EPJ Nuclear Sci. Technol. **11**, 66 (2025)

https://doi.org/10.1051/epjn/2025047

EPJ Nuclear Sciences & Technologies

Available online at:
https://www.epj-n.org

Status and advances of Monte Carlo codes for particle transport simulation, edited by Andrea Zoia, Cheikh Diop, and Cyrille de Saint Jean

REGULAR ARTICLE

OPEN ACCESS

The Geant4 software toolkit evolution over the past decade

Marc Verderi*, for the Geant4 Collaboration

Laboratoire Leprince-Ringuet, CNRS/IN2P3, Ecole Polytechnique, Institut Polytechnique de Paris, Palaiseau, France

Received: 28 November 2024 / Received in final form: 21 July 2025 / Accepted: 4 August 2025

Abstract. Geant4 is an open software toolkit for the Monte-Carlo simulation of particle transport in matter. It is used in many domains, like high energy and nuclear physics, medical and space science, as well as homeland security and material science. Started 30 years ago, it is still subject to very active developments. This paper presents the developments in Geant4 over the past 10 years, covering the kernel, physics models, and software aspects. It also discusses ongoing R&D projects.

1 Introduction

At the time of the SNA+MC2013 conference, Geant4 [1–3] was entering the era of parallelism releasing its first version with Multi-Threading (MT) support, Geant4 v10.0 [4]. Since then the toolkit has evolved by many aspects with a revised parallelism scheme, refinements and extensions of the physics coverage, augmented functionalities, and investments in R&D.

This paper presents an overview of the evolution in the various areas of the toolkit and provides more details on a selected set of topics.

2 Kernel

2.1 Event parallelism

In 2013, Geant4 moved to an event[1] parallelism scheme to adapt the processing to the emerged trend at that time of architectures composed of multi- many-cores, each of them often running several threads, served by a shared memory. This move started the 10.x series of the software. This evolution was motivated by large detector simulation problems, like HEP ones, where a large but almost constant memory portion (>O(100 MB)), untouched after initialization, is used to describe the geometry and materials and to store the related cross-section tables. The sharing of this portion among the threads was the basis of the MT design, avoiding multiple copies of this same portion, as it is the case in a naive processing with independent jobs. The large memory hence saved allows the full exploitation of all cores. An MT job consists in an initialization phase performed on the master thread, followed by the independent processing of events on worker threads, and the merging of these processing onto the master.

This MT processing model was the default one from the 10.0 version (Dec. 2013) up to the 10.7 one (Dec. 2020). It has been largely adopted, in HEP and outside. In December 2021, Geant4 started the 11.x series, adopting in v11.0 a new default parallelism scheme based on "Tasking". This model exploits a dedicated lightweight task-oriented library, PTL (Parallel Tasking Library) [5] which introduces a logical layer of queued tasks, dynamically assigned to threads. This scheme conserves the advantages of the MT one but offers more flexibility through this dynamic assignment of tasks, opening the use of tasks beyond the event parallelism processing flow. The parallelization of the heavy initialization phase, before the event loop starts, is a first example: it aims at using all available threads for this operation instead of the single master one. The sub-event parallelism is another coming capability to deal with large events. The memory needed to process typical HEP events, tracks and hits, is a few O(10 MB). Large events, with many initial particles, like in high energy ion-ion collisions, can go far beyond that memory size making it difficult their event-level parallel processing. The sub-event approach consists in splitting the initial bunch of particles into several ones and to treat them as (sub-)events, hence spawning the processing of one event onto several threads. This approach can hence cope with both large geometries and large events. A last example of possible usage of this task library is the exploitation of GPU devices as back-end in hybrid computing. The offloading of the simulation of electromagnetic

* e-mail: marc.verderi@llr.in2p3.fr

[1] In here, an event denotes the initial particles to be tracked, typically resulting from a simulated collision outcome produced by an external event generator, followed by the transport of these in the geometry and recursively all their descendants, and the associated simulated outcomes such as hits and trajectories.

showers in calorimeters is currently investigated in R&D projects (see Sect. 7.3) and an approach is to store a bunch of $e^{\pm}$ and γ particles at the calorimeter entrance and to send their transport to GPU[2] once the bunch is large enough.

2.2 Geometry modeler and transport in fields

The incremental refinements and improvements of the existing geometry classes, both in terms of speed and robustness, is a continuous activity. This section underlines more the novelties.

2.2.1 Geometry modeler

During the past decade, a new library, `VecGeom` [6,7], has been proposed as possible alternative for the solid primitives, replacing partly or fully, the native Geant4 representation ones. `VecGeom` has been developed during the GeantV project [8] as part of the AIDA Unified Solid Library [9]. Its design was motivated by the vectorized parallel transport of tracks, making it natively exploiting SIMD instructions. The solid interfaces hence support vector calls, which Geant4 does not exploit. But SIMD operations are also used internally and, as such, can bring some benefit, even in scalar transport mode. This feature has notably improved the performances of tessellated solids for example, the parallel calculation on the many solid facets bringing a speedup of more than one order of magnitude compared to the initial implementation. Together with the large earlier design improvement reported in [4], such complex solids can be used in production, allowing importing advanced shapes from CAD systems using such representations. From release 10.2 in December 2015, Geant4 started to gradually interface the solids from the `VecGeom` library as an option to replace the native Geant4 solids. Most of the solids in Geant4 have now a `VecGeom` implementation. The choice is made by the user when configuring the compilation, for all available solids, or specific ones.

Several benchmarks have been carried on, with results depending much on the application, but the LHC experiments report a typical speedup in the O(1–10%) range [10, 11], an improvement good enough to make them using `VecGeom` in production. Further improvements are under way. For example, today, the native Geant4 navigation is used, regardless of the underlying solid representation. In parts fully described with `VecGeom` solids, possible speedups can however be expected if using a dedicated navigation. The benefit of this approach is being assessed. A significant on-going evolution of the `VecGeom` library is to provide a surface-based representation of the solids. Such an implementation may improve the compliance to GPU processing flow and is developed in the context of R&D projects (Sect. 7.3).

[2] An adapted geometry and physics representation is defined on the device, it sends back to the CPU hits or steps and exiting tracks.

2.2.2 Transport in fields

The transport of tracks in fields -electric, magnetic, gravitational...– is a critical operation both in terms of CPU performances and of the precision of the simulated curved trajectory. This one is divided in a set of chords used to check for possible volume boundary crossing by linear interpolation. The maximum distance between the chord and the true trajectory is controlled by two tolerance conditions. The first, applied in free space, reduces the risk that the chord misses any volume crossed by the actual trajectory. The second, more stringent, applies at volume boundary crossing: here, the track is positioned at the chord–surface intersection, which induces a slight shift inward the real curved trajectory; the specified tolerance limits this effect. Chord finding generally requires several iterations, each involving the integration of the equations of motion, making it computationally intensive. The tools responsible for this process are called the "steppers" (distinct from Geant4 "steps", as seen by users).

Over the past decade, several steppers have been developed. The classical 4th order Range-Kutta, RK4, method, implemented in the `G4ClassicalRK4` class, was the default stepper up to Geant4 version 10.3. From 10.4 (Dec. 2017) it has been superseded by the 5th order Dormand-Prince method [12], implemented in `G4DormandPrince745` class, named as such to refer to the 7 stages[3] of the method and the use of the 4th and 5th order to estimate the error. Beyond its high precision, the Dormand-Prince method requires less calculations than the RK4 one, as it computes 6[4] points instead of 11.

Fewer calculations are still possible without loss of precision. The steppers in Geant4 can be used in an interpolation technique: once the initial and final points of a trajectory segment have been determined, the state of the stepper allows it to calculate any intermediate point in a light way. This method has been implemented in the `G4InterpolationDriver` class.

Combining several steppers offers the advantage of using advanced ones for the most demanding fields, and cheaper but faster ones for slowly varying fields. The `G4BFieldIntegrationDriver`, completed in version 10.6, uses the `G4DormandPrince745` class for short steps and the `G4HelixHeum` stepper for smooth fields. If the field varies sufficiently slowly, `G4HelixHeum` can even integrate over many helical turns, still leading to a single Geant4 step.

Recent releases introduced new features. The `G4BorisDriver` aims at better controlling energy and phase space volume conservation, even when a large number of integrations are performed. A second order symplectic method is the driving concept. This development is critical for experiments like the g-2 one [13], where several thousands of turns of the muons in the ring have to be simulated with essentially exact calculation. A new

[3] A "stage" is the calculation of the field and equations of motion.

[4] The Dormand-Prince method has the so-called FSAL property (First Same As Last). There are actually 7 points needed for each chord, but the last one being the first one of the next chord, in most of the cases, only 6 points are evaluated.

integration method, based on Quantum State Simulation (QSS) [14–16], is under investigation. QSS methods rely on quantized state variables instead of quantized time like with Runge-Kutta, the quantum defining the accuracy. QSS methods are efficient in discrete problems and the goal of this study is to assess their capability in faster finding of intersection of trajectories with surfaces under the desired precision of the specified quantum.

2.2.3 Usability

The usability for the transport in field has been improved by the recently introduced user class G4TransportationParameters to give easy access by UI commands to the various parameters to control the navigation and in particular the transport in fields. These parameters are for example the tolerances on the trajectory-chord distance, the energy threshold for killing silently a particle endlessly looping in vacuum or low density medium, etc.

3 Evolution of the physics packages

The physics packages have been largely evolved during the past decade. Incremental evolution was continued, with refinements both in terms of physics and processing performances, and a number of new features have been added.

3.1 Electromagnetic physics

The electromagnetic (EM) physics package consists of several sub-packages tailored to different energy ranges or projectile types. The so-called "standard EM" module covers interactions of $e^{\pm}, \mu^{\pm}, \tau^{\pm}, \gamma$, and ionization of charged hadrons and ions above 1 keV and is suitable for HEP applications. A low energy package extends the coverage down to about 100 eV and comes in two variants: Penelope (PEN), a re-implementation of the PENELOPE [17] package in its 2008 version, and Livermore (LIV), based on Livermore data library [18]. It serves in particular medical and space applications. Below typically 100 eV, particle–molecule effects have to be taken into account to model interactions. An ultra-low energy package dedicated to DNA microdosimetry simulation (Sect. 3.1.5) provides specific processes down to a few eV and handles the post-irradiation chemical stage. Other more specific packages, for ultra-high energy muon interactions and optical and X-ray transports, are also provided.

Over the past decade, many improvements or additions were brought to the EM packages –like more efficient multiple-scattering, improved ion ionization modeling, extended optical material properties, or recently transport of charged particles in crystalline structures [19,20], application of microdosimetry DNA approach to cosmic ray in atmosphere for climate studies [21], or X-ray grazing physics for cavern background studies– together with several upgrades on the data tables used to describe cross-sections, final-states, or atomic relaxation effects.

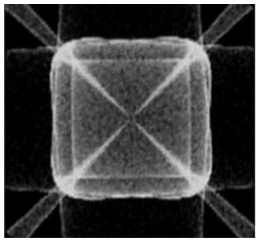

Fig. 1. Phonon caustics as obtained by Geant4 simulation. The picture is from reference [22].

In addition, special attention has been given in recent years to improve performances, identifying various areas where small speedups could be achieved. This is of importance for large scale productions, like the LHC ones.

A few topics on physics modeling and performances have been selected below, for more details.

3.1.1 Phonon transport

By the beginning of past decade, the phonon transport has been released [22] to serve ultra-low temperature condensed matter applications. The phonon polarization states are implemented as distinct particle species, each with related processes; this modeling comes together with the ability to describe the material lattice structure. Figure 1 shows the caustics formed by the phonon transport inside a simulated crystal.

This functionality is at the basis, for example, of the G4CMP [23] software package. It was initially developed for sub-Kelvin cryogenic sensors simulation, but its usage now extends to superconducting sensors for satellite missions and superconducting qubits for quantum computing.

3.1.2 Goudsmit-Saunderson model rewrite

Few years ago, a complete rewrite of the Goudsmit-Saunderson (GS) model [24], for precise multiple scattering (MSC) simulation of at low energies (<100 MeV), has been undertaken [25,26]. The use-case of high-granularity calorimeters has in particular motivated this effort. Such devices are based on an alternation of thin sensitive and absorber layers which makes their simulation quite demanding in terms of simulated MSC quality. As explained in [26], the numerous low energy e^-'s make the main contribution to the total energy deposit, and their distribution in the shower is driven by the transmission through sensitive layers and reflections from the absorber ones, phenomena in which the MSC plays a leading role.

The GS model is a fully theory-based approach of MSC. The re-implementation [25] goes however beyond the initial model of screen Rutherford differential cross-section, and includes energy loss corrections and spin relativistic effects based on Mott description. In addition, the model generates proper correlated lateral and longitudinal end point displacements, which makes it close to what the reference single scattering calculations do. In terms of speed, this re-implementation is about twice slower than the default "Urban" model [1–3], but its precision sur-

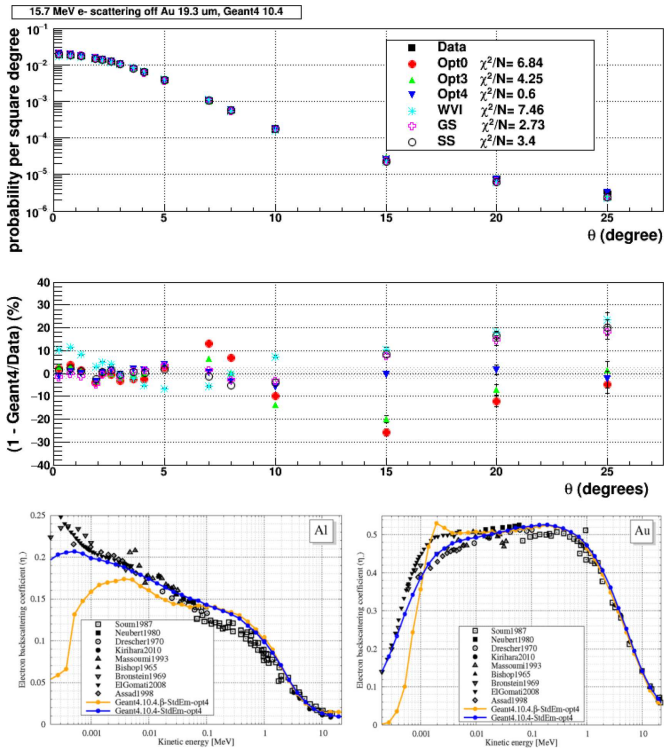

Fig. 2. Figures from reference [26]. The new `GS` revised model including the Mott correction is indicated as "Opt4" on the two top angular plots (blue triangle). The `GS` model without this correction is indicated as "`GS`" (pink open cross marker). The bottom plots show the e^- backscattering coefficient as function of energy on aluminum and gold targets. The yellow curve is obtained with the default "Urban" model, the blue one is for the new `GS` revised implementation.

passes all other models available in Geant4, as shown in Figure 2.

3.1.3 Quantum entanglement

Quantum entanglement of γ's from e^+ annihilation can be exploited to improve the signal to noise ratio in PET systems [27]. The γ entanglement has been introduced in Geant4 by authors of [27] with sharing of information between the correlated γ's: the polarization at the first Compton interaction is stored and used for the second γ at its first interaction. Their relative azimuthal angle, $\Delta\phi$, measured in a plane orthogonal to their line of flight, is used as an observable in Figure 3.

3.1.4 Approximation-free polarized γ conversion model

In the context of the cosmic γ-ray polarimetry project HARPO [28], a new approximation-free polarized γ conversion model, `G4BetheHeitler5DModel`, was produced [29]. A precise description down to the reaction threshold was the main requirement, which implies a proper sampling of the target nucleus recoil. The final state is described by a 5D differential cross-section, with the leptons polar and azimuthal angles, and their energy sharing. The model can be operated with high precision from the reaction threshold up to TeV

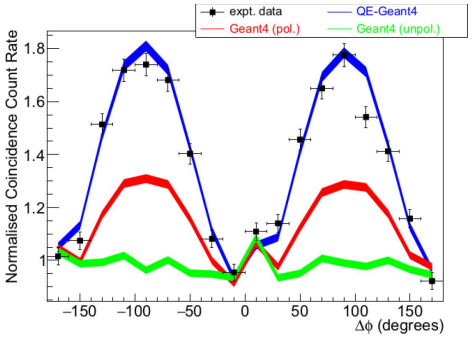

Fig. 3. Figure from reference [27] showing the distribution of the γ's $\Delta\phi$ relative azimuthal angle after their first Compton scattering. The angle is measured in the plane perpendicular to the γ's line of flight, and is relative to an axis in this plane. The experimental setup is described in [27], its segmentation allows measuring the γ's interaction points and hence to reconstruct the γ's direction after their first scattering. The unpolarized model (green) and polarized one without entanglement (red) curves are shown, together with the new model developed (blue) which is in very good accordance with the data.

scale. It was released with Geant4 v10.5 as a $\gamma \to e^+e^-$ pair conversion model, and has been extended shortly after as a $\gamma \to \mu^+\mu^-$ one. Its precision makes it a reference model in Geant4.

3.1.5 Radio-induced DNA damage simulation

The Geant4-DNA extension in Geant4 [30] can simulate damages induced directly by the physical interactions of ionizing particles with DNA, and indirectly through chemical reactions between DNA and molecules, such as radical species, produced by water radiolysis in the liquid water surrounding DNA. The physics processes are modeled down to a few eV without condensed history, ensuring maximum tracking precision [31–33].

The chemical stage presents a significant challenge, as it involves handling the mutual interactions of numerous chemical species. Over the past decade, various solutions have been developed to manage the computational cost of this stage and extend its simulation to long post-irradiation times [34].

In particular, a recently introduced mesoscopic approach bridges the gap between microscopic simulation of radiolysis for short times after irradiation (<5 ns) and the final homogeneous phase (>1 s). This approach uses a fine mesh to map the reaction volume, ensuring a uniform distribution within each voxel and simplifying the calculations. As the reactions progress over time and neighboring voxel contents become more uniform, the voxels are gradually merged, eventually reducing the system to a single voxel for the final homogeneous stage.

The DNA-scale geometries available in dedicated extended and advanced examples have largely evolved too, with a multi-scale description of biological models, as shown for example in Figure 4, containing up to several billions of DNA base pairs for a single cell nucleus [35].

This module finds a large community of users in radiobiology and medical applications [36,37].

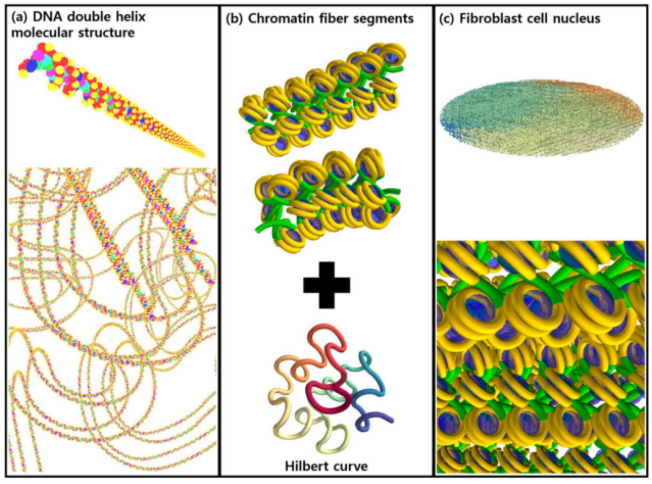

Fig. 4. Figure from reference [35] displaying an example of various levels of description of the DNA geometry that can be used in the simulation: double helix, chromatin, and fibroblast. The modeling of a full cell nucleus involves several billions of base pairs.

3.1.6 Improving performances

Improving performances is an effort that is carried out in many areas of the EM physics packages. Options were developed, like merging the MSC with the transportation (the process caring about geometry boundaries) [38] to allow charged particles to make bigger steps, hence leading to fewer of them.

Another option is the merging of the γ processes into a single one, `G4GammaGeneralProcess`. This improves the simulation speed by a few percents (see Sect. 6), as less calculations are needed and fewer communications with the tracking logic are made. The `G4GammaGeneralProcess` holds the total γ cross-section, and, on the basis of the relative sub-processes (conversion, Compton, etc.) cross-section, makes a random selection of the one to be applied.

This merging of processes has also been taken as an opportunity to introduce the Woodcock tracking acceleration technique [39]. As the `G4GammaGeneralProcess` holds the total cross-section, it makes simple to apply this technique to all sub-processes at once. When used for the ATLAS calorimeter, a sizable boost of more than 15% (see Sect. 6) is obtained.

3.1.7 Usability

The usability of the electromagnetic physics package has been improved by making the various parameters controlling the simulation directly accessible to users through the `G4EmParameters` class. More accurate modeling, like DNA physics, energy loss by photo-absorption ionization, or alternative ionization fluctuation, can be chosen by simple command line selection of these modeling, in specific geometry parts.

More information in utility classes on the step, like the NIEL for example, is also provided to users.

3.2 Evolution in hadronic physics

The hadronic physics package models interactions as elastic and inelastic processes (with neutrons treated separately for more details). Each process uses one or a set of inclusive cross-sections to describe interaction probabilities and relies on various models to sample the final state. These models cover different, usually overlapping, energy ranges, with transitions handled via energy-based random selection. In addition, in each energy range, several models, with respective advantages and weaknesses, are usually available. Inelastic hadronic final-state models can be categorized by energy as:

- high energy ($> \sim$3 GeV): with the string-based models Fritiof (`FTF`) and Quark-Gluon-String (`QGS`).
- Intermediate energy ($\sim$100 MeV – $\sim$6 GeV): with intra-nuclear cascade models Bertini (`BERT`), Binary (`BIC`), and the Liège cascade `INCLXX`.
- Pre-equilibrium ($\sim$10 MeV – $\sim$100 MeV): with Precompound (`P`) and `ABLA`.
- De-excitation ($< \sim$10 MeV): with the cooperative set of models Fermi-break-up, multi-fragmentation, fission, and photon evaporation model or alternative `ABLA`.

3.2.1 High-energy models

The high-energy models describe the collisions by means of strings formation, which dissipate their energy via successive fragmentations. The products resulting from these strings hadronization may be reinserted into the nucleus for possible re-interaction. The release of excess energy from the remaining wounded nucleus is simulated through a pre-equilibrium stage followed by de-excitation processes, such as evaporation, Fermi break-up, or others mechanisms.

The `FTF` model [40] implements the Fritiof [41,42] semi-phenomenological model which treats hadron-hadron interactions as binary reactions $h_1 + h_2 \rightarrow h_1^{(\prime)} + h_2^{(\prime)}$ where one or two of the initial hadrons, h_1 or h_2, can be excited during the collision. Excited hadrons are treated as strings which are handled by the Lund model [43] which simulates string fragmentation. Incremental improvements of the Lund model have consisted in a better di-quark fragmentation into baryons, and, based on the NA61/SHINE data [44], in a revised production of strange mesons and baryons in proton-proton interactions, revised mixing probability between the ρ^0 and ω vector mesons, revised probabilities ratios between pseudo-scalar and vector meson production. The fragmentation process has been further enriched by incorporating rotating strings [45,46] to address the lack of transverse momentum of the new strings generated during fragmentation phase.

The Quark Gluon String model (`QGS`) is based on considerations by Kaidalov [47] (dual topological unitarisation, reggeon phenomenology). Compared to `FTF`, it is more theory-based, closer to QCD, making it capable to cover higher energies than `FTF`.

By the beginning of the past decade, the Geant4 `QGS` implementation was revised in depth, as it was not conforming enough to Kaidalov's ideas. In particular, the

fragmentation was rewritten, as being based on fragmentation functions from annihilation or deep inelastic scattering, which is not justified in soft processes, and inconsistent with Reggeon theory. This led to a better agreement with data and FTF description, with wider, longer and lower energy response for hadronic showers. A series of incremental improvements have followed: implementation of the Reggeon cascading and Fermi motion, revised formation of the residual nucleus, improved kaon-nucleon and γ-nucleon cross-sections, new algorithm for the decay of the last string inspired by the Lund model. A final tuning of the string formation and fragmentation parameters led to a better agreement with thin target data.

As above with FTF, the NA61/SHINE data [44] have been used to improve QGS on the same topics.

These evolutions make QGS competitive against FTF above O(15 GeV), while FTF offers better performances below.

An optional coalescence capability, designed for use with both string models, has been recently developed to address the specific needs of cosmic ray applications. This functionality utilizes the list of secondaries generated by the string model to form deuterons and antideuterons from proton-neutron and antiproton-antineutron pairs, respectively, provided their momenta are sufficiently close. This option can be activated interactively.

The ABLA de-excitation module was previously only used coupled to the INCLXX cascade model, but recently it became a de-excitation option, still at an experimental stage at this time, for the FTF and QGS (which use Precompound as default) string models.

3.2.2 Extension of high-energy models to charm & bottom hadrons and to hyper-nuclei

In the past three years an important extension has been the addition of the treatment of two sets of projectiles: charm and bottom hadrons, and light hyper-nuclei.

Charm and bottom hadrons are short lived particles, but at the LHC they can receive enough boost to travel up to the first layers of silicon trackers and possibly interact there. The need for hadronic interaction of these particles in matter has been expressed by the ATLAS and LHCb collaborations, but this capability is even more of interest to studies conducted for the FCC-hh facility, given the higher energy involved.

The particle species concerned are the D's, D_c's, B's, B_s's, B_c's, (anti-)Λ_c^+, (anti-)Ξ_c's, (anti-)Ω_c^0, (anti-)Λ_b^0, (anti-)Ξ_b's, (anti-)Ω_b^+, were neutral and charged species are assumed when not specified. Ionization for the charged particles has been provided first and has been followed the simulation of the nuclear interactions. Elastic and inelastic hadronic processes are provided, their cross-sections being based on the Glauber-Gribov approach. For elastic scattering, the final state is simulated using a basic parameterized approach (GHEISHA [48] (Gamma-Hadron- Electron- Interaction SH(A)ower)). For inelastic collisions, the high-energy use-case allows focusing, in a first approach, on the FTF and QGS models; the low energy limit is however pushed down to 100 MeV. These two models have been extended to deal with c and b quarks and di-quarks carried by the projectile, and to make it possible the creation of $c\bar{c}$ and $b\bar{b}$ pairs from vacuum during the string fragmentation phase. The decay of heavy hadrons produced in such collisions is today handled by a simple phase-space multi-body model for a single decay channel but the ability to couple to an event generator for improved realism is considered for future improvements. Cascade models, for intermediate energies, will be dealt with later on; which means that today hadronic interactions of charm and bottom hadrons below 100 MeV are not yet simulated.

The development for hadronic interactions of light hyper-nuclei, together with their anti-particles, is motivated by the ALICE experiment which is interested in cross-section measurements below 15 GeV. The light hyper-nuclei concerned are hyper-triton $(p-n-\Lambda^0)$, hyper-hydrogen-4 $(p-n-n-\Lambda^0)$, hyper-helium-4 $(p-p-n-\Lambda^0)$, double-hyper-hydrogen-4 $(p-n-\Lambda^0-\Lambda^0)$, double-hyper-double-neutron $(n-n-\Lambda^0-\Lambda^0)$, hyper-helium-5 $(p-p-n-n-\Lambda^0)$, and their anti-particles. Before this development, the hyper-nuclei were simulated as usual nuclei, which is a too simple approximation for ALICE needs.

Similarly to the treatment of c and b hadrons above, a same 100 MeV workaround is used, the cross-section is based on Glauber-Gribov, and the elastic final state is using the GHEISHA [48] parameterized model.

For inelastic collisions, FTF and INCLXX models were already capable of treating the particle species. In a first stage, FTF was augmented to deal with the antiparticles case, INCLXX will be adapted later on. The FTF model has hence been extended to treating the anti-hyper-nuclei. In case an excited hyper-fragment is produced, it is decayed promptly with γ emission. The model also handles annihilation of the anti-hyper-nuclei, and applies also at rest, which is possible because of the several GeV in the center of mass of the reaction involved, which is high enough for FTF.

3.2.3 Cascade models

Intra-nuclear cascade models are efficient at intermediate energies, where the projectile de Broglie wavelength is of the order of nucleon size. The collision can hence be modeled to first order by nucleon-nucleon interactions, with fragments which can themselves interact inside the same nucleus, hence the cascade. The models distinguish themselves by the degree of realism they consider to describe the nucleus structure and nuclear potential. The Bertini cascade, BERT, is a classical model which does not take into account quantum mechanics effects, the binary cascade, BIC, is a time-dependent model which considers resonances during the interaction and decay them according to their quantum numbers, and INCLXX includes in addition a realistic nuclear potential and Pauli blocking. The higher realism is of course at the higher CPU cost.

The INCLXX model [49,50] has been improved the most in the past decade. Its energy validity has been extended up to O(15–20 GeV) by opening multi-pion channels in the final state. Other particles can be produced then and the model has been augmented with η and ω mesons

production, and species in the strangeness sector –kaon and hyperons– that the model supports both as projectile or as produced particles. A major and recent extension has been the handling of antiproton annihilation at rest and in-flight. On this topic, `INCLXX` is an alternative to `FTF`. When both models are used, `FTF` and `INCLXX` handle annihilation in-flight and at-rest, respectively.

Very recently, the interfaces of `BERT` and `BIC` have been evolved to accommodate `ABLA` as de-excitation module. The usage of `ABLA` with `INCLXX` leads to very good physics performances in spallation simulation; the estimation of these performances with `BERT` and `BIC` is ongoing.

3.2.4 Pre-equilibrium & de-excitation models, radioactive decay

The pre-equilibrium corresponds to the phase where the wounded nucleus are de-excited to be brought to a state where it has no memory of its collision history. The de-excitation phase takes over to bring it to its ground level or long-lifetime excited state. The numerous and low energy fragments or particles produced by these modules play an important role for what the simulation of detector response is concerned, they hence play a critical role in the simulation, in particular in calorimeters.

Several efforts have been made at the beginning of the past decade to streamline these various modules, written at the beginning of Geant4. The use of nuclear data has been made consistent across them (and with the radioactive decay module), the same applies to other physics calculations, in particular cross-section.

The main incremental improvements are as follows. The nucleons and γ's evaporation models were revised leading, respectively, to an increase of the visible energy in low Z calorimeters and the narrowing of showers, especially for heavy absorbers. The built-in Bertini evaporation was revised too, correcting the overproduction of low-energy neutrons and protons.

The Radioactive Decay module has grown in richness over time, and is now handling several thousand of decay channels. In the past decade, it was equipped with the `G4UAtomicDeexcitation` module for handling atomic relaxation effects like fluorescence and Auger e^- emission following decays with a nucleus charge change. This atomic de-excitation became activated by default later on. Additionally, the code was reviewed to consolidate it into a single table of radioactive decay channels and was streamlined to ensure thread safety.

3.2.5 Low energy neutrons

Recently, the low-energy neutron module (<20 MeV) in Geant4 has benefited from new expertise, leading to rapid improvements in both its physics accuracy and performance. A comprehensive revision of the software was undertaken, accompanied by significant upgrades. These efforts are detailed in [51–53]. In [51], the treatment of epithermal neutrons (energies ranging from a few eV to a few hundred keV) is discussed, with particular attention given to the proper handling of relative velocities in the free-gas approximation for materials. This includes accounting for Doppler broadening using the "Sampling of the Velocity of the Target nucleus" (SVT) algorithm [54]. In [52], the case of heavy nuclei, where large cross-section variations occur in the epithermal region, is addressed with the Doppler Broadening Rejection Correction [55,56], which overcomes the limitations of the SVT algorithm in such cases. Figure 5 illustrates the effect of these upgrades imported in Geant4.

Reference [56] addresses the Unresolved Resonance Region (URR) problem, in which the resonances are so close that they are not accessible experimentally or may even overlap. URR is subject to strong self-shielding effect [56]. To treat this effect, a Probability Table approach was introduced, in replacement of the naive smooth cross-section used previously. Large enhancements of neutron flux by about a factor 3 are obtained for example on tungsten target.

With these upgrades, the description of Doppler-broadened elastic scattering in Geant4 and URR treatment have reached a quality comparable to reference codes such as TRIPOLI-4® [57].

3.2.6 Usability

The class `G4HadronicParameters` was introduced in release 10.5 (Dec. 2018) which allows various parameters to be accessible by interactive commands. The transition region between the string models and the cascade ones can be controlled this way. The class offers the ability to scale cross-sections in order to estimate the effect of their uncertainties on the systematic errors.

The `G4FTFTunings` class is another usability class that has been introduced to expose the parameters used by the `FTF` string model and to allow expert users to change them interactively.

4 Usability & examples

This section summarizes the evolution of functionalities which are close to daily work of users.

4.1 Physics constructors and physics lists

The "physics list" is a key concept in Geant4. A physics list is a consistent set of processes (and their possible underneath models) and particle types. A physics list defines the physics setup of the application; it is for example there that the various energy ranges of models are defined. The "modular physics list" is a subsequent concept which allows defining physics lists by composing them by modules, called physics constructors. A physics constructor is similar to a physics list, but focuses on a domain, for example "standard electromagnetic physics". A modular physics list like `FTFP_BERT` is hence made of the physics constructor `FTFP` for high energy hadronic interactions (`FTF` string model and Pre-compound for the wounded nucleus), `BERT` for intermediate energies, and "standard" electromagnetic physics, as no electromagnetic physics constructor is specified. Geant4 provides a set of

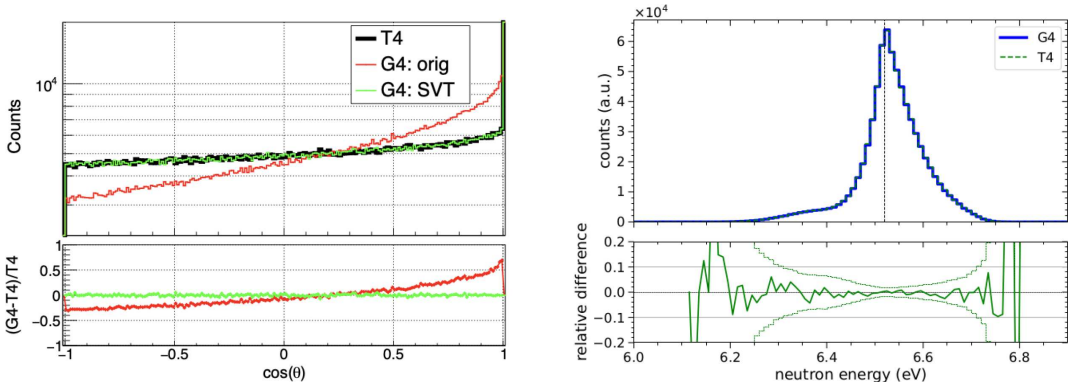

Fig. 5. Figures from references [51], left, and [52], right. The left figure shows the effect of taking into account the Doppler broadening with the SVT technique on the scattering angle of the neutron on a 12C free gas target; the upgraded version of Geant4 becomes in excellent agreement with the reference code TRIPOLI-4® [57]. The right picture shows the energy distribution after scattering on a heavy 238U target at 300°K of 6.52 eV incident energy neutrons. In here, Geant4 has been upgraded with the DBRC technique; a very good agreement is shown with TRIPOLI-4®.

"pre-packaged" physics list [58] which cover frequent use-cases, and which are regularly monitored for both physics and computing performances.

A new process or an underneath new model, developed in the physics packages, gets reflected into an existing or a new physics constructors and/or modular physics lists, to make it usable. It means that physics lists don't have a frozen content: they evolve with their underneath processes and models, and may be augmented with new ones.

Over the past decade, many changes and new features have hence appeared in the physics constructors and physics lists. In the EM domain, and for topics which have been presented in this paper, new physics lists making use of the Goudsmit-Saunderson re-implementation have been created, they are labeled with a `_GS` suffix. The EM constructor `G4EmStandardPhysics_option4` is meant to collect the best Geant4 can provide in terms of EM physics quality; it hence uses Goudsmit-Saunderson MSC model and `G4BetheHeitler5DModel` conversion model. The DNA models are combined too in dedicated physics constructors and physics lists. In order to combine from high to very low energy models, the DNA physics lists can now use also standard EM models, with a default 300 MeV transition energy.

The hadronic parts of physics lists have been subject to many evolutions. We can mention the parameterized cross-section which are getting deprecated over time, with a trend for preferring Glauber-Gribov or Barashenkov-Glauber-Gribov ones. While the high and intermediate energy models evolve, the random transition regions (Sect. 3.2) between them are adapted; this is based on physical considerations and on practical ones, to avoid generating, for example, discontinuities in the observables. Some specific physics lists are also produced. The `FTFP_BERT_ATL` one has dedicated tunes (Sect. 3.1.7) of the `FTF` model to better describe the pion showers as measured on ATLAS calorimeter test beams. The extension to charm & bottom hadrons and hyper-nuclei (Sect. 3.2.2) has led to new physics constructors, which are active and inactive, by default, respectively.

4.2 Scoring, analysis and visualization

The `analysis` module of Geant4 was created at the time of the move to Multi-Threading to support filling in parallel histograms, profiles and trees. Various formats (ROOT, hbook, cvs, etc.) were made available, and storage per thread or global to the run was provided as an option. The popular HDF5 format has been added later on as a storage capability. The `analysis` module was further revised in depth to introduce the flexibility of supporting multiple files and multiple storage formats in a same run, these being configurable interactively. The ability to save multiple times the "same" analysis objects was added, typically to monitor on some observables the effect of changing run conditions. Recently, the set of UI commands to create all types of analysis objects interactively was provided.

The trend with developments of scoring capabilities in Geant4 is to make them more and more UI-based, fully configurable interactively. Over the past decade, a lightweight statistical analysis class, `G4StatAnalysis`, that calculates mean, FOM, relative error, standard deviation, etc. has been provided and made easy to be used within the frame of hits, hit maps, etc. The scoring has been bridged with the `analysis` capabilities of Geant4 with the `G4VScoreNtupleWriter` interface class and derived `G4TScoreNtupleWriter` one, which both come with their related set of UI commands. These tools capture the user desired quantities and fill the `analysis` trees defined by the user. The same functionality has been provided for histograms, with the `G4VScoreHistFiller` interface class and `G4TScoreHistFiller` derive one.

Interactive definition of the mesh when using a parallel geometry has been implemented, with a variety of choices for defining this mesh.

The visualization of Geant4 system relies on the concept of visualization drivers (OpenGL, Qt, Open Inventor, Qt3D, etc.) that provide the visualization environment; the Qt system, thanks to its cross-platform capability, became rather popular over the past years. The MT adoption at the beginning of the decade has required some design iterations on the way events in parallel could be

visualized. A dedicated thread, collecting the events, has then been implemented. Various functionalities have been added: coloring of detector elements based on materials density, ray-tracing to visualize geometries placed in parallel worlds on top of the mass geometry, ability to create movies by a "fly-through" the scene. A fast rendering of complex mesh has been developed, as needed for the case of human phantom visualization for example. The panel of supported drivers has been augmented too, with Qt3D and Vtk, and the ability to bypass the native graphics systems in interactive 3D session has been recently released.

4.3 Examples

The Geant4 distribution provides numerous examples, split in three categories:

- basic: to initiate to Geant4, from new user's level, up to numerous demonstration of features, with 5 main examples (and several sub-examples),
- extended: to focus on more specialized aspects, for example fields, electromagnetic physics, tracking in field, biasing, etc., with more than 180 examples in 20 categories (and sub-categories),
- advanced: which is a collection of simplified users' applications, demonstrating real usages of Geant4, with more than 30 examples.

Almost all new features which may regard users' application are demonstrated in one or several examples in the extended example category. In particular, all examples have been migrated to multithreaded mode. Over the past decade, about 80 new examples have been created (see for example Fig. 6), and many existing ones have been augmented for the demonstration of new features. Analysis, exotic physics, generic biasing, `VecGeom` features, fields, hadronic (model per region, CERN FLUKA coupling to Geant4, fragments handling, etc.), EM (5D generator demonstration, Goudsmit-Saunderson, etc.), DNA, or creation of movies with the visualization, etc. are few examples where new demonstration code has been added.

In the advanced category, 13 examples have been created. They demonstrate very diverse use-cases: DNA-repair, aerosol (simulate clouds of billions of randomly-positioned solid or liquid droplets), high granularity test-beam calorimeter, use of ICRP110 and ICRP145 human phantoms, xray detectors, ...

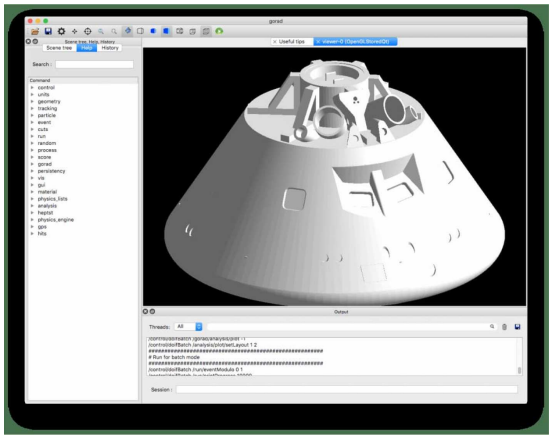

Fig. 6. Figure illustrating in the Qt environment the Gorad (Geant4 Open-source Radiation Analysis and Design) advanced example, co-developed with the NASA. Gorad is developed as a turn-key application for radiation analysis and spacecraft design. The geometry should be provided in the form of GDML. The application is controlled by UI commands, it uses Qt in interactive mode, and it can run in batch mode with an input macro file.

5 Software infrastructure evolution

5.1 Evolution the development process

5.1.1 Adoption of GitLab

The GitLab [59] software development platform has been adopted by the Geant4 Collaboration at the end of year 2019; SVN was used before. The Geant4 GitLab instance handles the `geant4-dev` repository, which is private to the collaboration.

The development workflow consists in a main branch, the "master" one, which is daily fed with development commits. It is regularly tagged, with internal "reference tags", and some versions are promoted as public releases. The feeding of the master branch is made relying on the concept of "topical branch": developers create one or several such private branches from, typically, the last reference tag on the master branch, to develop new functionalities[5]. When a development is mature enough, the developer submits a "Merge Request" (MR) in order to publish this work onto the master branch. This process is subject to series of testing and approval, as described below.

In addition to hosting the incremental developments, the `geant4-dev` repository is also an "experimental" area for developers where they can evaluate models or approaches, and possibly withdraw them.

Overall, about 1000 MRs are submitted per year.

In addition to the Geant4 download page, a public GitHub instance is used to distribute the software. Public releases and patches are published there, and users can submit pull requests (PRs). Around 10 PRs are submitted annually, reviewed by Geant4 developers, and manually ported to `geant4-dev` if deemed valid, with authorship properly attributed.

5.1.2 CMake & CDash

At the time of SNA+MC2013 conference, the Geant4 Collaboration was adopting the CMake and CTest/CDash software tools. Over the past decade, CMake has demonstrated to largely facilitate the software installation to users; its native cross-platform capability has shown to be a clear asset. The variety of build options has been

[5] Branches can be created from other references, which is typically the case when producing a patch to a given release.

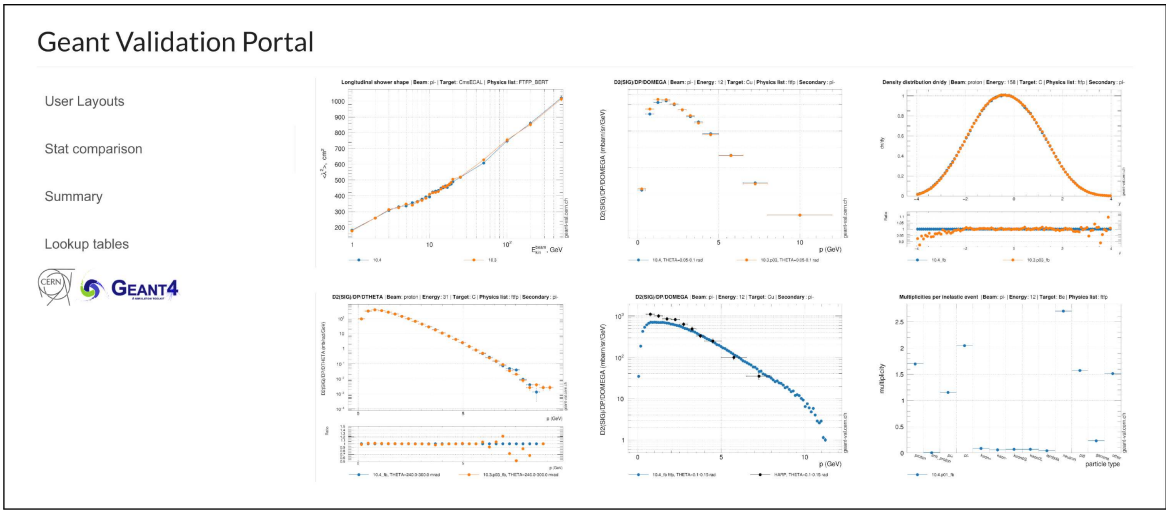

Fig. 7. An illustration of the `geant-val` portal front page, the Geant4 validation tool.

augmented over time (eg: option for automated download of process cross-sections and final states data).

The adoption of CMake led to using the CTest and CDash as testing tools for CD/CI. Automatic Continuous and shifter-triggered Nightly testing were created at that time. Initially configured for running on top of SVN, the system has been evolved to be triggered by GitLab. A "robot" oversees the developers' MRs, performs minimal checks, and automatically triggers the Continuous. Developers have their tests identified and can iterate for fixes if needed. A weekly shifter has, among tasks, the responsibility to approve the MR that will be processed by CDash during the Nightly.

There are O(20) build setups for the Continuous and O(70) for the Nightly. A build is a combination of an operating system (Linux, Mac, Windows, on different versions) compiler version, together with several processing options (use of `VecGeom`, maximum number of threads, sequential mode, check for floating point exception, etc.).

The Continuous runs a set of O(100) tests for each build on top of the current master version. The Nightly processes O(500–600) tests per build, this coverage being regularly augmented (it consisted of O(200) tests at the time of the SNA+MC2013 conference [4]). CDash then reports the success and possible failures. If a MR passes all tests, it is approved and merged to the master; it is rejected otherwise, for further iteration.

5.2 Evolution of the validation tools

The key activity of validation is continuously evolving to augment the coverage of physics validations. In contrast with the tests discussed in the previous section, the analysis of validation results cannot be fully automated as they require expertise to assess that the observed changes are consistent with the physics features introduced in the releases. Dedicated tools are needed to accommodate the growing volume and variety of these validation tests.

The Geant4 validation used to rely on the suite of unit tests and single model validation made at the developers level, followed by monthly regression tests performed with large applications (eg.: LHC calorimeters), the results of these being published on various pages.

A first effort was made to manage the tests and centralize the results, leading to the creation of the DoSSiER (Database of Scientific Simulation and Experimental Results) [60] tool, co-developed by FNAL and CERN. DoSSiER is a central database for collecting and organizing regression test results and the experimental data used for these validation tests. It comes with a web portal to query and download the validation data. Geant4 made use of it, as well as the GeantV R&D project and the GENIE collaboration [61].

In parallel, an initiative, `geant-val`, started at CERN by 2013 to initially cover the needs of local hadronic physics developers in a rapid turn-over. The tool hosts in particular the thin target data which are continuously used to verify that the developments indeed improve the physics description. Thick target data are also hosted and used for validation by checking if the observed improvements on thin targets also translate to these data. Over time, `geant-val` became the new validation tool, expanding to other validation areas, like the medical, with a large series of validations.

Still evolving, `geant-val` handles today the validation suite, a python framework for job setup, submission and analysis, and a web application for uploading, storing, and comparing validation with data. The `geant-val` portal page is shown on Figure 7.

6 Performances evolution

Geant4 performances of reference tags (including public releases) are monitored regularly at FNAL using a set of benchmark codes (simplified calorimeter, CMS experiment, etc.). Results are displayed on the portal `https://g4cpt.fnal.gov/`.

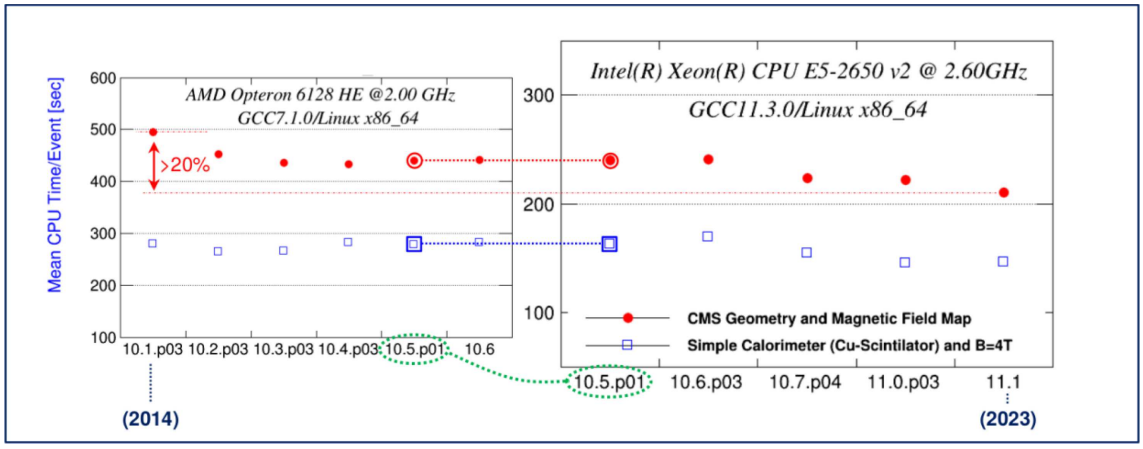

Fig. 8. Illustration of the intrinsic Geant4 performance evolution over almost the past decade. These performance tests were run on two different machines; the vertical axes are adjusted to allow the relative performance to be compared.

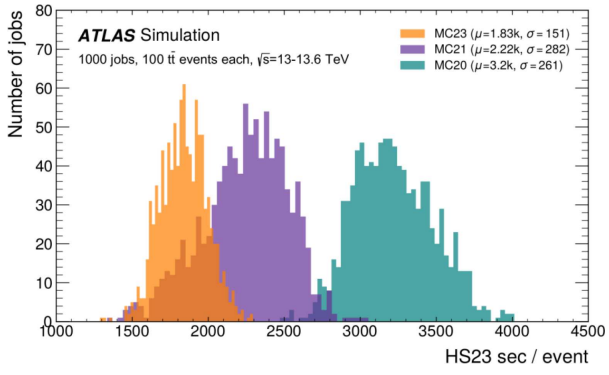

Fig. 9. Illustration of the performance gains achieved by the ATLAS collaboration through an in-depth code analysis. The figure shows the processing time distributions with three ATLAS simulation software versions. This figure is extracted from reference [10]. The large speedups are discussed in the text.

Figure 8 illustrates the performance evolution over the past decade. Two series of performances obtained on two different machines are scaled on a pair of points obtained from a same release & patch number (10.5.p01). The drop of scale from the left to the right plot is due the change of machine. The most recent release performance of these plots (11.1) can be traced back to the first one (10.1) showing that an intrinsic better than 20% speedup has been obtained over the past decade. It results from the many improvements brought to the code, with some of them being reported in this paper.

The Geant4 collaboration is committed to improve the toolkit processing speed, but performances depend also significantly on users' applications. The ATLAS experiment has for example undertaken an in depth revision of their simulation software, this effort being made in contact with Geant4 experts.

Figure 9 shows the distribution of the time needed to process one event for three software versions of the ATLAS simulation. A large speedup by about a factor 1.8 is obtained between the MC20 and MC23 processing. Details can be found in reference [10]; we underline here the main sources of acceleration reported by ATLAS:

- from MC20 to MC21:
  - revised production threshold in the electromagnetic physics : ∼6%
  - application of Russian roulette to γ's and neutrons : ∼10%
  - build of Geant4 as a single library: ∼5–7%
  - use of `G4GammaGeneralProcess`: ∼3%
  - magnetic field optimization: ∼3%
  - optimized electromagnetic end-cap geometry: ∼5–6%.
- From MC21 to MC23:
  - usage of `VecGeom` polycones, tubes, and cones : ∼2–7%
  - Woodcock tracking in the electromagnetic calorimeter end-cap : 17.5%.

It is important to note that above speed improvements are obtained at no cost or nearly no cost on the physics quality. We note that "classical" techniques, like Russian Roulette or Woodcock tracking, bring large speedups.

7 R&D activities

The R&D activities mentioned here have a focus on exploiting new techniques, new technologies or new ideas to speed up the simulation.

The HL-LHC era will be very demanding in terms of simulation statistics. Various estimates [62–64] show that the typical target to be considered is a factor 10 in throughput obtained at nearly constant resources. As outlined in Section 6, incremental speedups will continue on both the Geant4 side and experiments one, but these will not be sufficient a priori to reach the desired speedup level. Ongoing R&D efforts aim at bridging the remaining gap.

An important remark is that improving the throughput by large factors is not solely a technological challenge: it is tightly bound to improving the physics quality

too. Indeed, the precision of detector simulation is limited by two sources: the statistical uncertainty due to the finite amount of simulated data –which requires the higher throughput– and the systematic uncertainties from the physics modeling. If the amount of simulated data is augmented by a factor F it is then desirable to improve the physics description so that the systematic uncertainty due to modeling decreases by a typical $\sqrt{F}$ factor. The intrinsic physics modeling description should hence be improved, which is quite the case for hadronic physics, but is also the case in electromagnetic processes, where, for example, going beyond the per mille precision level in shower description requires going to the next QED leading order, including rare phenomena, like the 3-γ e^+ annihilation in fly.

Most of the items presented here are under development, and report on performances will come later.

7.1 AI-accelerated simulation

7.1.1 Fast simulation of electromagnetic showers

Generative AI techniques are promising for what fast simulation of showers in calorimeters is concerned. Compared to classical techniques, they may reach higher degrees of realism. The flexibility of these techniques may also open the possibility to extend their application to hadronic fast simulation showers. The classical GFLASH [65] approach relies on an analytical parameterized model of electromagnetic shower developed in calorimeters, obtained from fits to the simulated data. Once the model is obtained, it is applied during tracking: when a proper particle enters the calorimeter, its shower is generated by distributing energy spots according to the model, which is using much less calculations than the full development. Deep learning and generative AI techniques can respectively be used to replace the modeling phase and the generation one. Both LHC experiments and Geant4 are investing on these techniques and share efforts [66]. Since release v11.0, Geant4 is distributing the full AI-based example `Par04`, in the `examples/extended/parameterisations/` section. It demonstrates the deep-learning phase and the generative ones. This example makes use of several inference libraries [67–69] (see Fig. 10).

7.1.2 Physics process level accelerated simulation

AI-based techniques are also considered to speed up the calculations at the physics process level. The `QMD` (Quantum Molecular Dynamics) model is the most precise hadronic cascade model in Geant4, with an advanced quantum mechanics treatment of the nucleons wave functions. But this treatment comes at a high computing cost. Recently, a team [71] in Geant4 started a feasibility study aiming at replacing the detailed `QMD` calculations by an AI-based model: this one would be trained in a deep-learning phase, and would then emulate at high speed the `QMD` calculations during tracking.

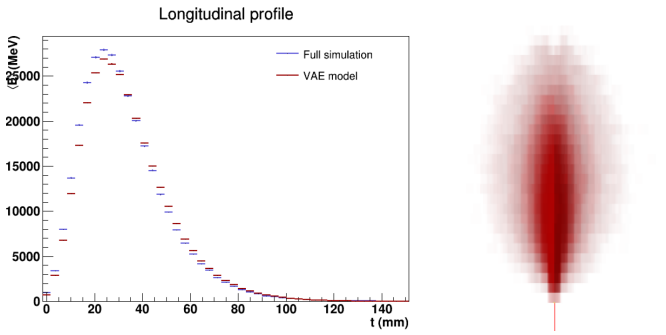

Fig. 10. Left: full simulation versus AI-based one comparison of longitudinal shower profiles from 10 GeV e^-'s in the segmented calorimeter in example `Par04`; right: an AI-generated shower; [70].

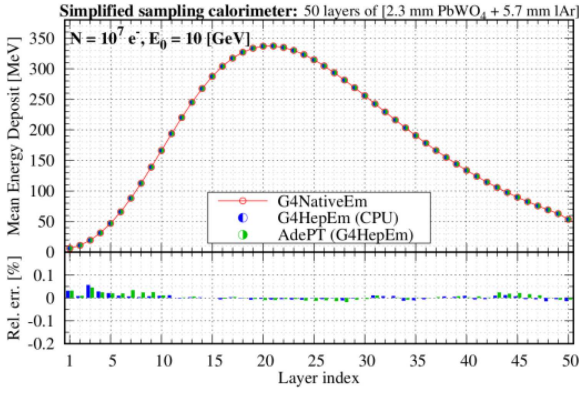

Fig. 11. Comparison of shower longitudinal profiles obtained with native Geant4 electromagnetic physics and the `G4HepEm` one, with processing on CPU and on GPU [74].

7.2 `G4HepEm` & specialized tracking

A rewrite of the electromagnetic physics needed in HEP applications has been made with the `G4HepEm` project [72]. This library relies on the idea of having a clear separation between the heavy classes, needed for the physics calculations made at initialization time to create various physics tables, from the light ones, used at tracking time and which essentially consist in fetching few values in some of these tables.

The tracking code hence produced is compact, which is favorable in terms of performances on CPU. Its simplicity makes it also portable to GPU. Figure 11 shows that it compares very well with the Geant4 EM native physics.

The Geant4 tracking is general, which comes at some performance cost. The `G4HepEm` library is currently used in an R&D consisting at evaluating what gain could be obtained with a tracking specialized for electromagnetic physics transport. Such optional tracking is indeed possible since release 11.0 with the `G4VTrackingManager` class. Preliminary evaluations show that a O(10%) gain is doable [73].

7.3 Studies of porting electromagnetic showers on GPU

The wide spreading of GPU usage in research fields in general and their massive use in HPC centers in particular, as well their potential benefit in terms of power consumption, call for assessing the portability of the particle transport simulation onto these devices.

The Geant4 physics has been partially ported to GPUs for medical applications [75] with O(100) or even larger speedup factors. These porting are however limited to low energy electromagnetic physics and voxel-like geometries, made simply of same size cubes.

A general HEP simulation porting is challenging because of the complexity of the physics, with many particle types suffering complex interactions, because of the diversity of geometry volumes and because of the Monte-Carlo nature of the transport code as it uses the previous features in a non predictable way. The GPU workflow relies on parallel threads that should execute the same instruction at any time to perform efficiently. A thread divergence appears when one or different instructions are needed in one of these threads, making the others waiting during this execution. The variety of the calculations needed together with the stochastic nature of the transport code make HEP simulation highly susceptible to thread divergence. The benefit of processing HEP simulation on GPU is then not easy to anticipate and must be evaluated practically.

The electromagnetic physics is the less complex one in HEP applications and it represents the dominating CPU consumption fraction because of electromagnetic showers simulation. The EM physics porting on GPU can hence be considered a priority milestone. Two projects, AdePT (Accelerated demonstrator of electromagnetic Particle Transport) [76] and Celeritas [77], are investigating along this line. They use either the `G4HepEm` package for the physics (AdePT) or a rewrite from scratch (Celeritas). They both can use `VecGeom` for the geometry, or a dedicated package, ORANGE (Oak Ridge Advanced Nested Geometry, Celeritas) [78]. The prototype codes are able today to process LHC-scale complex geometries and full electromagnetic physics. The CPU-GPU communications appear fast enough to send back GPU information to the CPU to process users' hits. Performance gains are still under evaluation, but factors 2 speedups look achievable [76,77] together with reduced power consumption by a same amount [77]. Performances do not scale with the number of CPU cores; an identified limiting factor is the geometry, with thread divergences caused by the variety of volumes. A rewrite of the solids, using surface-based representation, is currently on-going to make the calculation more GPU-friendly; benchmarks results are expected in the coming year.

8 Conclusion

The past decade has shown an important evolution of Geant4 in all areas of the toolkit. The parallel processing of events has become a de facto default processing, and the parallelism scheme has recently been renewed offering opportunities for new functionalities. The kernel part has included several new features, with new geometrical primitives, improved tracking in fields, more interactive scoring. The physics packages have largely been enriched and refined in all areas, and many options have been created in addition. This better physics quality could be provided without affecting the throughput performances, which have been continuously improved too.

Despite these progresses, strong demands on the throughput have been expressed by the HL-LHC community which also constraints the physics quality to augment adequately. To address these challenges, a number of new ideas and investments of R&D are going on, making Geant4, after 30 years of existence, a still very active project.

Acknowledgments

The author of this paper is author of a tiny part of the rich and various functionalities presented here. The credit for these works is to be given to the talented Geant4 developers, in all areas of the software. Possible mistakes or misinterpretations in this paper are to be attributed to its author.

Funding

The author thanks IN2P3/CNRS for the funding related to his developments. The Geant4 Collaboration is supported worldwide by many public institutions and some private ones, through various domestic or international programs. This support is an invaluable contribution to the continuous development and improvement of the Geant4 toolkit.

Conflicts of interest

The author declares no conflict of interest.

Data availability statement

No data are associated with the paper.

Author contribution statement

The author of this paper has been approved by the Geant4 Publication Board to represent the work of the Geant4 Collaboration developers.

References

1. S. Agostinelli et al., "Geant4 – A Simulation Toolkit", Nucl. Instrum. Meth. A **506**, 250 (2003)
2. J. Allison et al., "Geant4 Developments and Applications", IEEE Trans. Nucl. Sci. **53**, 270 (2006)
3. J. Allison et al., "Recent Developments in Geant4", Nucl. Instrum. Meth. A **835**, 186 (2016)
4. M. Asai et al., "Recent Developments in Geant4", Ann. Nucl. Energy **82**, 19 (2015)
5. J.R. Madsen, Parallel Tasking Library (PTL) – Lightweight C++11 multithreading tasking system featuring thread-pool, task-groups, and lock-free task queue (2020), https://github.com/jrmadsen/PTL
6. A. Gheata et al., VecGeom (2015), https://gitlab.cern.ch/VecGeom/VecGeom
7. J. Apostolakis et al., "A vectorization approach for multifaceted solids in VecGeom", EPJ Web Conf. **214**, 02025 (2019)

8. G. Amadio et al., "GeantV: Results from the prototype of concurrent vector particle transport simula- tion in HEP", Comput. Softw. Big Sci. **5**, 3 (2021)
9. AIDA, Common Software Tools (2015), https://gitlab.cern.ch/VecGeom/VecGeom
10. ATLAS Collaboration, "Software and computing for Run 3 of the ATLAS experiment at the LHC" (2024), ArXiv eprint [arXiv:2404.06335] (hep-ex)
11. S. Norraphat et al., "Full Simulation of CMS for Run-3 and Phase-2", EPJ Web Conf. **295**, 03017 (2024), https://doi.org/10.1051/epjconf/202429503017
12. J. Dormand, P. Prince, "A family of embedded Runge-Kutta formulae", J. Comput. Appl. Math. **6**, 19 (1980)
13. Fermilab. Muon g-2, 2024, url: https://muon-g-2.fnal.gov/index.html
14. F.E. Cellier, E. Kofman, Continuous System Simulation. en. (Springer, New York, NY, Mar. 2006)
15. B. Cockburn, C.-W. Shu, "The Runge-Kutta Discontinuous Galerkin Method for Conservation Laws V: Multidimensional Systems", J. Comput. Phys. **141**, 199 (1998)
16. E. Kofman, S. Junco, "Quantized-state systems: a DEVS Approach for continuous system simulation", Trans. Soc. Comput. Simul. Int. **18**, 123 (2001)
17. F. Salvat, J. Fernandez-Varea, J. Sempau, "PENELOPE-2008: A Code System for Monte Carlo Simulation of Electron and Photon Transport" (2009)
18. D.E. Cullen, J.H. Hubbell, L. Kissel, EPDL97: The Evaluated Photon Data Library '97 Version (1997), https://www-nds.iaea.org/epdl97
19. A. Sytov, V. Tikhomirov, L. Bandiera, "Simulation code for modeling of coherent effects of radiation generation in oriented crystals", Phys. Rev. Accel. Beams **22**, 064601 (2019)
20. A. Sytov, L. Bandiera, K. Cho, et al., "Geant4 simulation model of electromagnetic processes in oriented crystals for accelerator physics", J. Korean Phys. Soc. **83**, 132 (2023)
21. F. Nicolanti et al., "Geant4-DNA development for atmospheric applications: N2, O2 and CO2 models implementation", Phys. Medica Eur. J. Med. Phys. **128**, 104838 (2024)
22. D. Brandt et al., "Semiconductor phonon and charge transport Monte Carlo simulation using Geant4" (2014), Arxiv eprint [arXiv: 1403.4984] (physics.ins-det)
23. M.H. Kelsey et al., "G4CMP: Condensed matter physics simulation using the Geant4 toolkit", Nucl. Instrum. Methods Phys. Res. A **1055**, 168473 (2023)
24. S. Goudsmit, J.L. Saunderson, "Multiple scattering of electrons", Phys. Rev. **57**, 24 (1940)
25. M. Novak, "On the new and accurate (Goudsmit- Saunderson) model for describing e−/e+ multiple Coulomb scattering (Geant4 Technical Note)" (2024), Arxiv eprint [arXiv: 2410.13361] (physics.comp-ph)
26. S. Incerti, V. Ivanchenko, M. Novak, "Recent progress of Geant4 electromagnetic physics for calorimeter simulation", J. Instrum. **13**, C02054 (2018)
27. D. P. Watts et al., "Photon quantum entanglement in the MeV regime and its application in PET imaging", Nat. Commun. **12**, 2646 (2021)
28. P. Gros et al., "Performance measurement of HARPO: A time projection chamber as a gamma-ray telescope and polarimeter", Astropart. Phys. **97**, 10 (2018)
29. I. Semeniouk, D. Bernard, "C++ implementation of Bethe-Heitler, 5D, polarized γ →e+e- pair conversion event generator", Nucl. Instrum. Methods Phys. Res. A **936**, 290 (2019)
30. S. Incerti et al., "The geant4-DNA project", Adv. Complex Syst. **01**, 157 (2010)
31. S. Incerti et al., "Comparison of GEANT4 very low energy cross section models with experimental data in water", Med. Phys. **37**, 4692 (2010)
32. M.A. Bernal et al., "Track structure modeling in liquid water: A review of the Geant4-DNA very low energy extension of the Geant4 Monte Carlo simulation toolkit", Phys. Med. **31**, 861 (2015)
33. S. Incerti et al., "Geant4-DNA example applications for track structure simulations in liquid water: A report from the Geant4-DNA Project", Med. Phys. **45**, e722 (2018)
34. H.N. Tran et al., "Review of chemical models and applications in Geant4-DNA: Report from the ESA BioRad III Project", Med. Phys. **51**, 5873 (2024)
35. W.-G. Shin et al., "A Geant4-DNA evaluation of radiation-induced DNA damage on a human fibroblast", Cancers (Basel) **13**, 4940 (2021)
36. I. Kyriakou et al., "Review of the Geant4-DNA simulation toolkit for radiobiological applications at the cellular and DNA level", Cancers (Basel) **14**, 35 (2021)
37. I. Plante., "A review of simulation codes and approaches for radiation chemistry", Phys. Med. Biol. **66**, 03TR02 (2021)
38. N. Srimanobhas et al., "Full Simulation of CMS for Run-3 and Phase-2", EPJ Web Conf. **295**, 03017 (2024)
39. E. Woodcock et al., Techniques used in the GEM code for Monte Carlo neutronics calculations in reactors and other systems of complex geometry. Tech. rep. ANL-7050. Argonne National Laboratory, 1965
40. V.V. Uzhinsky, "The Fritiof (FTF) Model in Geant4", in *International Conference on Calorimetry for the High Energy Frontier* (2013), pp. 260
41. B. Andersson, G. Gustafson, B. Nilsson- Almqvist, "A model for low-pT hadronic reactions with generalizations to hadron-nucleus and nucleus- nucleus collisions", Nucl. Phys. B. **281**, 289 (1987)
42. B. Nilsson-Almqvist, E. Stenlund, "Interactions between hadrons and nuclei: The Lund Monte Carlo – FRITIOF version 1.6 –", Comput. Phys. Commun. **43**, 387 (1987)
43. B. Andersson, G. Gustafson, C. Peterson, "A statistical model for quark fragmentation into mesons with emphasis on vector meson contributions", Nucl. Phys. B. **135**, 273 (1978)
44. A. Galoyan et al., "Towards model descriptions of the latest data by the NA61/SHINE collaboration on ^{40}Ar + ^{45}Sc and ^{7}Be + ^{9}Be interactions, Eur. Phys. J. C Part. Fields, **82**, 2 (2022)
45. V. Lugovoi, "Rotating string" (1998), ArXiv eprint [arXiv: hep-ph/9811486] (hep-ph)
46. H.J. Schulze, J. Aichelin, "Fragmentation of color strings", Phys. Rev. D Part. Fields **43**, 2111 (1991)
47. A. B. Kaidalov, "The quark-gluon structure of the pomeron and the rise of inclusive spectra at high energies", Phys. Lett. B **116**, 459 (1982)
48. H. Fesefeld, Simulation of Hadronic Showers, Physics and Applications. Tech. rep. Technical Report PITHA 85-02. Aachen, Germany, 1985

49. P. Kaitaniemi et al., "INCL intra-nuclear cascade and ABLA DE-excitation models in Geant4", Prog. Nucl. Sci. Technol. **2**, 788 (2011)
50. D. Mancusi et al., "Extension of the Liège intranuclear-cascade model to reactions induced by light nuclei", Phys. Rev. C Nucl. Phys. **90**, 5 (2014)
51. L. Thulliez, C. Jouanne, E. Dumonteil, "Improvement of Geant4 Neutron-HP package: From methodology to evaluated nuclear data li- brary", Nucl. Instrum. Methods Phys. Res. A **1027**, 166187 (2022)
52. M Zmeškal, L. Thulliez, E. Dumonteil, "Improvement of Geant4 Neutron-HP package: Doppler broadening of the neutron elastic scattering kernel", Ann. Nucl. Energy **192**, 109949 (2023)
53. M Zmeškal et al., "Improvement of Geant4 Neutron-HP package: Unresolved resonance region description with probability tables", Ann. Nucl. Energy **211**, 110914 (2025)
54. R.R. Coveyou, R.R. Bate, R.K. Osborn, "Effect of moderator temperature upon neutron flux in infinite, capturing medium", J. Nucl. Energy **2.3**, 153 (1956)
55. B. Becker, R. Dagan, G. Lohnert, "Proof and implementation of the stochastic formula for ideal gas, energy dependent scattering kernel", Ann. Nucl. Energy **36**, 470 (2009)
56. A. Zoia et al., "Doppler broadening of neutron elastic scattering kernel in Tripoli-4@", Ann. Nucl. Energy **54**, 218 (2013)
57. E. Brun et al., "Tripoli-4@", Ann. Nucl. Energy **82**, 151 (2015)
58. Geant4 Physics List Guide. The Geant4 Collaboration (2024)
59. G. Inc. GitLab, 2024, url: https://about.gitlab.com/
60. A. Dotti et al., "Software aspects of the Geant4 validation repository", J. Phys. Conf. Ser. **898**, 042030 (2017)
61. G. Collaboration, GENIE Event Generator & Global Analysis of Neutrino Scattering Data, (2024) url: https://hep.ph.liv.ac.uk/?costasa/genie/index.html
62. A. Collaboration, ATLAS Software and Computing HL-LHC Roadmap (2022), url: https://cds.cern.ch/record/2802918/files/LHCC-G-182.pdf
63. C. Collaboration, CMS Phase-2 Computing Model: Update Document (2022), url: https://cds.cern.ch/record/2815292/files/NOTE2022008.pdf
64. L. Collaboration, LHCb CPU Usage Forecast (2019), url: https://lhcbproject.web.cern.ch/lhcbproject/Publications/f/p/LHCb-FIGURE-2019-018.html
65. G. Grindhammer, S. Peters, "The parameterized simulation of electromagnetic showers in homogeneous and sampling calorimeters" (2000), Arxiv eprint [arXiv: hep-ex/0001020] (hep-ex)
66. M.F. Giannelli et al., Fast Calorimeter Simulation Challenge 2022 (2022), url: https://calochallenge.github.io/homepage/
67. ONNX Runtime, 2018, url: https://github.com/microsoft/onnxruntime
68. lwtnn. 2024, url: https://github.com/lwtnn/lwtnn/
69. LibTorch. 2024, url: https://pytorch.org/cppdocs/frontend.html
70. Courtesy of Anna Zaborowska and Dalila Salamani. 2024
71. Carlo Mancini Terracciano team, Department of Physics, Sapienza University of Rome, Italy. INFN, Section of Rome, Italy. 2024
72. M. Novak et al., The G4HepEm R&D Project, 2020, url: https://github.com/mnovak42/g4hepem
73. J. Hahnfeld, B. Morgan, M. Novak, G4HepEm and Specialized Stepping/Tracking in Geant4. Report during 26th Geant4 Collaboration Meeting, 2024
74. Courtesy of M. Novak, J. Hahnfeld and AdePT Team. 2024
75. See for example the MPEXS project and references therein : 2024, https://wiki.kek.jp/display/mpexs/MPEXS+Project
76. A. Gheata et al., AdePT – Accelerated demonstrator of electromagnetic Particle Transport, 2024, https://github.com/apt-sim/AdePT, url: https://geant4.web.cern.ch/collaboration/workinggroups/taskforcerd/g4rd14.
77. S.R. Johnson et al., Celeritas, https://celeritas-project.github.io/celeritas/ and references therein, 2024, url: https://geant4.web.cern.ch/collaboration/workinggroups/taskforcerd/g4rd12
78. S.R. Johnson, R. Lefebvre, K Bekar. Orange: Oak ridge advanced nested geometry engine. Technical Report Technical Report ORNL/TM-2023/3190, Oak Ridge National Laboratory, 2023

Cite this article as: M. Verderi, for the Geant4 Collaboration. The Geant4 software toolkit evolution over the past decade, EPJ Nuclear Sci. Technol. **11**, 66 (2025), https://doi.org/10.1051/epjn/2025047