

Latest FLUKA developments

Gabrielle Hugo^{1,*}, Claudia Ahdida¹, Davide Bozzato¹, Daniele Calzolari^{1,2}, Francesco Cerutti¹, Alessia Ciccotelli^{1,3}, Anna Cimmino⁴, Arnaud Devienne¹, Andre Donadon Servelle^{1,5}, Patrycja K. Dyrzcz¹, Luigi Salvatore Esposito¹, Alice Formento¹, Robert Froeschl¹, Ruben García Alía¹, Simone Gilardoni¹, Andrea Gomes¹, Dávid Horváth⁴, Barbara Humann¹, Angelo Infantino¹, Anton Lechner¹, Benoit Lefebvre⁴, Giuseppe Lerner¹, Tommaso Lorenzon¹, David Lucsanyi^{1,6}, Matteo Magistris¹, Stefano Marin¹, Giuseppe Mazzola¹, Samuel Niang¹, Elzbieta Nowak¹, Francisco Ogallar Ruiz^{1,7}, Jean-Baptiste Potoine^{1,6}, Fabio Pozzi¹, Daniel Prelicpean^{1,8}, Volodymyr Rodin¹, Stefan Roesler¹, Marta Sabaté Gilarte¹, Mario Sacristan Barbero^{1,6}, Francesc Salvat Pujol¹, Philippe Schoofs¹, Alexandra-Gabriela Şerban^{1,9}, Ivan Sharankov¹⁰, Chris Theis¹, Marco Tisi¹, Andrea Tsinganis¹¹, Roberto Versaci⁴, Vasilis Vlachoudis¹, Andreas Waets^{1,12}, Markus Widorski¹ and Illia Zymak⁴

¹ European Organization for Nuclear Research (CERN), Geneva, Switzerland

² University of Padova, Department of Physics and Astronomy, Padova, Italy

³ University of Manchester and Cockcroft Institute, Department of Physics and Astronomy, School of Natural Sciences, Oxford Road, Manchester M13 9PL, United Kingdom

⁴ The Extreme Light Infrastructure ERIC, ELI Beamlines Facility, Za Radnicí 835, 252 41 Dolní Břežany, Czech Republic

⁵ Ecole Polytechnique Fédérale de Lausanne (EPFL), Institute of Physics, Lausanne, Switzerland

⁶ University of Montpellier, Institute of Electronics and Systems, Montpellier, France

⁷ University of Granada, Department of Atomic, Molecular and Nuclear Physics, Granada, Spain

⁸ Technical University of Munich (TUM), Department of Physics, Munich, Germany

⁹ Faculty of Physics, University of Bucharest, and “Horia Hulubei” National Institute of Physics and Nuclear Engineering, Bucharest-Magurele, Romania

¹⁰ Uppsala University, Department of Physics and Astronomy, Uppsala, Sweden

¹¹ European Commission, Joint Research Centre (JRC), Geel, Belgium

¹² University of Zurich (UZH), Physik-Institut, Zurich, Switzerland

Received: 11 June 2024 / Received in final form: 18 June 2024 / Accepted: 2 October 2024

Abstract. The FLUKA Monte Carlo code has recently undergone significant enhancements, driven by needs from its user community. Key improvements are discussed, such as a new point-wise treatment for the interactions of low-energy neutrons, the incorporation of a new model for nuclear elastic scattering of protons below 250 MeV, explicit generation of synchrotron radiation photons during charged particle tracking, a revised modeling approach for coherent effects in bent crystals, and the addition of arc-DPA scoring. While improving and extending FLUKA physics performance is essential, it is equally important to ensure the long-term maintainability of the codebase. This paper also outlines the strategy and substantial progress in evolving FLUKA to meet this objective. The FLUKA features are being translated into a new codebase, fulfilling essential criteria such as continuity in the user experience and compatibility with existing inputs, thus laying the foundation for a new FLUKA generation. The codebase makes use of the Geant4 toolkit when appropriate. Additionally, the enhanced collaboration with Geant4 has resulted in the development of an interface, detailed in this paper, that enables access to the FLUKA hadronic models from any Geant4 application.

1 Introduction

FLUKA [1–3] is a general-purpose Monte Carlo code for the simulation of coupled hadronic and electromagnetic radiation showers. It simulates the transport and interac-

tion of photons, leptons, hadrons, and ions in complex material geometries over a wide energy domain, spanning from the keV (or thermal neutron) to cosmic ray energies. Its development has been driven for over six decades to serve the needs of accelerator design, detector radiation background studies, radiation protection, medical physics, space research, etc. Over this period, four

* e-mail: gabrielle.hugo@cern.ch

generations of the codebase have been sequentially developed.

Since 2020, the FLUKA development team of the FLUKA.CERN Collaboration has been working on the maintenance of the existing FLUKA code, as well as the enhancement and extension of its performances. Additionally, the team engaged in a significant refactoring and rewriting of the codebase to ensure its long-term maintainability. This involves improving code readability, adopting a modular core structure to facilitate the integration of fixes and new contributions, broadening expertise within the team across all aspects of the codebase, and fostering synergies with Geant4 to mutually exploit parallel efforts.

This paper is structured as follows. [Section 2](#) outlines recent improvements of FLUKA physics capabilities, introduced in the FLUKA v4-3 and v4-4 releases. On the other hand, [Section 3](#) elaborates on the motivation for transitioning to a new FLUKA generation and on the migration strategy. Furthermore, it provides an overview of the ongoing developments and describes the interface enabling to access the FLUKA hadronic models from any Geant4 application.

2 Recent physics developments in FLUKA v4

2.1 New point-wise treatment for the interactions of low-energy neutrons

FLUKA versions preceding v4-3.0 relied on a group-wise treatment for neutron interactions below 20 MeV, except for a handful of isotopes and selected channels, for which an ad-hoc point-wise treatment could be invoked. While this approach offered appealing advantages (fast simulation and modest memory requirements), its drawbacks (lack of charged-secondary generation and transport, discrete angular distributions, uncorrelated residual nuclei production, etc.) made it inapplicable to e.g. neutron detector and other event-by-event simulations. A major effort was therefore recently invested in developing and implementing a novel point-wise treatment in FLUKA v4-3.0, with strong emphasis not only on physics performances but also on computational efficiency [4].

This new FLUKA point-wise implementation is based on cross-sections from evaluated nuclear data libraries, preprocessed at 0 K with PREPRO [5] by the Geant4 collaboration [6–8], and freely distributed by IAEA [9]. While the default choice is the JEFF library, other neutron databases (ENDF, JENDL, etc.) are made available as well. Furthermore, users are offered the flexibility to select which neutron library source to invoke on a per-isotope basis. Point-wise cross-sections are Doppler broadened at initialization at the user-prescribed temperature, using an adaptive numerical convolution algorithm respecting the initial tabular precision of 0.1%, while being fast enough to process all available 550 database isotopes in less than 20 s on a 3 GHz single-core computer.

For performance reasons, the interaction-channel selection is performed through a hierarchical structure, starting from a total-cross-section tabulation for each invoked element, followed by a selection of the actual isotope, after

which the interaction channel (elastic scattering, radiative capture, inelastic interaction, or fission) is determined, with a final subdivision into specific channels for inelastic interactions (to discrete levels or the continuum) and fission channels. Runtime interpolations in tabulated cross-section values employ a bisection method paired with an original indexing algorithm which exploits the IEEE754 floating-point format to constrain the search to a limited range [4]. This indexing method yields a search performance up to a factor of 10 faster than a bisection search for the sizeable point-wise cross-section tabulations employed in this approach, containing $\mathcal{O}(10^5)$ points.

All point-wise interactions are performed in a fully correlated manner: FLUKA reproduces a prescribed N -body final state through a series of two-body emissions, with the kinematics gradually constraining particle-emission distributions from the databases. In final states involving other particle species, emission priority is given to the neutrons, in order to faithfully respect neutron spectral features. Should this constrained subsequent emission yield an unphysical situation, the selected channel is resampled. Finally, the excited residual nucleus is treated by the FLUKA gamma de-excitation model, which includes known nuclear levels. Thanks to the refinement of the prompt gamma emission in FLUKA v4-4.0, relying on evaluated nuclear data in the ENSDF database, the FLUKA gamma de-excitation model is henceforth able to faithfully reproduce the gamma cascade following a neutron capture, including all correlations. For the fission channel, the available information in the databases (neutron multiplicity, pre-scission neutron multiplicity, mass distributions, gamma spectra) is employed; when missing, the Wahl systematics [10] for the fragment mass distribution is invoked, as well as an ad-hoc model for the pre-scission neutron multiplicity.

At thermal energies, the code uses the free-gas approximation with the constant cross-section model, and (upon user request) it can account for binding effects using $S(\alpha, \beta, T)$ thermal scattering laws (TSL). Both Geant4 and ACE format TSL libraries are supported, including mixed inelastic, elastic coherent, and incoherent scattering. An internal energy rescaling mechanism is used for the inelastic channel in order to physically correct the aliasing effect due to the sharp peak when the outgoing neutron energy is equal to that of the incoming neutron.

The point-wise implementation outlined above was intensively benchmarked and validated by the FLUKAVAL framework [11], offering superior physics performances to the ones of the previous group-wise implementation with a minimum CPU penalty. After more than one year of testing by our user community, it became the default model for neutron interactions below 20 MeV in FLUKA v4-4.0.

2.2 Nuclear elastic scattering of protons below 250 MeV

In FLUKA v4-4.0, a new model for the nuclear elastic scattering of protons below 250 MeV has been included [12], superseding a simpler model employed up to version 4-3.4

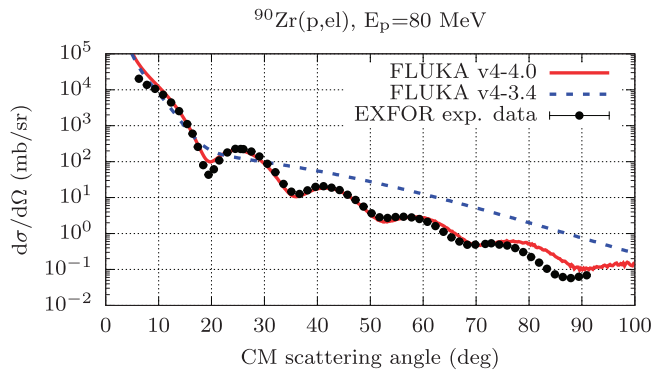


Fig. 1. Differential cross-section for the elastic scattering of 80 MeV protons on ^{90}Zr . Experimental data [16,17] are represented by the black symbols, while the differential cross sections sampled from FLUKA v4-4.0 and from FLUKA v4-3.4 are displayed by the solid red and dashed blue curve, respectively (statistical uncertainty bars have been omitted for clarity).

[13], covering only protons above 10 MeV. This newly developed model is based on both partial-wave analyses [14,15] and experimental angular distributions [16,17]. To reduce memory requirements in the framework of a general-purpose Monte Carlo code like FLUKA, a parametrized expression based on the black-disk limit has been adopted [18–20]. This model not only addresses FLUKA’s lack of nuclear elastic scattering for protons below 10 MeV, but it also provides a more accurate description of the structure of minima and maxima of the differential cross-section at large scattering angles, as shown in Figure 1.

The performances of this new model have been assessed in two applications where nuclear elastic scattering plays a significant role. On the one hand, FLUKA v4-4.0 has been employed to estimate the cross section for single-event-upset production induced by low-energy protons in commercial static random-access memories [12]. An improved agreement with experimental estimates has been obtained with respect to earlier FLUKA versions, for which the lack of nuclear elastic scattering of protons below 10 MeV led to an underestimation of several orders of magnitude in the 1–10 MeV range [21, 22]. On the other hand, the off-axis dose deposition by 100–250 MeV proton beams in water phantoms [23,24] has been assessed. The agreement between experimental and FLUKA dose estimates has been enhanced considerably at distances of 5–10 cm from the beam axis when using the new (p,el) model.

2.3 Emission of synchrotron radiation photons along charged particle steps

In FLUKA versions prior to v4-3.0, the emission of synchrotron radiation (SR) could be activated along a maximum of two trajectory arcs in uniform magnetic fields (at most with a sign variation among arcs) by means of a special source card, without explicit tracking of the emitting

charged particle. This approach offered limited applicability in practical simulations, especially when dealing with complex, non-uniform magnetic fields, further failing to account for changes in the energy of the emitting particle, and not being applicable to problems where SR emitted by secondary particles is significant. To overcome these limitations, SR emission by charged particles tracked in magnetic fields was included in FLUKA v4-3.0.

This new feature has recently allowed FLUKA to assess radiation loads in the FCC-ee [25], a future e^-e^+ collider with beam energies of up to 182.5 GeV, implying an energy loss of 10 GeV per turn due to SR emission. A dedicated study using FLUKA v4-3.0 provided valuable insight into the absorber and shielding design to protect the machine from the radiation load coming from the emitted SR photons [26]. FLUKA v4-3.0 was furthermore instrumental for evaluating the radiation load in prospective studies for a muon collider [27]. While the primary muons do not produce a significant SR amount due to their large mass, their decay generates a substantial number of secondary electrons with energies of up to 5 TeV. These electrons eventually impact the machine, generating showers and causing an important radiation load on the superconducting magnets. In this scenario, the proper account of their SR reduces the required amount of shielding to protect the equipment, as it disperses the energy deposition more uniformly on both sides of the beam aperture [28].

2.4 Coherent effects in bent crystals

The FLUKA module for the tracking of positively charged particles in crystal lattices is currently being recast in terms of explicit tracking in the electrostatic potential of adjacent crystal planes. This approach is free from earlier a priori assumptions and ad hoc treatments for various coherent transport effects, which now arise naturally, including: (i) the channeling of particles when their direction of motion is sufficiently close to the channel orientation, (ii) volume reflection when a particle trajectory gradually becomes tangent to a crystal channel orientation after having crossed a typically large number of crystal planes, and (iii) the dechanneling or volume capture as a result of an interaction significantly changing the particle direction of motion [2].

The oscillating trajectory of channeled particles is now tracked explicitly, with an amplitude that prescribes a minimum impact parameter with respect to the atoms in the adjacent crystal planes. A series of interaction mechanisms are accordingly suppressed runtime for channeled particles, including Coulomb scattering, nuclear elastic scattering as well as nuclear reactions, and the electronic stopping power [29]. For quasi-channeled particles, whose direction of motion may eventually become tangent to the crystal orientation, the nuclear interaction cross-sections are also modified, resulting in a higher interaction rate because an effectively higher atomic density is encountered as particles traverse atomic planes at grazing angles.

2.5 Scoring of athermal-recombination-corrected displacements per atom (arc-DPA)

FLUKA offers built-in scoring capabilities for a range of physical observables to quantify cumulative radiation damage effects in materials exposed to intense radiation fluxes, including dose deposition, non-ionizing energy loss, gas production (typically H and He), and displacements per atom (DPA). The latter is particularly relevant as a widely employed measure for the radiation-induced degradation of a material's microscopic structure (displacement damage). The DPA scoring capabilities of FLUKA were significantly enhanced in v4-3.0. On the one hand, the Norgett-Robinson-Torrens measure for DPA (an industry-standard in the foregoing decades) was added as a scoring quantity. On the other hand, a state-of-the-art evaluation of DPA was added, accounting for the athermal ion-hole recombination efficiency, following the prescription of Nordlund et al. [30]. This approach supersedes the recombination efficiency used in earlier FLUKA versions, which was strictly applicable to graphite only. This novel arc-DPA scoring functionality in FLUKA has been recently employed e.g. to assess the lifetime of superconducting magnets in the proposed 3-10 TeV muon collider [28].

2.6 Nuclear database modernization

FLUKA relies extensively on nuclear data libraries to operate, containing diverse elements such as nuclear levels and respective transition information as well as radioactive decay modes and products. This data is obtained from widely available sources such as NNDC or IAEA-RIPL [31,32] and stored in a nuclear database, which is in turn loaded during the code initialization. In its current form, the nuclear database has been part of FLUKA for several decades and was designed at a time when the capabilities of both programming languages and hardware were not comparable with the tools and methods at our disposal today.

The database has recently gone through a substantial modernization [33] which has brought forward several benefits. Firstly, it allows the data to be reorganized in a more modular way, facilitating its documentation, maintenance, and testing, as well as simplifying collaborative work on it. Secondly, legacy data structures have been replaced with up-to-date data science-inspired formats, which permits the storage of more data more flexibly. The whole database has been designed and built with the new FLUKA generation in mind and is fully integrated into it.

3 FLUKA v5: A new FLUKA generation

While the improvement of FLUKA radiation-matter interaction models and technical capabilities remains a necessity to satisfy the evolving needs of our user community, it is equally crucial to ensure the long-term maintainability of its codebase, currently hindered by a number of factors. The code structure excessively intertwines different components, such as tracking and scoring, which raises the

likelihood that a modification in one feature introduces regressions in another. Extensive use of shared variables increases the risk of unintended side effects when implementing new features. Finally, readability is compromised by factors such as the legacy use of 6-letter variable names, an occasionally convoluted control flow, and blocks of code replicated at multiple instances, each with specific variations. Most of these limitations stem from constraints imposed by the use of the FORTRAN 77 programming language, a lack of modularization of the codebase, and the at times incoherent addition of developments over various decades.

It is therefore necessary to undertake a substantial modernization of the code and a redesign of its architecture. Such a work allows us to set a solid core foundation for a new FLUKA generation, to which many collaborators can contribute in a sustainable manner, and facilitate synergies with other Monte Carlo code communities.

3.1 Codebase modernization and release strategy

Such an evolution should abide by a series of constraints. Firstly, simulation results should exhibit no drastic discontinuity: for instance, FLUKA stands as a reference for CERN radiation protection activities, encompassing shielding, activation, dosimetry studies, and instrumentation characterization for regulatory bodies like the Swiss OFSP (Office fédéral de santé publique) and French ASN (Agence de sûreté nucléaire). Secondly, the new framework should still support the extensive set of FLUKA v4 geometries and input files accrued over the past decades in the course of a wide variety of studies for accelerator design and operation. Additionally, it should remain compatible with associated tools such as the FLUKA coupling with particle tracking codes in accelerator lattices [34–36], the LineBuilder [37], and other post-processing codes, to name a few. Thirdly, FLUKA's nature as a fully integrated application and its derived user experience should be preserved. This approach offers users a ready-to-run application: they can directly focus on their studies descriptions and settings, as facilitated by the Flair graphical interface [38,39], and rely on a single set of physics models that represent the best of our knowledge at a given time. This approach is complementary to the one adopted by Geant4 [6–8], which, as a toolkit, enables users to develop and customize their own applications.

Concerning the code migration strategy, several options were evaluated. A complete rewrite of a general-purpose Monte Carlo code from scratch would be a remarkably extensive effort, resulting in redundant work. Conversely, adopting an alternative Monte Carlo code was impractical due to the aforementioned constraints. A hybrid strategy was adopted, consisting of the migration of FLUKA v4 physics models and technical capabilities onto a modular multi-threaded C++ application, henceforth called FLUKA v5. FLUKA v5 is specifically designed to ensure continuity in the FLUKA user community: it offers compatibility with FLUKA v4 inputs, geometry descriptions, auxiliary tools, and user interface, and a continuity in physics performance and technical features. It makes use of the

Geant4 toolkit, benefiting from its high degree of flexibility and customization. The migration of v4 features is not a mere translation from one language to another: a critical assessment is performed, and possible improvements are integrated, both in terms of physics and computational performances. Geant4 physics models and technical capabilities are adopted when they align with those of FLUKA, otherwise, they are temporarily used until the migration of their FLUKA v4 counterparts is completed.

The evolution strategy is illustrated in Figure 2. Migrated v4 physics models and technical capabilities are always made accessible from the new FLUKA v5 framework. Additionally, when technically possible, migrated modules are also made accessible from the current FLUKA v4 framework: such shared modules shall progressively be distributed in public releases of FLUKA v4, offering testing and possible enhancements in a gradual manner. When all relevant FLUKA v4 features have been interfaced in a sufficiently complete way to ensure equivalent (or improved) performances and user experience, possibly even before their C++ re-writing completion, FLUKA v5 will be publicly released.

The impact of such a migration on the Monte Carlo user communities is substantial. While FLUKA users should expect minimal disruption in their existing work, as enabled by the FLUKA v5 framework, a particularly notable side-effect is that they will gain access to all Geant4 physics lists alongside FLUKA physics. While not the primary motivation for the migration, this will enable them to directly conduct code inter-comparison studies using their existing v4 input files. On the other hand, the Geant4 user community will gain access to FLUKA physics models by design, thanks to their modularization and integration into a physics list. Regarding FLUKA v5 technical features, those of general interest for the development of other applications shall also be adapted and made available in the Geant4 toolkit.

3.2 Present status of FLUKA v5

In this section, a general overview of recent developments in FLUKA v5 is presented.

Geometry description paradigms in Geant4 and FLUKA are inherently different, somehow reflecting the distinct communities for which they were designed. Detectors in the LHC experiments have a very intricate nature and a huge number of volumes, which led to the creation of hierarchical and modular description schemes. Instead, the accelerator lattice has a lower level of intricacy, allowing the FLUKA geometry to be flat in the hierarchy and optimize navigation with closer neighbors caching. In order to avoid any error-prone conversion mechanism, the FLUKA v4 geometries, made of bodies, zones, and regions, can be converted into exactly equivalent entities, purposefully created as an extension to the GDML format [40]. Subsequently, FLUKA v5 builds the equivalent Geant4 geometry with all available information, treating each zone and its associated material as a Geant4 solid, logical, and physical volume. The resulting geometry is a set of Geant4 physical volumes, all placed at the same

hierarchy level. To efficiently navigate within this flat hierarchy structure, a custom navigator was created, incorporating neighbors caching algorithms akin to those used in FLUKA v4.

FLUKA v5 relies on the Geant4 implementation of particle production thresholds. It offers the additional possibility of setting particle production thresholds in terms of kinetic energy instead of range only. Regarding particle transport cutoff, FLUKA-v4-equivalent functionalities remain to be added.

Support for FLUKA v4 particle tracking capabilities in magnetic fields is already included in FLUKA v5, allowing the user to define analytical fields (from dipole to decapole), possibly combined with 2D/3D interpolated field maps. The implementation directly derives from the Geant4 one (`G4MagneticField`). Additionally, the user has the possibility to apply symmetries to define magnetic fields.

FLUKA v5 incorporates region importance and weight window biasing. These techniques are derived from the Geant4 implementation (`G4VBiasingOperator` and `G4VBiasingOperation`). Interaction length biasing needs to be extended to match the FLUKA v4 capabilities, while other features such as particle decay biasing and direction biasing remain to be implemented.

Efforts are ongoing to migrate FLUKA v4 physics models into v5. The models can be made available to the FLUKA v5 framework by isolating and re-writing outermost layers of the FLUKA v4 code, while still partly calling FLUKA v4 Fortran subroutines. This also grants the Geant4 community access to the FLUKA models at an early stage of the rewrite, and possibly outlines design constraints stemming from integration into a Geant4 application. In this spirit, the models for hadron-nucleus inelastic interactions were made accessible in v5 and any Geant4 application via an interface to the v4 code, as exposed in Section 3.3. Similarly, the v5 framework has now access to ion-ion inelastic interactions as interfaced in FLUKA v4. This includes the FLUKA BME module [41] and the dedicated deuteron model between ~ 1 MeV/n and 125 MeV/n, the RQMD event generator between 125 MeV/n and 5 GeV/n [42,43], and the DPMJET event generator above 5 GeV/n [44,45]. Other physics modules are being migrated: the de-excitation of excited residual nuclei by gamma decay, the Fermi breakup of excited nuclei, and nuclear fragmentation through fission. For now, they have been ported to C++ and provide binary identical results to the original FLUKA v4 versions. The modularization was achieved by isolating functions such as the determination of the nuclear temperature and moment of inertia for the gamma-ray de-excitation (called during preparatory steps in the simulation of the gamma-ray cascade), or the decay of unstable nuclei, as well as conservation checks, reference frame transformations and storage of the final state for the Fermi breakup of excited nuclei.

The analog simulation of radioactive decay is primarily based on the Geant4 implementation, enhanced by a flagging strategy for immediate discrimination between prompt and delayed radiation. In addition, an activation mode has been added, where the user can define an

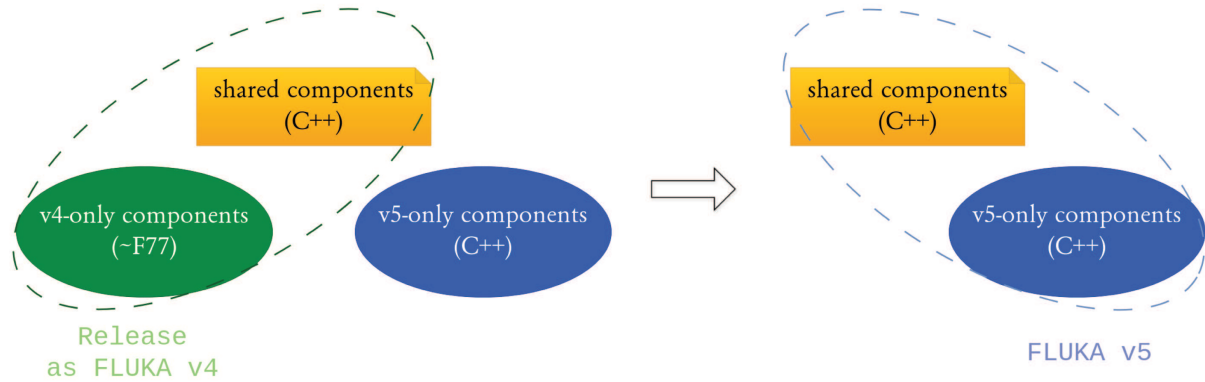


Fig. 2. FLUKA repository structure and evolution strategy. The plot on the left of the arrow illustrates the current development status and associated release content. The v4 features are progressively ported into v5-only and shared components. The plot to the right of the arrow illustrates the final repository structure once all v4 features have been ported.

arbitrary number of irradiation and cooling times profiles. This activation mode is based on an external build-up and decay engine [46,47] for a fully analytical resolution of the Bateman equations (in contrast with the Geant4 approach, where decay times are sampled), and on the use of Geant4 decay tables for the emission of decay products. It allows for a highly efficient analytical calculation of isotope activities and enables the evaluation of residual dose rates. Prompt and delayed radiation calculations can be conducted simultaneously, and all particles need to be tracked only once to provide results for each irradiation profile, due to the use of sets of weights calculated via the build-up and decay engine.

The scoring infrastructure in FLUKA v5 is a standalone implementation. It does not rely on the Geant4 sensitive detectors and hits collections mechanism. This offers more flexibility in the way the scoring is triggered and associated with the geometry, and the choice of scoring data containers. Various scoring quantities are available, including track length in a volume, fluence and current across a boundary, reaction product yields, activity, and dose equivalent through the integration of fluence-to-dose coefficients. Scoring can be differential in one or two parameters, such as particle kinetic energy, angle, charge, LET, and time. A simple and flexible design allows for the straightforward incorporation of additional scoring quantities and differential parameters. Event-by-event scoring (including pulse-height energy deposition spectra) has been added, while the implementation of other features related to radiation effects on electronics, available in G4SEE [48], is in progress.

In this paragraph, the implementation of mesh scoring (in Cartesian and cylindrical modes) in FLUKA v5 is discussed. The rationale for developing a standalone implementation arises from the requirement to maintain consistency with the FLUKA v4 approach. In FLUKA v4, the mesh definition should not interfere with tracking, while in Geant4 the scoring manager and parallel worlds lead particle steps to be dependent on the mesh bin size. In addition, it is frequent for FLUKA users to define a very significant number of meshes for a study. Therefore, possible runtime performance optimizations and memory

usage reductions are critical. The implementation includes all necessary features: a custom user interface, step splitting algorithms applied inside the mesh (implying entire replacement of Geant4 parallel worlds), mesh scoring data containers, and handling of scoring data synchronization among threads. The aforementioned step-splitting algorithms in the mesh were written in a computationally performant way. They employ caching of the distance to the closest bin in each dimension and minimize the use of resource-intensive arithmetic operations. For instance, in cylindrical mode, there is no runtime call to trigonometric functions; instead, a binary search on cached azimuth bin borders is employed. Specific care is taken to limit memory allocation during runtime. Additionally, in order to limit total memory usage, worker containers have a fixed size limit. When the buffer is full, a synchronization mechanism is triggered, flushing the worker data to the master data container. One drawback of this design is that it precludes event-based statistics on mesh scores, aligning instead with the FLUKA v4 approach of a run-based granularity. The obtained mesh scores are consistent with those obtained through the Geant4 implementation. The achieved mesh scoring runtime performance ranged from an x40 to x100 speed-up with respect to the Geant4 implementation in the tested studies. The extent of the speed-up is directly dependent on the use cases, notably on the frequency of the splitting of the step across mesh bins. A substantial reduction in RAM usage was consistently observed, typically reaching one order of magnitude.

In terms of run management, the current framework relies on the Geant4 run and events structure. Additionally, an event generator mode has been introduced, enabling the printing of cross-sections and final states outside any event loop. Users can select the process of interest, with the description chosen directly from the active physics list. A noteworthy debugging feature has been added, allowing the restart of a simulation after a crashed event in a multi-threaded run. It does not rely on printing random engine status, thus minimizing I/O overhead. The solution involves computing the unique set of seeds associated with the specific event one wishes to reproduce. This uniqueness is ensured in the event strong-reproducibility

mode. A simulation can be directly resumed from any event.

FLUKA v5 can be distributed in source and binary formats. The binary distribution includes all necessary dynamic dependencies, ensuring ease of installation for the end user. Regarding quality assurance, continuous integration is being integrated into FLUKA v5 and will perform multi-platform builds and testing. Additionally, the FLUKA extensive testing suite FLUKAVAL [11] has been made compatible with FLUKA v5. A comprehensive set of testing scenarios remains to be incorporated.

We ensured compatibility with the Flair graphical user interface. By its application-agnostic design, the interface has been readily extended to enable users to maintain their accustomed FLUKA v4 workflow also when using FLUKA v5: this encompasses the editing of input files, the creation, visualization, and debugging of the simulation geometry, the launching and management of runs, the post-processing of output files, and the visualization of simulation results. Additionally, scripts were added in Flair to allow users to seamlessly convert FLUKA v4 input files into their v5 equivalents. Note that while such a conversion is technically necessary, it is rather the ongoing implementation of the v4 features into the v5 framework that enables support for the converted input files.

The FLUKA v5 framework is already being tested inside the FLUKA.CERN Collaboration, allowing users to directly access the Geant4 models from their FLUKA v4 input files, and hence carry out FLUKA-Geant4 inter-comparison studies. Notably, such studies are being performed on a wide range of geometries, stemming from simplified targets to more complex real cases such as CERN n_TOF [49], CHARM and LHC collimation [50].

3.3 Interface to FLUKA v4 hadron-nucleus inelastic interactions

An interface was created, giving access to the FLUKA v4 hadron inelastic interaction treatment from any Geant4 application. It is integrated into FLUKA v5, as well as into Geant4 from its 11.2 beta release (June 2023) and 11.2 release (December 2023) onward. The interface wraps the entire FLUKA PEANUT package [1,20,51], which models hadron-nucleon interactions (via resonance production and decay below a few GeV, and Dual Parton Model above) and hadron-nucleus interactions (via Gribov-Glauber multiple collision mechanism and detailed Generalised Intra-Nuclear Cascade followed by a pre-equilibrium stage) up to 20 TeV. The most relevant nuclear effects in the target nucleus are modeled (density, Fermi motion, Pauli-blocking, etc.). The interface also encompasses FLUKA equilibrium processes: evaporation, fission, Fermi break-up, and gamma de-excitation. The FLUKA hadron reaction cross-sections were obtained from parametrizations that fit available experimental data. Support for photonuclear reactions is also included, leveraging the nuclear environment description.

The interface incorporates modular C++ code, performing the necessary initialization and isolating inelastic scattering cross-sections and hadron-nucleus inelastic interactions final states as computed by FLUKA v4. It also includes generic C++ helper tools and wrappers to FLUKA v4 utilities. Finally, it contains the necessary utilities to plug the work into a Geant4 environment, such as FLUKA-Geant4 interfacing tools for particle identification and random numbers engines, as well as a physics list and related classes.

The interface provides access to the FLUKA models both at the interaction and at the physics list level. Users have the flexibility to activate a few options, such as coalescence and heavy fragment evaporation. Two extended hadronic examples were included in the Geant4 repository, demonstrating how to make use of this FLUKA interface in any Geant4 application.

4 Summary and outlook

FLUKA has recently undergone physics performance enhancements. Concurrently, in order to ensure the code-base's long-term maintainability, FLUKA physics models and technical capabilities are being progressively rewritten onto a modular multi-threaded C++ application (FLUKA v5). The latter is purposefully created to ensure the persistence of the FLUKA user community experience: it shall offer compatibility with FLUKA v4 inputs, geometry descriptions, auxiliary tools, and user interface (Flair), along with the continuity in respect to FLUKA v4 physics and computing performances. It makes use of the Geant4 toolkit to avoid duplication of efforts and mutually exploit natural synergies. While the primary objective is to establish a robust core foundation for the next FLUKA generation, FLUKA users will also derive immediate benefits: they will have the opportunity to directly conduct code inter-comparison studies using their existing v4 input files.

While the FLUKA v5 development version is currently in use in the context of the FLUKA.CERN Collaboration, efforts are ongoing by the latter to implement crucial v4 features and carry out extensive testing, so as to be able to proceed with its public release. A particularly notable perspective of this endeavor is that it shall enable the Geant4 user community to gain access to several FLUKA physics models (thanks to their integration into a physics list) and a selection of relevant technical features.

Funding

This research did not receive any specific funding.

Conflicts of interest

The authors declare that they have no competing interests to report.

Data availability statement

Data is available upon request.

Author contribution statement

The authors of this paper are members of the FLUKA.CERN Collaboration. G. Hugo wrote Sections 1–4 and coordinated the paper preparation. V. Vlachoudis, A.-G. Şerban, D. Calzolari, L. S. Esposito, F. Salvat Pujol, and P. Schoofs wrote Sections 2.1–2.6, respectively. All authors read the paper and provided feedback. The manuscript revision was performed by F. Cerutti, F. Salvat Pujol and V. Vlachoudis.

References

- G. Battistoni et al., Overview of the FLUKA code. *Ann. Nuclear Energy* **82**, 10 (2015)
- C. Ahdida et al., New capabilities of the FLUKA multi-purpose code. *Front. Phys.* **9**, 788253 (2022)
- The official CERN FLUKA website, <https://fluka.cern>
- V. Vlachoudis et al., Recent developments in the point wise neutron treatment for FLUKA v4. *EPJ Web of Conf.* **284**, 03021 (2023)
- D. Cullen, PREPRO 2023, IAEA: IAEA-NDS-0241, 2023
- S. Agostinelli et al., Geant4—a simulation toolkit. *Nucl. Inst. Meth. A* **506**, 250 (2003)
- J. Allison et al., Geant4 developments and applications. *IEEE Trans. Nucl. Sci.* **53**, 270 (2006)
- J. Allison et al., Recent developments in Geant4. *Nucl. Inst. Meth. A* **835**, 186 (2016)
- IAEA nuclear data services online resource, Evaluated neutron cross section libraries for the Geant4 code (v2.0, 17/05/2018), <https://www-nds.iaea.org/geant4/>
- A.C. Wahl, Systematics of Fission-product Yields Tech. Rep., 2002 (Los Alamos National Lab.)
- M. Witorski, D. Bozzato, R. Froeschl, V. Kouskoura, FLUKAVAL – A validation framework for the FLUKA radiation transport Monte Carlo code, in *EPJ Web of Conferences*, Vol. 284 (2023)
- A.-G. Serban, A. Coronetti, R. García Alía, F. Salvat Pujol, Nuclear elastic scattering of protons below 250 MeV in FLUKA v4-4.0 and its role in single-event-upset production in electronics, *Comput. Phys. Commun.* **303**, 109276 (2024)
- J. Ranft, Estimation of radiation problems around high-energy accelerators using calculations of the hadronic cascade in matter. *Part. Accel.* **3**, 129 (1972)
- A.J. Koning, J.P. Delaroche, Local and global nucleon optical models from 1 keV to 200 MeV, *Nucl. Phys. A* **713**, 231 (2003)
- F. Salvat, J.M. Fernandez-Varea, RADIAL: A Fortran subroutine package for the solution of the radial Schrodinger and Dirac wave equations, *Comp. Phys. Comm.* **240**, 165 (2019)
- N. Otuka et al., Towards a More Complete and Accurate Experimental Nuclear Reaction Data Library (EXFOR): International Collaboration Between Nuclear Reaction Data Centres (NRDC), *Nucl. Data Sheets* **120**, 272 (2014)
- V. Zerkin, B. Pritychenko, The experimental nuclear reaction data (EXFOR): Extended computer database and Web retrieval system. *Nucl. Instrum. Meth. A* **888**, 31 (2018)
- I.M. Dremin, Elastic scattering of hadrons. *Phys. Uspekhi* **56**, 3 (2013)
- V.M. Grichine, Geant4 hadron elastic diffuse model. *Comp. Phys. Comm.* **181**, 921 (2010)
- A. Ferrari, P. Sala, The physics of high energy reactions, in *Proc. Workshop on Nuclear Reaction Data and Nuclear Reactor Physics, Design and Safety* (World Scientific, 1998), p. 424
- A. Coronetti et al, Assessment of Proton Direct Ionization for the Radiation Hardness Assurance of Deep Submicron SRAMs Used in Space Applications, *IEEE Trans. Nucl. Sci.* **68**, 937 (2021)
- A. Coronetti, Relevance and guidelines of radiation effect testing beyond the standards for electronic devices and systems used in space and at accelerators PhD thesis (University of Jyväskylä, 2021), <https://cds.cern.ch/record/2799812/files/CERN-THESIS-2021-255.pdf>
- N. Verbeek et al., Single pencil beam benchmark of a module for Monte Carlo simulation of proton transport in the PENELOPE code, *Med. Phys.* **48**, 456 (2021)
- D.C. Hall et al., Validation of nuclear models in Geant4 using the dose distribution of a 177 MeV proton pencil beam, *Phys. Med. Biol.* **61**, N1 (2015)
- I. Agapov et al., Future Circular Lepton Collider FCC-ee: Overview and Status, 2022, ArXiv preprint [arXiv:2203.08310] [physics.acc-ph]
- B. Humann, FCC-ee Radiation Challenges and Mitigation Measures, Ph.D. Thesis, TU Vienna, 2023, <https://cds.cern.ch/record/2879400>
- C. Accettura et al., Towards a muon collider, *Eur. Phys. J. C* **83**, 864 (2023)
- D. Calzolari et al., Radiation load studies for superconducting dipole magnets in a 10 TeV Muon Collider, in *Proc. IPAC'22 Bangkok, Thailand* (2022), pp. 1671–1674
- H. Esbensen et al., Random and channelled energy loss in thin germanium and silicon crystals for positive and negative 2-15-GeV/c pions, kaons, and protons, *Phys. Rev. B* **18**, 1039 (1978)
- K. Nordlund et al., Improving atomic displacement and replacement calculations with physically realistic damage models, *Nat. Commun.* **9**, 1084 (2018)
- ENSDF database online resource, <https://www.nndc.bnl.gov/ensdf/>
- RIPL database online resource, <https://www-nds.iaea.org/RIPL-1/>
- P. Schoofs, Towards a modernized physics database for FLUKA, in *Proc. SNA+MC 24, Paris, France* (2024)
- A. Mereghetti et al., SixTrack-FLUKA active coupling for the upgrade of the SPS scrapers, *Conf. Proc.* **130512**, WEPEA064 (2013)
- E. Skordis et al., FLUKA coupling to sixtrack, *CERN Yellow Rep. Conf. Proc.* **2**, 17 (2018)
- F. Van der Veken et al., Recent developments with the new tools for collimation simulations in *Xsuite*, in *Proc. 68th Adv. Beam Dyn. Workshop High-Intensity High-Brightness Hadron Beams Oct. 9–13, 2023* (2024), pp. 474–478
- A. Mereghetti, R. Versaci, V. Boccone, F. Cerutti, V. Vlachoudis, The fluka linebuilder and element database: tools for building complex models of accelerators beam lines, in *Conf. Proc.* (2012), Vol. 1205201, p. WEPPD071
- V. Vlachoudis, FLAIR: A powerful but user friendly graphical interface for FLUKA, in *Proc. Int. Conf. on Mathematics, Computational Methods and Reactor Physics* (2009)
- V. Vlachoudis, FLAIR3 – recasting simulation experiences with the Advanced Interface for FLUKA and other Monte Carlo codes, in *Proc. SNA+MC 24, Paris, France* (2024)

40. R. Chytráček, J. McCormick, W. Pokorski, G. Santin, Geometry description markup language for physics simulation and analysis applications, *IEEE Trans. Nucl. Sci.* **53**, 2892 (2006)
41. F. Cerutti et al., Low energy nucleus-nucleus reactions: the BME approach and its interface with FLUKA, in *Proceedings of the 11th International Conference on Nuclear Reaction Mechanisms* (2006)
42. H. Sorge, H. Stöcker, W. Greiner, Poincaré invariant Hamiltonian dynamics: Modelling multihadronic interactions in a phase space approach, *Ann. Phys.* **192**, 266 (1989)
43. V. Andersen et al., The FLUKA code for space applications: recent developments, *Adv. Space Res.* **34**, 1302 (2004)
44. S. Roesler, R. Engel, J. Ranft, The monte carlo event generator DPMJET-III, in *Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications: Proceedings of the Monte Carlo 2000 Conference, Lisbon, 23–26 October 2000* (2001), pp. 1033–1038
45. A. Fedynitch, Cascade equations and hadronic interactions at very high energies Ph.D. Thesis (KIT, Karlsruhe, Dept. Phys., 2015), <http://cds.cern.ch/record/2231593/files/CERN-THESIS-2015-371.pdf>.
46. H. Vincke, C. Theis, ActiWiz – optimizing your nuclide inventory at proton accelerators with a computer code, *Prog. Nucl. Sci. Tech.* **4**, 228 (2014)
47. H. Vincke, C. Theis, ActiWiz 3an overview of the latest developments and their application, *J. Phys. Conf. Ser.* **1046**, 012007 (2018)
48. D. Lucsányi et al., G4SEE: A Geant4-based single event effect simulation toolkit and its validation through monoenergetic neutron measurements, *IEEE Trans. Nucl. Sci.* **69**, 273 (2022)
49. J.A.P. Rodríguez, Commissioning of the nTOFEAR2 facility with the third-generation target and the related Monte Carlo simulations with Geant4, Ph.D. Thesis, Universitat de Sevilla, 2024
50. A.D. Servelle, Bridging Monte Carlo Worlds: A New Framework for Accelerator Physics Simulations Applied on Shielding and Machine Protection Studies, Ph.D. Thesis, EPFL, 2024
51. F. Ballarini et al., The physics of the FLUKA code: Recent developments, *Adv. Space Res.* **40**, 1339 (2007)

Cite this article as: Gabrielle Hugo, Claudia Ahdida, Davide Bozzato, Daniele Calzolari, Francesco Cerutti, Alessia Ciccotelli, Anna Cimmino, Arnaud Devienne, Andre Donadon Servelle, Patrycja K. Dyrzcz, Luigi Salvatore Esposito, Alice Formento, Robert Froeschl, Ruben García Alía, Simone Gilardoni, Andrea Gomes, Dávid Horváth, Barbara Humann, Angelo Infantino, Anton Lechner, Benoit Lefebvre, Giuseppe Lerner, Tommaso Lorenzon, David Lucsanyi, Matteo Magistris, Stefano Marin, Giuseppe Mazzola, Samuel Niang, Elzbieta Nowak, Francisco Ogallar Ruiz, Jean-Baptiste Potoine, Fabio Pozzi, Daniel Prelipcean, Volodymyr Rodin, Stefan Roesler, Marta Sabaté Gilarte, Mario Sacristan Barbero, Francesc Salvat Pujol, Philippe Schoofs, Alexandra-Gabriela Șerban, Ivan Sharankov, Chris Theis, Marco Tisi, Andrea Tsinganis, Roberto Versaci, Vasilis Vlachoudis, Andreas Waets, Markus Widorski, Illia Zymak. Latest FLUKA developments, *EPJ Nuclear Sci. Technol.* **10**, 20 (2024)