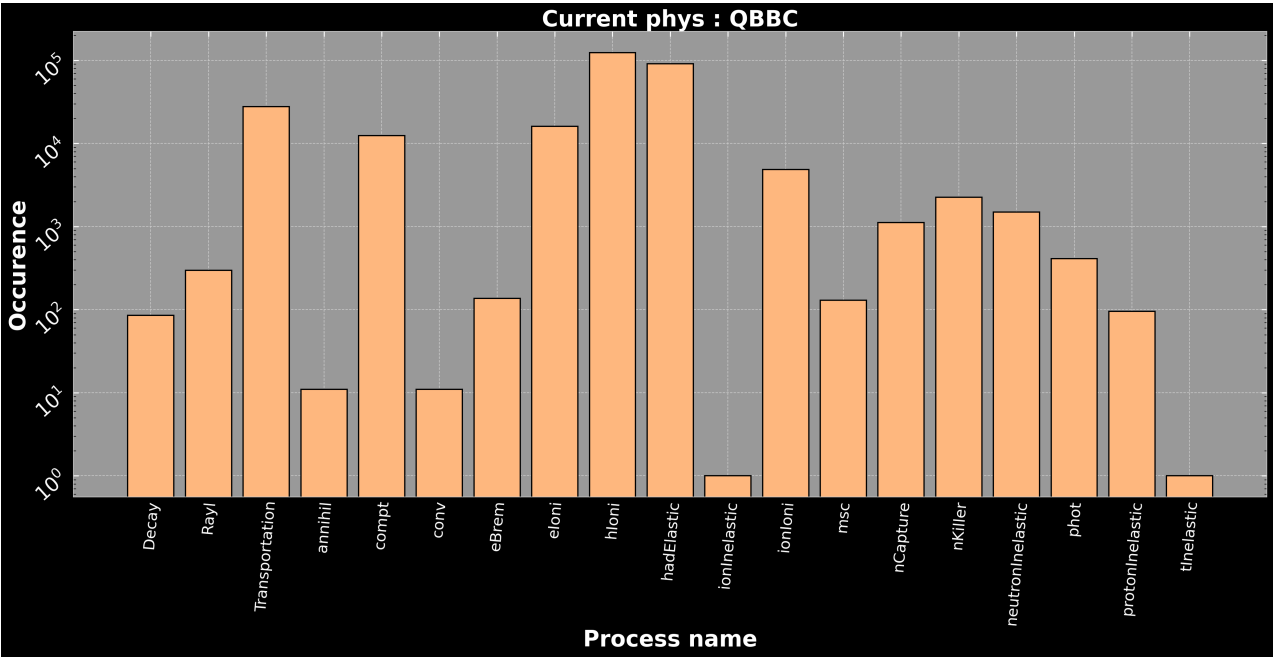


Figure 2

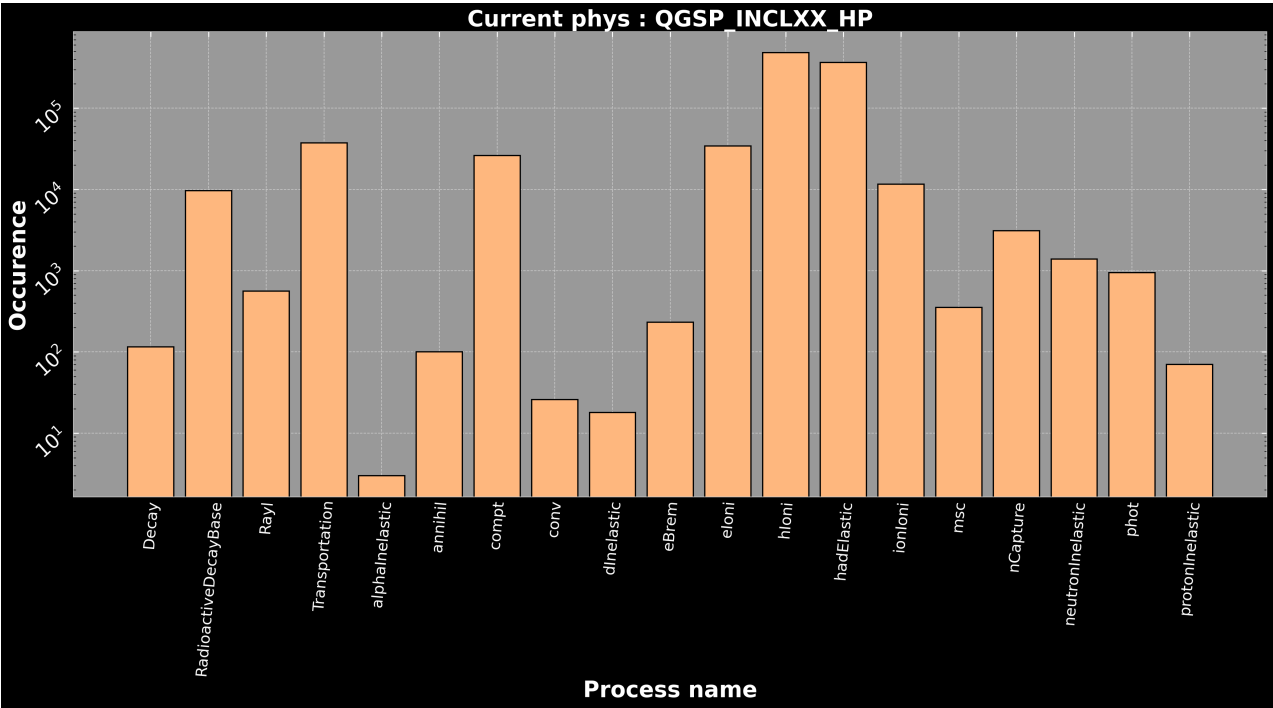


Figure 3

## 6.2. Process distribution in two different physics lists

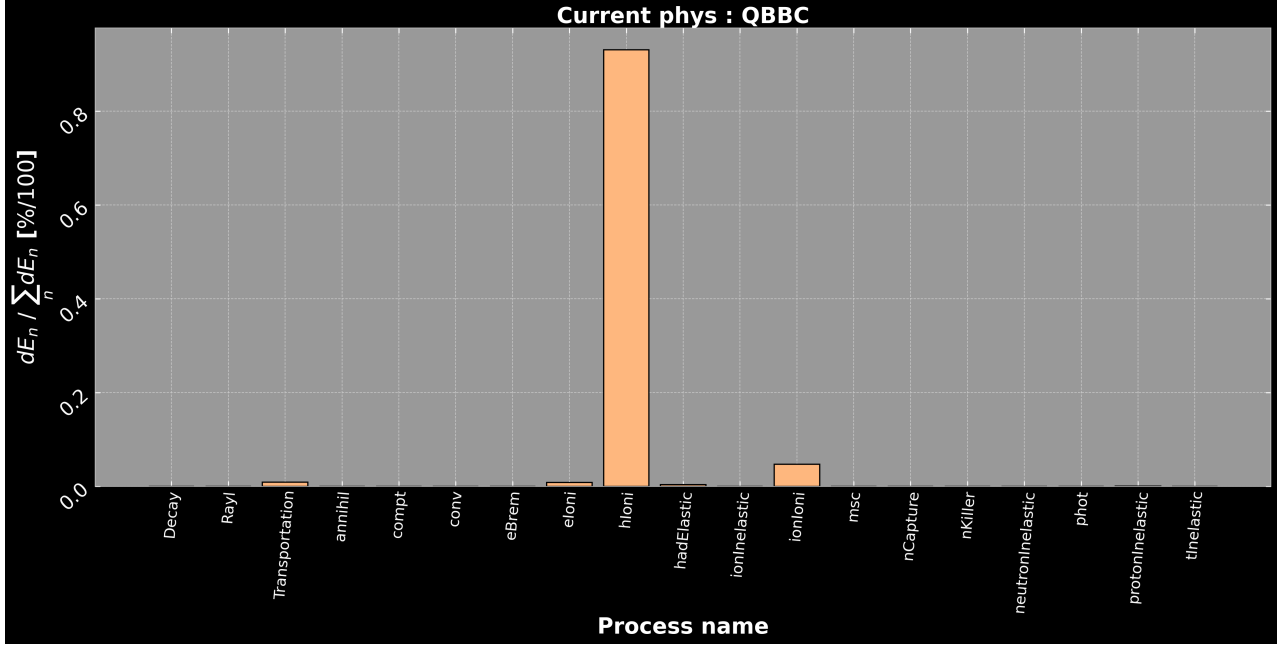


Figure 4

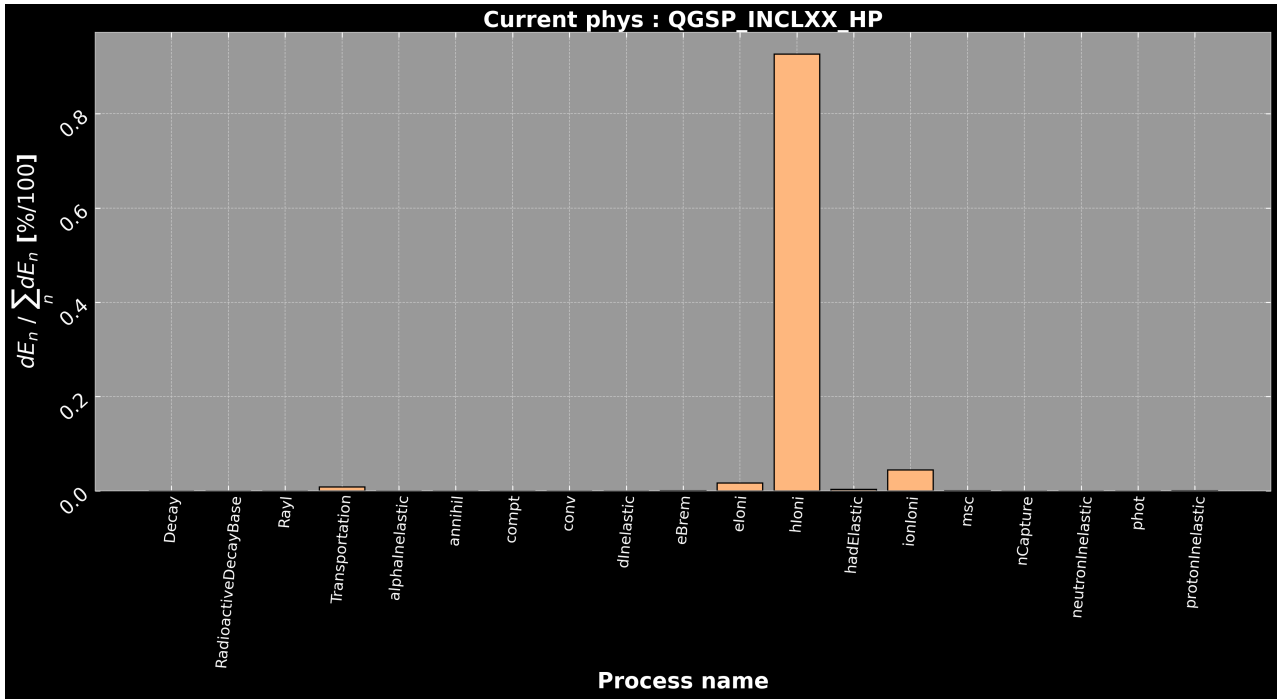


Figure 5

### 6.3. Particle steps during the simulation

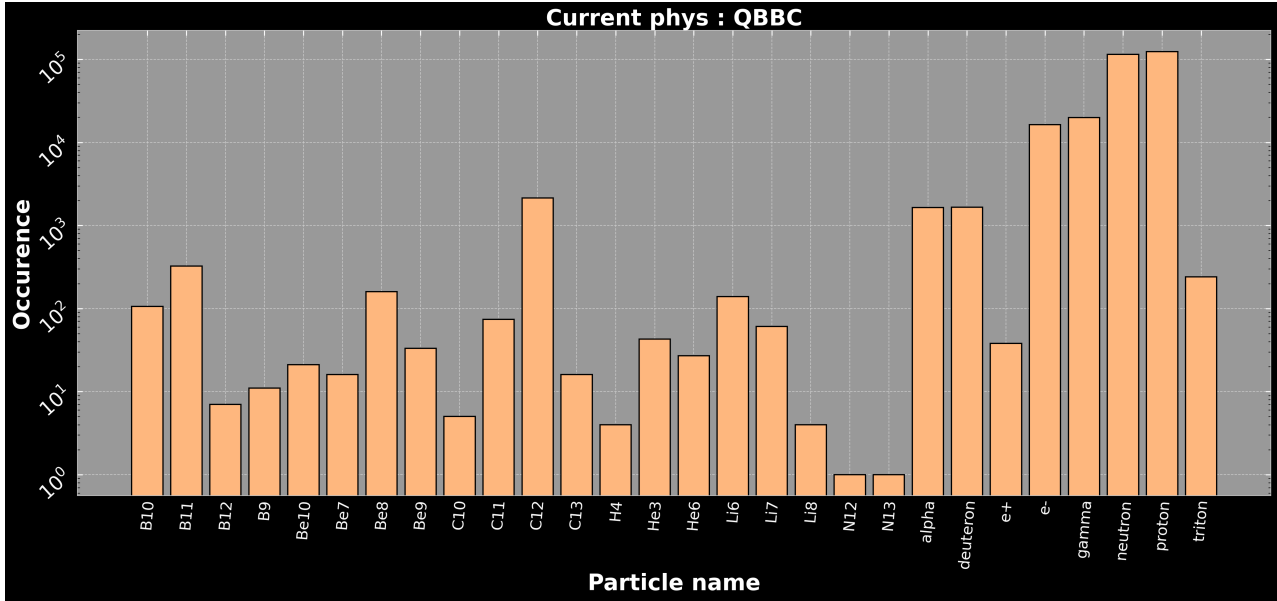


Figure 6

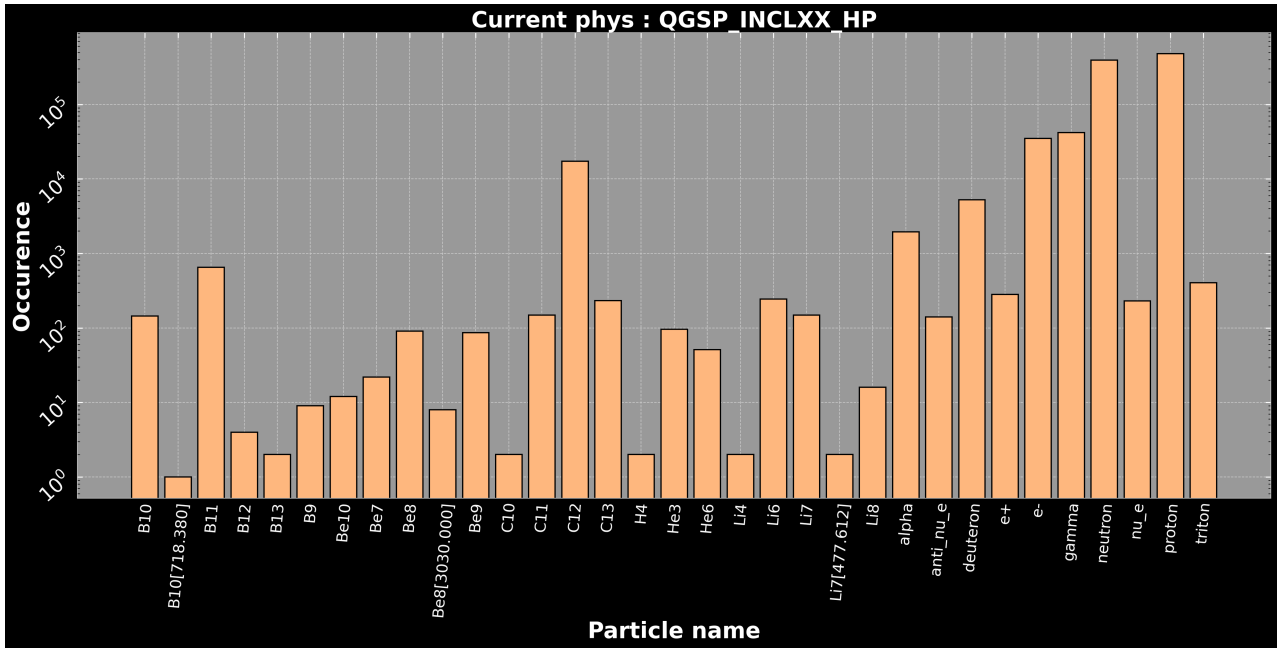


Figure 7