

Radioactive Decays in Geant4

Steffen Hauf, Markus Kuster, Matej Batič, Zane W. Bell, Dieter H.H. Hoffmann, Philipp M. Lang, Stephan Neff, Maria Grazia Pia, Georg Weidenspointner and Andreas Zoglauer

Abstract—The simulation of radioactive decays is a common task in Monte-Carlo systems such as Geant4. Usually, a system either uses an approach focusing on the simulations of every individual decay or an approach which simulates a large number of decays with a focus on correct overall statistics. The radioactive decay package presented in this work permits, for the first time, the use of both methods within the same simulation framework — Geant4. The accuracy of the statistical approach in our new package, RDM-extended, and that of the existing Geant4 per-decay implementation (original RDM), which has also been refactored, are verified against the ENSDF database. The new verified package is beneficial for a wide range of experimental scenarios, as it enables researchers to choose the most appropriate approach for their Geant4-based application.

Index Terms—Geant4, Radioactive Decay, Monte-Carlo Simulation, Validation, ENSDF.

I. INTRODUCTION

Radioactive decays and the resulting radiation play an important role for many experiments, either as an observable, as a background source, or even as a potential hazard when they are a source of radiation-induced damage for hardware and human beings. Detailed knowledge of the radiation inside and around an experiment and its detectors is thus required for a successful outcome of the experiment and to guarantee the safety of the operator. The increasing complexity of experiments often makes it prohibitively expensive, if not impossible, to completely determine the radiation characteristics and response of an experiment from measurements alone. In order to circumvent these limitations, it has become increasingly important to estimate an experiment's radiation and response characteristics with the help of computer simulations.

General-purpose Monte-Carlo simulation codes either focus on the correct simulation of individual decays (e.g., Geant4 [1], [2], see Section IV-A) or the statistical outcome of many decays (e.g., MCNP [3], [4] and FLUKA [5]).

Whereas the first approach may be inefficient if the individual decay is not of interest, the latter approach does not allow for the physically correct simulation of an individual decay and its associated effects. General purpose Monte Carlo

codes would benefit from the capability of providing both approaches in the same environment, in response to the simulation requirements of different experimental scenarios.

In the following a software package for the simulation of radioactive decay, which realizes both approaches for Geant4, is presented. This package includes a refactored implementation of the existing Geant4 per-decay approach [6], and extends the functionality of Geant4 radioactive decay simulation by a novel implementation based on a statistical approach. It is based on the ENSDF (Evaluated Nuclear Structure Data File) data library [7], which was chosen due to its widespread usage in the nuclear science community.

This paper reports on the verification of both implemented approaches against a large set of evaluated data. To the best of the authors' knowledge, such a thorough verification of Geant4 radioactive decay simulation has not yet been documented in the literature. The experimental validation of the software for both approaches is reported on in a separate paper [8].

II. RADIOACTIVE DECAY PHYSICS

Radioactive decay is a physical process where an atomic nucleus of an unstable atom transmutes into a lower energy state by spontaneous emission of ionizing radiation. The process does not require external interactions to occur. It results from either nucleus-internal processes or interactions of the nucleus with (inner) shell electrons. A brief overview of the main physics of radioactive decay is summarized here to facilitate the comprehension of the functionality of the software described in this paper.

Different types of radioactive decay are commonly identified according to the type of emitted particles.

- During an α -decay a He-nucleus is emitted from the parent nucleus. This results in a daughter nucleus with two fewer protons and two fewer neutrons than the parent nucleus.
- The β^- -decay is a weak process during which a neutron is converted into a proton. An electron and anti-neutrino are emitted by the parent nucleus: consequently, the atomic number of the daughter nucleus increases by one and the atomic mass number stays constant. The electron and anti-neutrino share the energy released during the decay. Since both particles are not bound in their final state, their energy distribution follows a continuous spectrum.
- During a β^+ -decay a bound proton of a nucleus is converted into a neutron. A positron and a neutrino are emitted by the parent nucleus; the atomic number decreases by one and the atomic mass number stays constant. Similar to β^- -decays, both particles are not bound in their final state, and accordingly, their energy distribution follows a continuous spectrum.

Manuscript submitted January 28, 2012. This work has been supported by Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR) under grants 50 QR 0902 and 50 QR 1102.

S. Hauf and M. Kuster are with European XFEL GmbH, Hamburg, Germany (e-mail: steffen.hauf@xfel.eu)

D.H.H. Hoffmann, P.M. Lang and S. Neff are with Institute for Nuclear Sciences, TU Darmstadt, Darmstadt, Germany

M.G. Pia and M. Batič are with the INFN Genova, Genova, Italy

G. Weidenspointner is with the Max-Planck Halbleiter Labor, Munich, Germany and the Max-Planck Institut für extraterrestrische Physik, Garching, Germany

A. Zoglauer is with the Space Science Laboratory, University of California, Berkeley, CA, USA

Z.W. Bell is with the Oak Ridge National Laboratory, Oak Ridge, TN, USA

- If a daughter nucleus is left in an excited state, after a transmutation by the previously mentioned decay types, it can deexcite by emitting γ -radiation. In case the excited daughter state is a long-lived (metastable) state, its deexcitation is called isomeric transition, which will also result in γ -radiation. In both cases the atomic number and atomic mass number remain unchanged.
- During an electron capture, the parent nucleus absorbs an inner shell electron (usually K- and L-shell electrons) and simultaneously emits a neutrino. During this process, which is also called inverse β -decay, a proton is transmuted into a neutron, thus the atomic number decreases by one and the atomic mass number stays constant. In contrast to a β -decay, an electron capture is a two-body decay, resulting in a discrete neutrino energy.
- As an alternative process to γ -emission, an excited nucleus can return to its ground state by transferring its excitation energy to one of the lower shell electrons of the atom. This process is called internal conversion, and results in the emission of an electron by the atom, leaving the atom in an excited state. The electron carries a discrete fraction of the decay energy, and by this is distinct from β -particles with continuous energy spectra. As with γ -decays, no transmutation of the nucleus takes place, and both the atomic number and atomic mass number remain unchanged.

Radioactive decay is a stochastic process. The time at which a given unstable atom decays is not predetermined; instead decays occur with a certain probability. In consequence, experiments will measure statistical observables such as the amount of ionizing radiation of a certain type and energy emitted within a given time period.

Due to practical impossibility of calculating all relevant parameters from theory, the simulation of radioactive decay physics for a large number of decays or decay chains relies on the usage of empirical or pre-calculated data.

III. EXPERIMENTAL SCENARIOS

Many experiments measure the time-accumulated statistical distributions of observables such as energy, type, momentum and timing of radiation resulting from radioactive decays. Often the radioactive decay products are not the intended observable but contribute to the experiment's or application's measured data as background.

Radioactive decay physics modeling plays an important role in various fields; the overview summarized here is not intended to be exhaustive.

Measurement and analysis of the properties of decay chains of naturally abundant radioactive isotopes is commonly performed in material sciences, radiation safety and nuclear proliferation monitoring. Radioactive decay modeling is also of interest to experimental scenarios involving the measurement of material properties after irradiation: for instance, of a sample irradiated by a neutron beam.

The study of activation and build-up of radioactive nuclei is relevant to various applicative domains: nuclear reactors (fission and fusion), particle accelerators or intense light

sources, where the statistical effect of many decays and activations is relevant. It also concerns space-borne X-ray and γ -ray instruments: in these scenarios meta-stable states usually must be accounted for, and the statistical effect of many decays and activations is relevant. Additionally, individual decays contribute to the prompt instrument background. The estimation of the in-orbit cosmic-ray induced background is important for space-based detectors to distinguish individual radioactive emission from intended observables.

Low background astro-particle physics experiments are concerned with the influence of the cosmic-ray induced background and natural radioactivity: here accurate simulation of the individual decays' spatial and temporal distribution can be important.

IV. FOUNDATIONS FOR A RADIOACTIVE DECAY SIMULATION

A. Requirements

The simulation of radioactive decays consists of the task of decaying an unstable nucleus and generating the resulting products.

The decay of the parent nucleus should proceed according to the physical parameters governing it: decay type, initial excitation and half-life time. A daughter nucleus should be produced, with the physical properties resulting from the decay: atomic number and mass determined by the decay type, excitation energy and kinetics determined by parent kinetics and decay kinetics. Secondary particles and radiation associated with the decay (e.g. β -, γ - and α -emission, neutrinos and conversion electrons) should be generated.

The software should handle the deexcitation of the daughter atom, involving the production of γ - and X-rays, and of Auger-electrons.

Theoretical calculations of the required parameters are not practically feasible in the course of the simulation; therefore the algorithm must use empirical or pre-calculated data.

Radioactive decays of nuclei are often associated with prior activation of stable nuclei of a given material; examples for such applications include shielding analysis and radiation safety analysis. This scenario requires the ability of dealing with activation, and thus replenishment of nuclei, within the simulation. Meta-stable states and isomeric transitions should be taken into account.

In addition to these functional [9], physics-induced requirements, the software should take into account non-functional ones, which derive from the experimental context of the simulation: the possibility to efficiently simulate a large number of decays, when the physical accuracy of the individual decay is of less importance, but overall statistics are relevant; the possibility of efficiently simulating decay chains, when intermediate products in a chain are of lesser interest; the provision of user input.

If radioactive decay simulation occurs in the context of a more general Monte Carlo simulation system, the software responsible for the radioactive decay process should interact with other components of the system.

B. Problem Domain Analysis

Software objects with specialized responsibilities collaborate to satisfy the requirements mentioned in Section IV-A. For the code presented in this work the division of responsibilities is as follows:

- data management
- sampling of the (discrete) emission resulting from the individual decay and generation of the daughter nucleus
- calculation of the β -emission spectrum,
- calculation of the number of nuclei within a decay chain — which may include activation
- the user interface
- the interface with the Monte-Carlo code

V. THE ENSDF DATA

The sampling of radioactive decays will usually rely on the use of empirical or pre-calculated data. The Evaluated Nuclear Structure Datafile (ENSDF) [7] is one such collection of data, which for instance contains information on half-life times, decay types, branching ratios, emission energies and transition types. It is maintained by the National Nuclear Data Center (NNDC) and distributed, amongst others, by the International Atomic Energy Agency (IAEA).

ENSDF is an evaluated library, i.e. it contains data from experiments and theoretical calculations which are recommended for use as a reference after a critical analysis of uncertainties, inter- and extrapolation methods and underlying models has been performed. IAEA defines it as the master library for evaluated experimental structure and decay data [10]. Other specialized libraries and bibliographic databases, such as NuDAT [11], CINDA [12] and NSR [13], exist as well, but are commonly derived from or related to ENSDF. Therefore ENSDF is frequently considered the de-facto standard for nuclear structure and radiation data.

For γ -ray intensities and energies the ENSDF data usually consists of evaluated measurements. Conversion electron intensities and energies are derived from theory.

However, the data present in ENSDF are not sufficient to calculate all atomic deexcitation emissions. Specifically, data on electron binding energies, fluorescence and Auger-electron yields are necessary to calculate the respective intensities. To mitigate this problem, analysis programs distributed with ENSDF, such as RADLST [14], are supplied with the necessary information taken from data of Bearden and Burr [15] and Bambynek *et al.* [16]. In order to stay consistent with these ENSDF-related programs, the aforementioned data are also used to derive quantities for the radioactive decay database of the RDM-extended package and the verification data used in this work.

VI. RADIOACTIVE DECAY IN MONTE CARLO CODES

Models for the simulation of radioactive decays exist in most general-purpose Monte-Carlo codes.

MCNP(X) [3], [4] does not include a full radioactive decay simulation by default except for the generation of delayed γ -rays resulting from decays sampled from MCNP's photon data library (phtlib). Instead MCNP can be linked via scripts to

specialized codes such as ORIGEN2 [17] or CINDER90 [18], [19]. Both of these use their own data libraries. These codes are generally used to model reactor fuel cycles and accelerator induced transmutations, but also provide functionality for simulating radioactive decays and decay chains. Due to the codes' nature, they include replenishment through activation. The α - and β -emission needs to be generated from tabulated user input. As a result, MCNP is specialized on the simulation of many decays and the resulting statistics.

FLUKA [5] generates and transports β - and γ -radiation, but only started including α -radiation in its latest release. Decay chains are possible and include replenishment through activation. FLUKA uses its own data libraries, largely based on NNDC (National Nuclear Data Center) data and thus ENSDF. Similar to MCNP, the emphasis lies on the simulation of a large number of decays.

In Geant4 [1], [2] radioactive decays are treated on a per-decay level, based on data taken from ENSDF [7]. Decay chains including activation are possible and produce the associated decay emission. The α - and β -emission are sampled from the decay database. Deexcitation radiation of the daughter nucleus is not produced by the radioactive decay simulation itself, but by other physics processes included in Geant4, which use their respective databases. The emphasis lies on the per-decay simulation, not on the sampling of a large number of decays.

None of the above mentioned Monte-Carlo codes allows the simulation of either individual decays or a statistical treatment of many decays in the same software environment.

VII. RADIOACTIVE DECAY IN GEANT4

A package for the simulation of radioactive decays [6], [20] has been available in Geant4 since version 2.0, where it was named *radiative_decay*. Since Geant4 version 6.0 it has been named *radioactive_decay*, although it is conventionally known as the Geant4 RDM (Radioactive Decay Module). This code was originally developed by P. Truscott and F. Lei; it implements per-decay sampling.

The following discussion is based on the radioactive decay code of Geant4 9.4p04, but is also pertinent to subsequent versions 9.5 and 9.6, the latter one being current at the time of submission of this paper. In these latter versions the problem of not producing fluorescence emission in case of decays other than electron capture has been addressed and the handling of forbidden β -decays has been added, but all other features and problems of the implementation mentioned in this work are still valid.

A Unified Modeling Language (UML) [21] class diagram of the RDM code in Geant4 9.4p04 is shown in Fig. 1. It shows the cooperation between the different classes in the code. This diagram is supplemented by the activity diagram shown in Fig. 2. The two diagrams highlight problems [22] inherent in the code's design:

- Since each decay type is defined by distinctive physics, it would make sense to implement the individual types as separate objects. Instead, in the original Geant4 RDM package all decay types are implemented together in the

G4NuclearDecayChannel class. The decay type classes merely provide an interface to this class with decay type-dependent initialization parameters. This complicates unit tests of individual components. Additionally, a much larger amount of code has to be checked if, for instance, an error is found for one decay type.

- Whereas the decay physics for each type is distinct, the interface to each type is similar: all decay type objects should have a method to produce decay emission and a daughter nucleus. Such an interface could be provided by a common base class. In the existing design the *G4RadioactiveDecay* class needs to know the implementation and interface details of each individual decay type.
- Objects should have one specific responsibility, e.g. the interface to the data library. This is not the case: the *G4RadioactiveDecay* class is responsible for initializing and loading values from the data libraries, initializing the decay types and the variance reduction and decay chain handling. Such a design again enlarges the fraction of code which needs to be checked for a specific error or maintained for the update of a specific responsibility. Additionally, due to inadequate domain decomposition, two distinct responsibilities - the simulation of radioactive decay and event biasing - are mixed in the same class.
- In case of the β -Fermi-function implementation the interface is implementation-dependent. Changing the algorithm may thus also involve changing the interface — and as a result all other code parts depending on this class.

A new package, named RDM-extended, has been developed to address existing issues of Geant4 RDM, and to extend and improve the capabilities of radioactive decay simulation in a Geant4-based environment.

The software design of the RDM-extended package follows an object-oriented programming approach with clear responsibility definitions. For this design the relevant entities and requirements for radioactive decay physics have been identified. Decay type-dependent approaches have been considered alongside common tasks for all decay types. The design also takes into account that two sampling methods are to be handled within a common framework, and that the external classes the code depends on may be subject to interface changes.

The UML diagrams in Fig. 3 and 4 document that the responsibilities of objects are clearly defined and that functionality is neither duplicated nor aggregated into non-specialized classes.

An example for implementing functionality only once is the class *G4RandomDirection*, which provides functionality for sampling random particle momentum vectors to the classes modeling the individual decay types and the statistical sampling.

The *G4RadioactiveDecay* class of the original RDM (Fig. 1) is an example of a non-specialized class: it is responsible for physics simulation, data preparation and decay chain handling. In contrast the *G4RadioactiveDecay* class of the RDM-extended is a pure management class, which coordinates the interaction of specialized classes for the aforementioned tasks: the different emission classes (physics simulation), *G4RadioactiveController* (data-management) and

G4DecayChainSolver (decay chains).

The RDM-extended package also respects encapsulation rules. Furthermore, physics functionality implemented in the different classes can be combined as needed, thus allowing both sampling approaches to use a common code base for functionality required by both.

The addition of a statistical sampling approach is reflected in the activity diagram: the initial configuration and library data access are common for both design approaches. They differ after a decay channel is chosen: in the per-decay approach fluorescence emission is sampled for electron capture decays by delegating responsibility to the *G4AtomicDeexcitation* class. At a later point the deexcitation emission of the nucleus is sampled by using the *G4PhotonEvaporation* class. If the statistical approach is chosen instead, all photon emission and discrete electron emission processes are sampled simultaneously at the end of handling decay physics.

VIII. PER-DECAY SAMPLING

Per-decay sampling is already present in Geant4. This original RDM code provides the required functionality: therefore the general approaches to physical treatment and data handling were preserved, but refactored before their inclusion into the RDM-extended in order to conform to the design discussed in the previous section. This refactored per-decay sampling is consistent with the original RDM. Additionally, small errors in the physical treatment were addressed. As a result the structure of the code is different from the original one, but the inherent functionality is conserved.

Per-decay sampling is based on reprocessed ENSDF data both in the original RDM and the RDM-extended package. For the RDM-extended we reprocessed the data using a parser, which implements the ENSDF format as given in the ENSDF manual [7]. The Geant4 classes handling nuclear and atomic deexcitation in the per-decay approach use their own libraries. For the *G4PhotonEvaporation* class (nuclear deexcitation) this is ENSDF-based, with conversion electron probabilities compiled from [23]–[25]. The *G4AtomicDeexcitation* class (atomic deexcitation), which has been previously validated in [26] and [27], uses EADL [28] data. Concerns about the accuracy of the EADL data have been reported in [29].

The electron capture probabilities included in both radioactive decay libraries are not given in ENSDF directly. They need to be calculated using an additional data source, which gives fluorescence yields and binding energy information. For consistency reasons the atomic data file distributed with ENSDF is used for the RDM-extended package, which is based on data by Bearden and Burr and Bambynek *et al.* [15], [16]. This also facilitates comparisons with ENSDF-based online-databases such as NuDat, which use the same data.

Further refactoring is foreseen to use a package for atomic data management exploiting the results of [30] and an improved package for the simulation of atomic relaxation, which are currently under development. These packages are intended to satisfy requirements common to the simulation of radioactive decay and of electromagnetic interactions.

The radioactive decay database of both codes is supplied in plain text on a per isotope basis.



Fig. 1. A UML class diagram of the original RDM, which implements per-decay sampling, showing the relations between the individual classes. The colors (grey shades) reflect the responsibilities and are matched to the activity diagram in Fig. 2: physics simulation (blue/medium grey) and decay chain processing (purple/dark grey). Yellow/light grey objects can not be attributed to a single responsibility. Data retrieval, initialization and management is provided by the main class (*G4RadioactiveDecay*). The individual decay channels initialize a *G4NuclearDecayChannel* object which includes the physics of all decay types in one class.

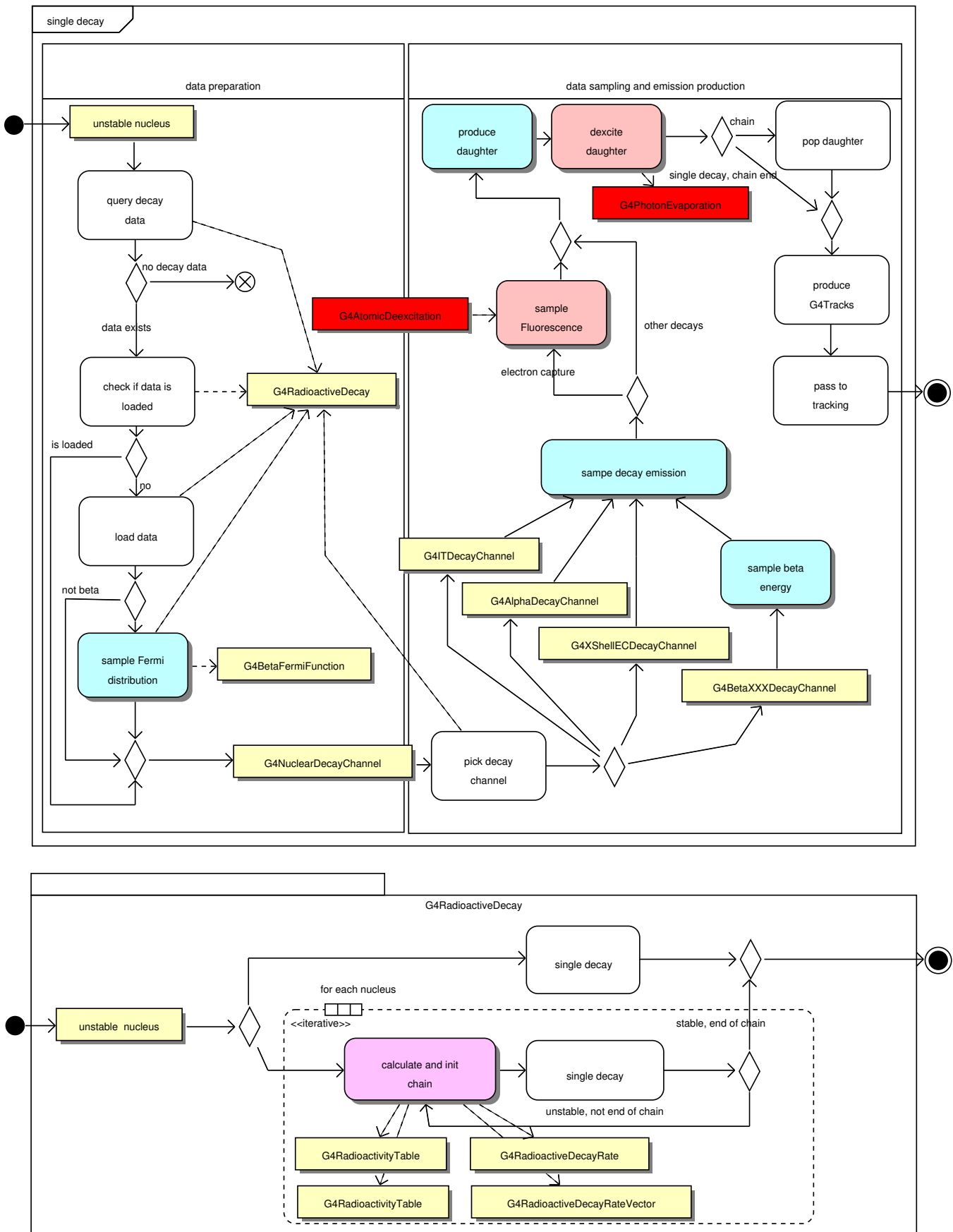


Fig. 2. A UML activity diagram of the original RDM, which implements per-decay sampling showing the workflow of the code when decaying a nucleus. The color (grey shade) scheme is matched to the class diagram in Fig. 1 and reflects different activity types: physics simulation within the radioactive decay code (blue/medium grey), physics simulation by other processes (red/darkest grey) and decay chain calculations (purple/dark grey). Yellow/light grey objects can not be attributed to a single responsibility.



Fig. 3. A UML class diagram for the RDM-extended package implementing a refactored per-decay sampling and a statistical sampling method. The colors (grey shades) reflect the responsibilities and are matched to the activity diagram in Fig. 4: physics simulation (blue/medium grey), data management and preparation (orange/medium dark grey) and decay chain processing (purple/dark grey). Each decay type class is responsible for simulating the type's characteristic physics. The deexcitation and fluorescence emission for the statistical simulation approach is produced by the *G4VDecayEmission* base class.

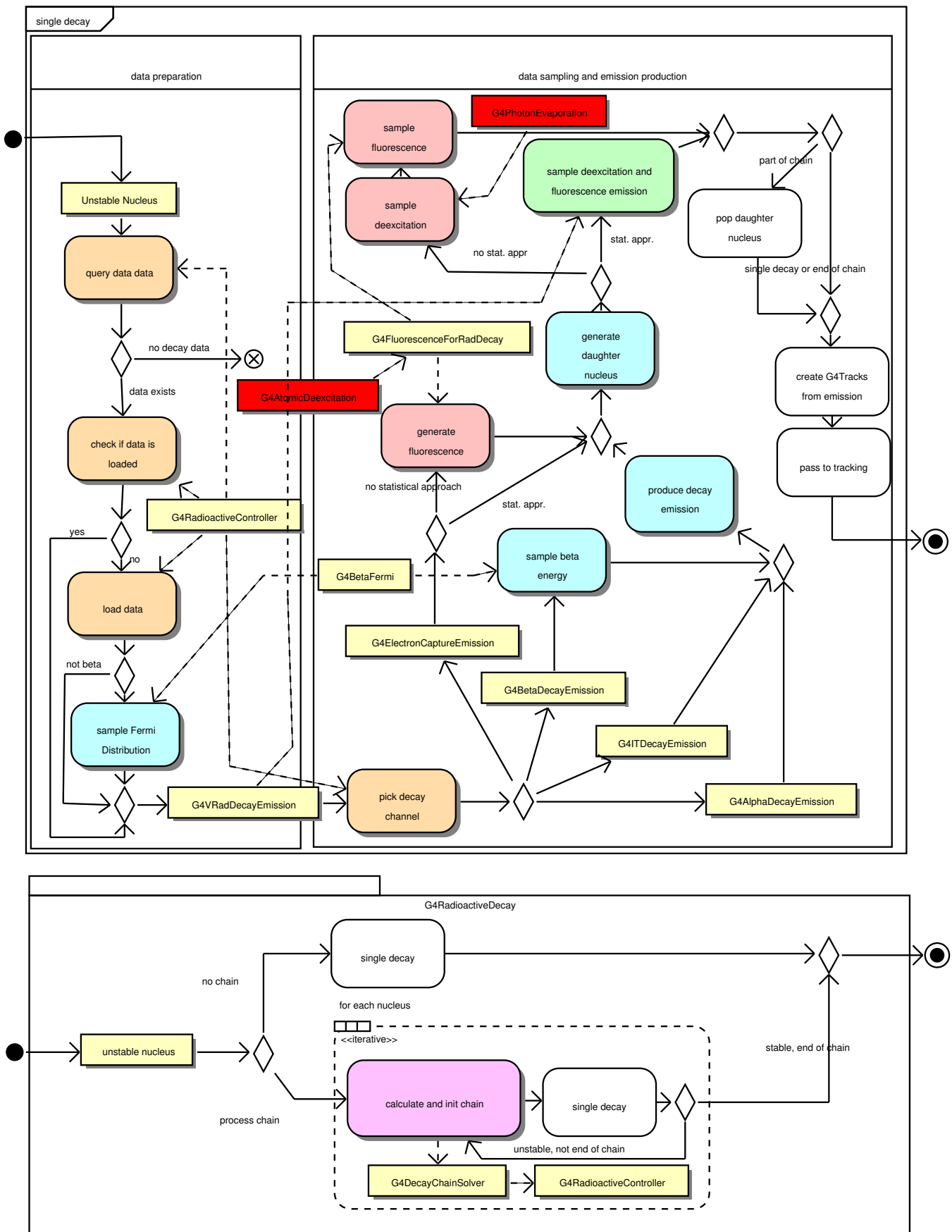


Fig. 4. A UML activity diagram for the RDM-extended package implementing a refactored per-decay sampling and a statistical sampling method, showing the workflow of the code when decaying a nucleus (top) and handling of decay chains (bottom). The color (grey shade) scheme is matched to the class diagram in Fig. 3 and reflects different activity types: physics simulation within the radioactive decay code (blue/medium grey), physics simulation by other processes (red/darkest grey), data management and preparation (orange/medium dark grey) and decay chain calculations (purple/dark grey). The emission produced by the statistical decay approach is highlighted in green/medium light grey.

A. Energy Conservation

For a physically correct outcome energy needs to be conserved for the decay, all decay emissions and when deexciting the daughter nucleus. Nuclear deexcitation is handled by the *G4PhotonEvaporation* class, which uses a reprocessed ENSDF-based data library different from the radioactive decay class. Since it is not guaranteed that the level-energies in the two data libraries are exactly the same, a possible deviation has to be taken into account. This is done in the original Geant4 RDM on three levels:

- If the level energy passed by the radioactive decay code does not correspond to a tabulated level in the *G4PhotonEvaporation* library, the nearest level present in the *G4PhotonEvaporation* database is used to retrieve the possible transitions. The initial deexcitation step occurs from the energy passed by the decay code and will transition to an energy tabulated in the deexcitation library. The energy of the first γ -ray will thus not be in accordance with the tabulated evaporation data, but deviate by the difference between the tabulated radioactive decay and evaporation library values. Further transitions will then result in emission at energies in accordance with the *G4PhotonEvaporation* data.
- For all transitions along a deexcitation chain it is checked if the level resulting from the transition is within a tolerance of 1 keV of the ground state. If this is the case, the photon energy is set to the energy of the excited level and the nucleus is deexcited to the ground state. Again energy is conserved, but the energy of the emitted photon may not be in accordance with the tabulated data.
- The final state energy is passed back to the radioactive decay code. In case this energy is not 0, but below 1 keV, the daughter's excitation energy is set to 0 keV. Here energy is not conserved. Otherwise the decay code outputs an excited (possibly meta-stable) daughter nucleus as defined by the data library.

The first two conservation treatments are necessary if slightly divergent data libraries exist. They are handled internally by the *G4PhotonEvaporation* class for all processes which delegate to it, and have thus not been altered as part of the radioactive decay code development. Rectifying these divergencies would require a consolidation of the Geant4 data libraries, which exceeds the scope of this work. The last approach to energy conservation is physically unjustifiable and accordingly has been corrected in the refactored implementation. In case of an energy mismatch, the energy of an excited nucleus is deposited in the geometrical volume it is located in. Whereas this does not address the library inconsistency, which leads to the energy mismatch, it does conserve energy and is accordingly seen as the preferable solution.

B. Momentum Conservation and Recoil of Nucleus

Like energy, momentum should be conserved during the complete decay simulation. In the per-decay approach nuclear deexcitation is delegated to *G4PhotonEvaporation*, which also generates the momenta of the γ -rays and conversion-electrons in a deexcitation cascade. The radioactive decay simulation

is then responsible to account for the resulting recoil of the daughter nucleus.

Similarly, the momenta of any fluorescence and Auger-emission are sampled by the *G4AtomicDeexcitation* class and should also be taken into account as vectorial quantities. This is not the case in the original Geant4 RDM code: here the difference of all produced emissions' summed kinetic energies and the binding energy of the innermost vacated shell are added to the kinetic energy of the daughter nucleus. Whereas the computation of scalar values might result in a marginal performance increase, it neglects that the emission is not unidirectional, but isotropic. A full vectorial treatment is the physically accurate solution and is thus implemented in the RDM-extended package.

IX. STATISTICAL SAMPLING

The statistical sampling approach discussed in this section is a novelty for Geant4-based radioactive decay simulation. The necessity for such an approach has become evident as a consequence of the experimental requirements identified in Section III. Statistical sampling in this context means that the full simulation of an individual decay is considered of lesser importance as long as the emission and nuclei produced by many decays on average lead to a physically correct result.

Because all decay emission is treated as independent of each other and the intensity of each emission is known, sampling generally reduces to the problem of repeated random number generation within the intensity range of $0 \dots 1$. Additionally, one has to take into account that intensities $I_{\text{em}} > 1$ may exist, if both nuclear and atomic deexcitation result in radiation at the same energy. In the latter case at least $I_{\text{em}} \pmod{1}$ emissions will be generated for every decay.

This approach does not require to take allowed or forbidden level transitions into account during nuclear deexcitation since the order of occurrence of transitions is considered irrelevant, as long as for a large enough number of transitions, each occurs with the correct probability, in result yielding the correct intensity.

These intensities are included in ENSDF alongside normalization information and as such all required decay information may be retrieved from a single consistent library. Accordingly, energy mismatches between libraries do not occur and no interdependencies between different physics classes exist. As a consequence, an alteration of tabulated library values will manifest itself in a straight-forward and immediate fashion in the simulated decay intensities and energies.

The conservation of energy and momentum is simplified, as the information of momenta and energy of all decay emissions is known to the radioactive decay simulation. Accordingly, it can be taken into account in a physically meaningful fashion.

In consequence, statistical sampling is a more efficient, straight-forward, ENSDF-consistent and performant approach for simulating radioactive decays for the majority of experimental applications which do not require knowledge of individual decays.

X. SAMPLING-METHOD-INDEPENDENT FUNCTIONALITY IN THE RDM-EXTENDED RADIOACTIVE DECAY CODE

In the previous two sections a distinction between the per-decay and statistical sampling approach was made. For a complete solution to radioactive decay physics, in accordance with the requirements, sampling schemes for β -emission and decay chains are necessary. These are approach-independent as they occur in both sampling scenarios.

A. β -Fermi-Function - Sampling of the Continuous β -Spectrum

β -decay is an unbound three-body decay and the emitted radiation will thus have a continuous spectrum. This spectrum can be sampled using the β -Fermi-function, which also takes corrections resulting from the interaction of the charged nucleus with the β -particle into account.

The parameters passed to the β -Fermi function should be physics-relevant, but code-independent. In particular this means that, instead of passing a binning scheme dependent energy, as is done in the original Geant4 RDM implementation, the physical parameters Z and endpoint energy E_0 , as well as the decay type (β^- or β^+) and optionally forbiddenness, are passed in the RDM-extended implementation. This parameter set is sufficient for many Fermi-function approximations, like to those summarized by Venkataramaiah *et al.* in [31]. Currently, the computationally most performant approximation given therein is used in the RDM-extended package for calculating the Fermi correction factor $F(Z, E)$ for an isotope with atomic number Z with the total energy E (given in MeV):

$$F(Z, E) = [A + B/(E - 1)]^{\frac{1}{2}}. \quad (1)$$

The constants A and B were determined by Venkataramaiah *et al.* through linear regression using the data from Rose [32], and were found to satisfy:

$$A = \begin{cases} 1 + a_0 \exp(b_0 Z) & \text{for } Z \geq 16 \text{ (2a)} \\ 7.3 \times 10^{-2} Z + 9.4 \times 10^{-1} & \text{for } Z < 16 \text{ (2b)} \end{cases}$$

with

$$\begin{aligned} a_0 &= 404.56 \times 10^{-3} \\ b_0 &= 73.184 \times 10^{-3} \end{aligned}$$

and

$$B = a Z \exp(bZ) \quad (3)$$

with

$$\begin{aligned} a &= \begin{cases} 5.5465 \times 10^{-3} & \text{for } Z \leq 56 \\ 1.2277 \times 10^{-3} & \text{for } Z > 56 \end{cases} \\ b &= \begin{cases} 76.929 \times 10^{-3} & \text{for } Z \leq 56 \\ 101.22 \times 10^{-3} & \text{for } Z > 56. \end{cases} \end{aligned}$$

The correction factor $F(Z, E)$ is then input into the β -Fermi function

$$N(p) dp = F(Z, E) p^2 (E_0 - E)^2 dp, \quad (6)$$

where E_0 is the end-point energy of the β -spectrum, obtained from the tabulated radioactive decay data library, E is the

total energy of a β -particle with momentum p . If a given parameter set is computed for the first time, a tabulated energy distribution is calculated, from which the β -particle energy is drawn. Future occurrences of the same parameter set will draw particle energies directly from this distribution, thereby minimizing processing time. Venkataramaiah *et al.* found this approximation to be accurate within a one-percent margin of error, when compared to the tabulated values of Rose [32]. Because the interface of the β -Fermi-function class is independent of the model implemented therein, it is easily possible for the user to substitute the RDM-extended's β -Fermi-function approximation, by one which has better accuracy for a certain isotope.

B. Decay Chains and Activation

Isotopes resulting from a radioactive decay are often unstable themselves. This leads to chains of subsequent decays until a stable daughter product is reached. Often it is desirable to only fully simulate those isotopes in a chain which are of interest as observables, either due to their half life or due to the radiation emitted when they decay. In such a case a full Monte-Carlo simulation of every decay is very inefficient. The sampling of interesting isotopes only requires determining the number of nuclei of a given species present in a chain at a specific time.

Additionally, radioactive isotopes may be created by nuclear activation, often as a result of proton or neutron collisions with a nucleus. While the collisions and resulting activation is handled by the hadronic processes of Geant4 [33], book-keeping of the activation buildup and the decay of the created unstable nuclei are tasks for the radioactive decay simulation. In the RDM-extended the activation simulation is also capable of altering the material composition on a per-volume basis to the isotope composition present at discrete user-defined time steps.

For a system of n nuclei, where the i th nuclei decays into the $(i + 1)$ th nuclei of the chain, this calculation can be done by solving a system of coupled differential equations with the general form:

$$\begin{aligned} \frac{dN_1(t)}{dt} &= -k_1 N_1(t) \\ \frac{dN_2(t)}{dt} &= \lambda_1 N_1(t) - k_2 N_2(t) \\ &\vdots \\ \frac{dN_n(t)}{dt} &= \lambda_{n-1} N_{n-1}(t) - k_n N_n(t) \end{aligned} \quad (7)$$

with N_i being the quantity of the i th nucleus at time t and $k_i = \lambda_i + \alpha_i$ containing information on the decay constant λ_i and a constant particle number dependent activation rate α_i .

A general approach for solving equation (7) for any number of products was first derived by Bateman [34] and the above equations have as such become known as the Bateman equations.

One of the main disadvantages of the original Bateman solution is that it does not consider branching, i.e. if a

parent nucleus can decay to different daughter nuclei via different decay types. Many computational algorithms using Bateman's approach or a derivative thereof exist, with the one implemented in the original radioactive decay code [6], being just one example which overcomes the branching limitation and includes activation. These algorithms are generally computationally expensive, in the sense that they require recursive loops for each nucleus in a decay chain, but may achieve good overall performance, if the calculations can be reused.

An alternative is the algebraic approach derived by M. Amaku *et al.* in [35], building upon work by R. J. Onega [36], D. Pressyanov [37], L. Moral and A. F. Pacheco [38], as well as T. M. Senkov [39] and D. Yuan and W. Kernan [40]. In this approach the properties governing the decay chain are written into a matrix in Hessenberg-form

$$\Lambda = \begin{bmatrix} -k_1 & 0 & 0 & 0 \\ k_1 & -k_2 & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & k_{n-1} & k_n \end{bmatrix} \quad (8)$$

and a vector $\mathbf{N}(t)$ with

$$\mathbf{N}(t) = \begin{bmatrix} N_1(t) \\ N_2(t) \\ \vdots \\ N_i(t) \end{bmatrix}. \quad (9)$$

The system of equations (7) may then be written as

$$\frac{d\mathbf{N}}{dt} = \Lambda \mathbf{N}. \quad (10)$$

This original algebraic approach was introduced by Onega in 1969 and extended by Yuan and Kernan as well as Semkov to include branching by modifying Λ to

$$\Lambda = \begin{bmatrix} \Lambda_{11} & & & & \\ \Lambda_{21} & \Lambda_{22} & & & \\ \vdots & \vdots & \ddots & & \\ \Lambda_{i1} & \Lambda_{i2} & \dots & \Lambda_{i,i} & \\ \vdots & \vdots & & \vdots & \ddots \\ \Lambda_{n1} & \Lambda_{n2} & \dots & \Lambda_{ni} & \Lambda_{nn} \end{bmatrix} \quad (11)$$

with

$$\Lambda_{ij} = k_{i-1} b_{ij} \quad (12)$$

for $i > j$ and

$$\Lambda_{ii} = -k_i. \quad (13)$$

In equation (12) b_{ij} denotes the branching ratio from the j th to the i th component of the decay chain with $\sum_{i=j+1}^n b_{ij} = 1$. In computational practice Λ can be easily constructed by iterating through the decay chain and taking activation rates into account, if necessary. Because Λ is independent of the actual nuclei numbers, it must only be constructed once for each set of nuclei and activations characterizing a given chain. Calculations using different nuclei numbers can reuse these initial matrices as needed. Using the above definitions, the number of nuclei of each species in the decay chain is then given by

$$\mathbf{N}(t) = e^{\Lambda t} \mathbf{N}(0) \quad (14)$$

which can be rewritten to

$$\mathbf{N}(t) = C e^{\Lambda_d t} C^{-1} \quad (15)$$

as described by Onega. In equation (15) C is a square matrix with the n th column consisting of the n th eigenvector of Λ , so that $C = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_n]$. C^{-1} is its inverse and Λ_d a diagonal matrix with the elements $\Lambda_{d,nn}$ being the n th eigenvalue of Λ .

M. Amaku *et al.* then derive an algorithmic approach for calculating the matrices C , C^{-1} and Λ_d , which is computationally less expensive and more accurate than a general approach of numerically calculating the matrix elements. The elements of $C = [c_{ij}]$ can be calculated with the recurrence expression

$$c_{ij} = \frac{\sum_{k=j}^{i-1} \Lambda_{ik} c_{kj}}{\Lambda_{jj} - \Lambda_{ii}} \quad (16)$$

for $i = 2, \dots, n$, $j = 1, \dots, i-1$ and $c_{jj} = 1$. Similarly the elements of $C^{-1} = [c_{ij}^{-1}]$ are given by

$$c_{ij}^{-1} = \sum_{k=j}^{i-1} c_{ik} b_{kj} \quad (17)$$

for $i > j$ and $b_{jj} = 1$. Using C and C^{-1} , Λ_d is given by

$$\Lambda_d = C^{-1} \Lambda C \quad (18)$$

resulting in a general form for 15 of

$$\mathbf{N}(t) = C \begin{bmatrix} e^{-\Lambda_{d,11}t} & & & \\ & e^{-\Lambda_{d,22}t} & & \\ & & \ddots & \\ & & & e^{-\Lambda_{d,nn}t} \end{bmatrix} C^{-1} \mathbf{N}(0). \quad (19)$$

In the RDM-extended code the replenishment and generation of unstable isotopes through activation is calculated using the above equations on a per-volume level. In a first simulation run, a bookkeeping class *G4ActivationBookkeeping* keeps track of all activations occurring in a given volume and calculates the new material and isotope composition of the volume at a given time. In a second run, the material properties are then updated within the geometry, and the volumes are set as radioactive background sources, with the calculated activity via the *G4RadDecayVolumeBookkeeping* class. This class samples the radioactive background emission, which should be generated alongside a primary particle and pushes it on the event stack. In this way minimal user intervention is required.

XI. VERIFICATION OF SAMPLING METHODS

The radioactive code implementations have been verified for consistency with ENSDF data. Since ENSDF is established in the experimental community as an authoritative reference, comparison of Monte Carlo models against it provides valuable information for the users of these codes. The results presented here are chosen to highlight the physics-performance improvements of the statistical sampling in the RDM-extended package when compared to the per-decay sampling of the

original Geant4 RDM code. Accordingly, the refactored per-decay code, which has been verified to produce equivalent results to the approach used in the original Geant4 RDM, is not detailed further.

In addition to the γ -ray and conversion electron emission directly given in ENSDF, the verification of intensities and energies of fluorescence and Auger-emission with the Bambynek *et al.* data [16] has been included.

A. Simulated Data Production

The Geant4 simulations consisted of 10^6 decays of an unstable nucleus in an otherwise empty geometry. Any resulting unstable daughter nuclei were not decayed further. The kinetic energy of radiation and particles resulting from these decays was recorded separately for each radiation type (α , β , γ , non- β electrons) into two histograms: the first histogram ranging from 0–30 keV with a binning size of 0.05 keV, the second histogram ranging from 0–30 MeV with a binning size of 0.2 keV. The two binning schemes were chosen in order to properly distinguish discrete radiation in the X-ray and γ -energy regime.

Using the above approach, data for 2910 parent excitation levels of isotopes were simulated using the original RDM code with radioactive decay database version 3.3 and the per-decay simulation. The statistical approach simulations included 3040 parent isotopes and excitation levels, i.e. 130 more than the sample used for the production with per-decay sampling. The numbers differ due the different versions of ENSDF data and different parsers used for reprocessing. For both sets of simulations β -emission was distinguished from Auger- and conversion electron emission.

B. Evaluated Data Preparation

The verification data were extracted from ENSDF into a tabulated form suitable for further analysis using the RADLIST [14] program. By using this ENSDF-supplied parser it was assured that no errors were introduced into the evaluated data during the automated extraction.

C. Data analysis

The automated, comparative analysis takes the level and particle energies from the evaluated data as an initial energy estimate $E_{0,\text{level}}$ or $E_{0,\text{particle}}$ for where emission may be present in the simulated data. The simulated data histograms are then scanned in an interval of $E_0 \pm 0.5$ keV at energies below 30 keV and $E_0 \pm 5.0$ keV at higher energies for emission events. Data outside these windows was considered not to belong to the currently compared emission energy.

In case of discrete emission contained in a single bin the intensity of the emission is given by

$$\text{Intensity} = \frac{\text{Events in bin}}{\text{Number of simulated decays}}. \quad (20)$$

The energy uncertainty is determined by the bin size. The intensity uncertainty is $\Delta I = \sqrt{N}/N_{\text{sim}}$ with the event number N per bin and the simulated number of decays N_{sim} .

Should the emission be distributed into (multiple) neighboring bins, which is possible due to recoil from previous emissions transferred onto the emitting nucleus, the intensity is calculated from the total number of the events in these bins. The evaluated data can indicate that emissions at a different energy, but in proximity to the energy currently being processed may have also contributed to the number of simulated events. In this case the summed intensities of all emissions in neighboring bins are used for further comparisons. The energy position of these emissions is then set to the median energy of the events, accordingly its uncertainty is the median deviation.

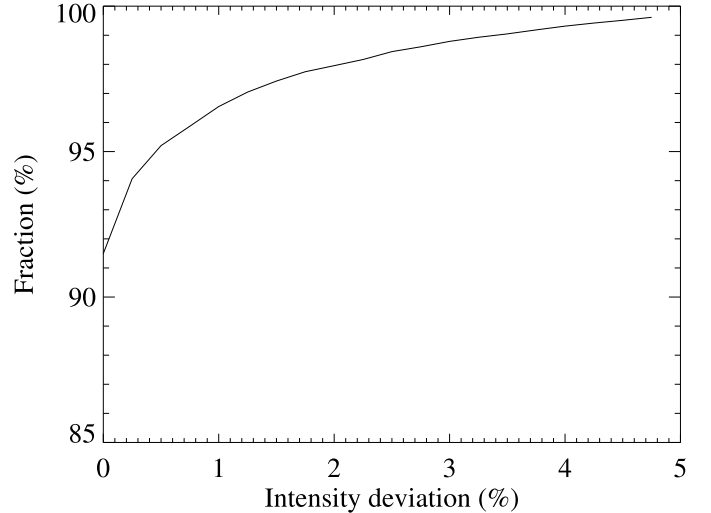


Fig. 5. Percentage of simulated emission within a given deviation from the simulation input.

These steps provide energy and intensities of the radiation from the evaluated and simulated data. Absolute and relative energy and intensity deviations can then be calculated. In this way overall comparisons (see Section XII) and comparisons of individual isotopes, such as those measured in [8], are possible.

In order to assess the consistency of the simulation model, a complete set of simulations using the statistical approach was compared to the simulations' input data. Ideally, one would not expect any energy deviations at all and the intensity deviations should be compatible with the statistical error of the simulation. In practice the simulations' output is binned spectra, in order to keep total data amounts at a manageable level. Due to this, in rare cases the emission at nearby energies may not be distinguished properly, which in turn leads to intensity deviations. As is apparent from Fig. 5, 99% of all simulated emissions are within an intensity deviation of 3% of the simulation input.

XII. RESULTS: CONSISTENCY OF THE RADIOACTIVE DECAY CODES WITH ENSDF

In the following we focus upon identifying global trends and exploring the regions in which the radioactive decay simulations gives reliable results. A validation with measured data can be found in [8].

For the comparisons presented in the following, knowledge of the comparisons' uncertainties is important. Fig. 6 shows

a compilation of the intensity deviation uncertainties, which depend on the uncertainty of the evaluated intensity for a given level and the statistical uncertainty of the simulated data. As is apparent from the figure, even at low intensities the uncertainties are in a range below 10% for 99% of the data points.

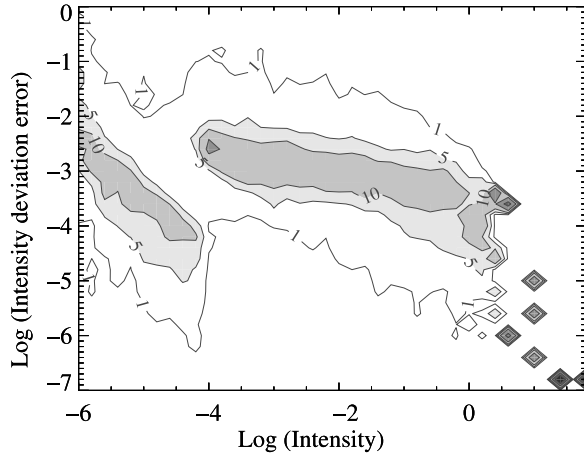


Fig. 6. Distribution of the relative error of the observed deviations between simulated and evaluated radiation intensities with respect to the level intensity. The contour levels correspond to the percentile of values at given deviation error with respect to the number of values at a given level intensity. At higher intensities values are more sparse resulting in the observed “single box” contours.

A. Intensity Deviations

In order to help readers quickly identify the simulations’ intensity discrepancy for isotopes occurring in their simulation, it was chosen to display the results as nuclide charts (a colored online version is available).

Fig. 7 shows a comparison of the intensity discrepancy of the original Geant4 RDM per-decay sampling alongside the discrepancy of the statistical approach of the RDM-extended when compared to ENSDF data. As is apparent from the figures, both sampling methods reproduce the γ - and α -emission intensities given in ENSDF within a few percent deviation for the majority of isotopes. Specifically, the mean deviations $(I_{\text{exp}} - I_{\text{sim}})/I_{\text{exp}}$ amount to $(8.57 \pm 2.22)\%$ for γ -rays and $(2.47 \pm 2.26)\%$ for α -emission when using the per-decay code.

For the statistical approach the deviations are minimal, with $(1.85 \pm 2.07)\%$ for γ -rays and $(5.61 \pm 2.62)\%$ for α -emission. The outliers ($> 50\%$ discrepancy) in the simulation using the RDM-extended package can be explained by the way the ENSDF data are parsed into the radioactive data library. In cases where multiple datasets describing the same emission exist, the implemented parser is tuned to pick experimentally determined data or, if such a distinction is not possible, use the first data set. The RADLIST program preferably uses theoretical data according to the documentation [14].

It is further apparent from Fig. 7 that the original per-decay code does not reproduce the ENSDF intensities of conversion

electrons well. This manifests itself in a mean deviation of $(35.67 \pm 6.32)\%$ compared to $(9.54 \pm 0.98)\%$ for the statistical approach.

For the original Geant4 RDM per-decay code the γ - and conversion electron intensity deviations must be attributed to the *G4PhotonEvaporation* model or more specifically its underlying data library [26]. A comprehensive verification and validation of this process would be beyond the scope of this paper and was thus not undertaken.

For fluorescence and Auger emission the deviation between evaluated ENSDF intensities and those produced by the original RDM per-decay approach are even larger, as is apparent from Fig. 8. For X-ray emissions the deviation amounts to $(52.64 \pm 1.97)\%$; for Auger-electron to $(52.57 \pm 0.96)\%$. Again not the radioactive decay code alone is responsible for these offsets but its interplay with the *G4AtomicDeexcitation* class and its associated EADL data library. It is interesting to note that the deviations are largest for the isotope on the left of the nuclide chart, i.e. isotopes which decay via electron capture. This substantiates the conclusion that the EADL data library is the source of the deviations, as it is only called by the per-decay approach if electron capture decays need to be sampled (in Geant4 9.4, changed in Geant4 9.5). Again the extended-RDM’s statistical method yields results with smaller deviations, which amount to $(4.09 \pm 1.78)\%$ for X-ray emission and $(1.34 \pm 1.16)\%$ for Auger electron emission. This is a more than 10-fold improvement in intensity consistency with respect to the per-decay approach of the original Geant4 RDM. A similar magnitude of deviations is reported in [29] for EADL data, in which different binding energy data libraries were compared with reference data. In this work EADL consistently showed the largest deviations.

It should be stressed that the observed intensity deviations for the per-decay approach result mainly from the incoherence of the data libraries involved in the sampling of nuclear deexcitation and fluorescence emission with the ENSDF-database, from which the reference data were derived. For the statistical approach the deviations are naturally smaller, as all radioactive decay data are derived from a single ENSDF-library and supplementary atomic data files.

B. Energy deviations

Most application scenarios will depend on the correct sampling of the energy of the γ - or X-ray radiation and that α particles as well as Auger- and conversion-electrons are emitted at the experimentally determined energies. From the comparisons shown in Fig. 9 and Fig. 10, one can conclude that at energies in the X-ray and Auger-electron range less than 30 keV, a deviation of less than 0.2 keV with respect to ENSDF data is to be expected for all radiation, when using the original Geant4 RDM per-decay method. Statistical sampling is again more consistent with ENSDF data. Here the observed energy deviation is less than 0.1 keV for the majority of emissions. For γ -radiation, α -particles and conversion electrons at higher energies, deviations of less than 0.5 keV are observed with the per-decay approach. The statistical approach shows deviations at or less than the bin size of 0.2 keV.

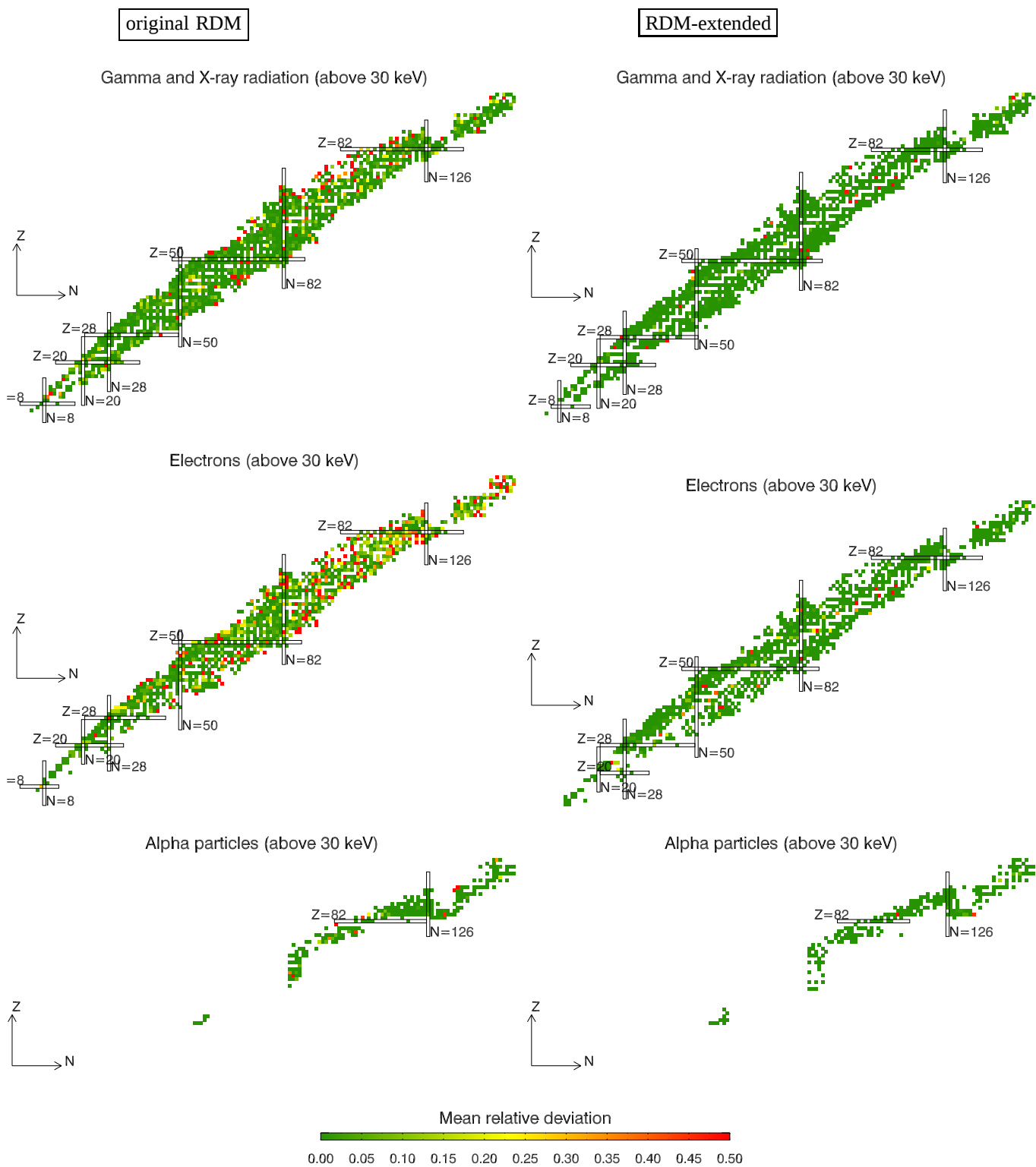


Fig. 7. Nuclide charts showing the median relative intensity deviations per isotope for γ - (top), conversion-electron (middle) and α -emission (bottom). Simulations using the per-decay approach of the original Geant4 RDM are shown on the left; simulations using the RDM-extended statistical sampling method are shown on the right.

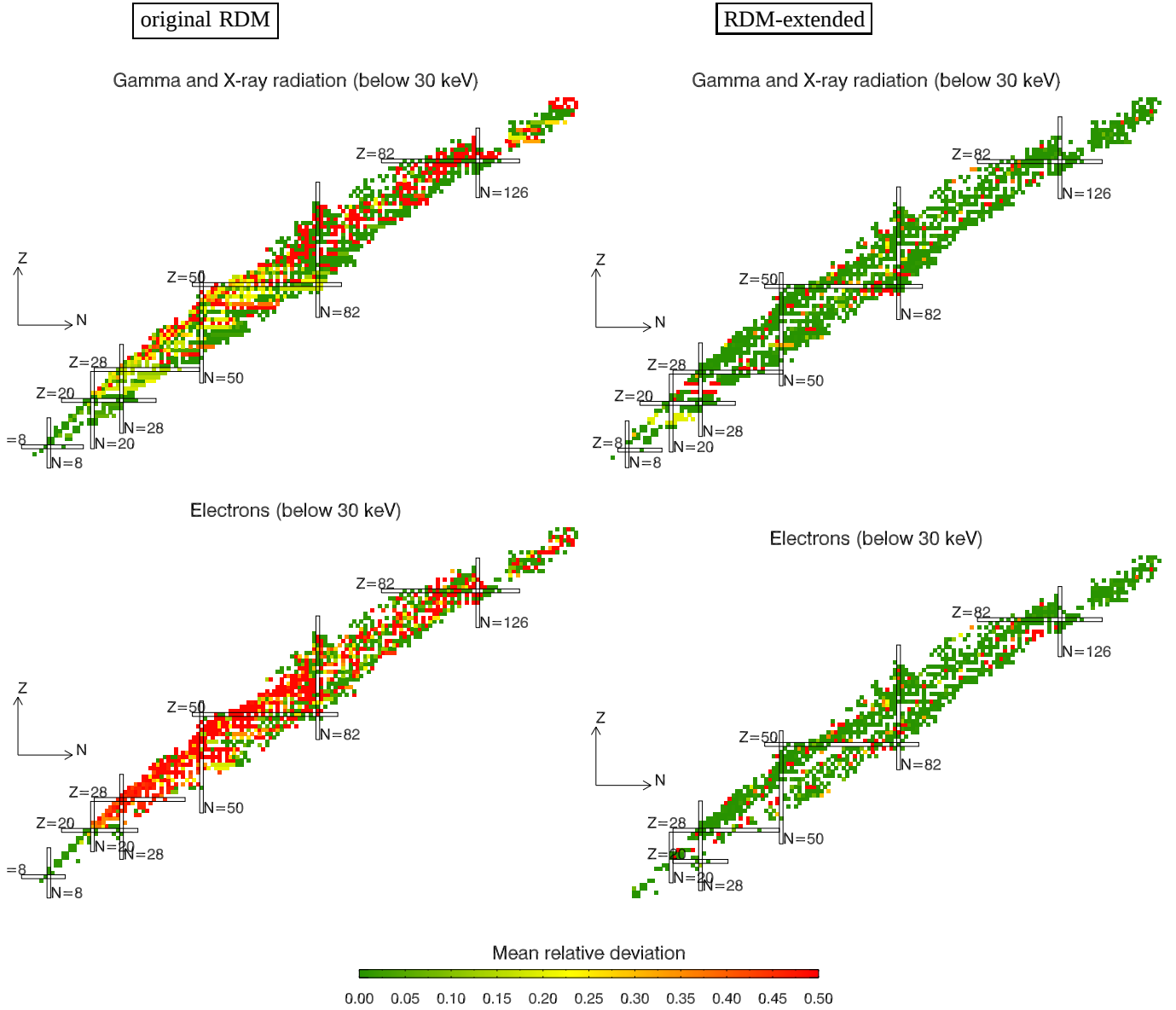


Fig. 8. Nuclide charts showing the median relative intensity deviations per isotope for X-ray (top) and Auger-electron emission (bottom). Simulations using the per-decay approach of the original Geant4 RDM are shown on the left; simulations using the RDM-extended statistical sampling method are shown on the right.

XIII. COMPUTATIONAL PERFORMANCE

Computational performance is a significant aspect for large scale Monte-Carlo simulations. The RDM-extended package was designed keeping this in mind (Section X-B). The implementation reflects this performance optimization: for instance, it uses hash-maps (`std::unordered_map`) from the C++11-standard [41], which allow element access times which scale with the number of elements in $O(1)$, instead of $O(\log(n))$, as would be the case for traditional `std::map` containers (see e.g. [42]). In order to estimate the performance gain achievable using this implementation in comparison to the original RDM, the following performance tests were undertaken:

- Decay an isotope 100 000 times using the refactored per-decay and the statistical approach.
- Decay a chain of isotopes and retrieve all emission which

has occurred from all isotopes in the chain within a sampled time period of Δt , which could be, for instance, an experiment's measurement duration, 10 000 times.

- Decay a chain of isotopes and retrieve the emission from a single isotope within the chain occurring within a time period Δt 10 000 times.

Each of the above tests was repeated 200 times to ensure that temporary CPU load from the operating system or other processes would not bias the results. For the first test the isotopes $^{22}\text{Na}(\beta^+)$, $^{60}\text{Co}(\beta^-)$, $^{229}\text{Th}(\alpha)$ and $^{133}\text{Ba}(\text{EC})$ were decayed, constituting an example for each of the particle emitting decay types. As an example of a full decay chain, the ^{233}U decay chain shown in Fig. 11 was simulated. In order to estimate the performance for different decay chain lengths, different initial nuclei in this chain were chosen.

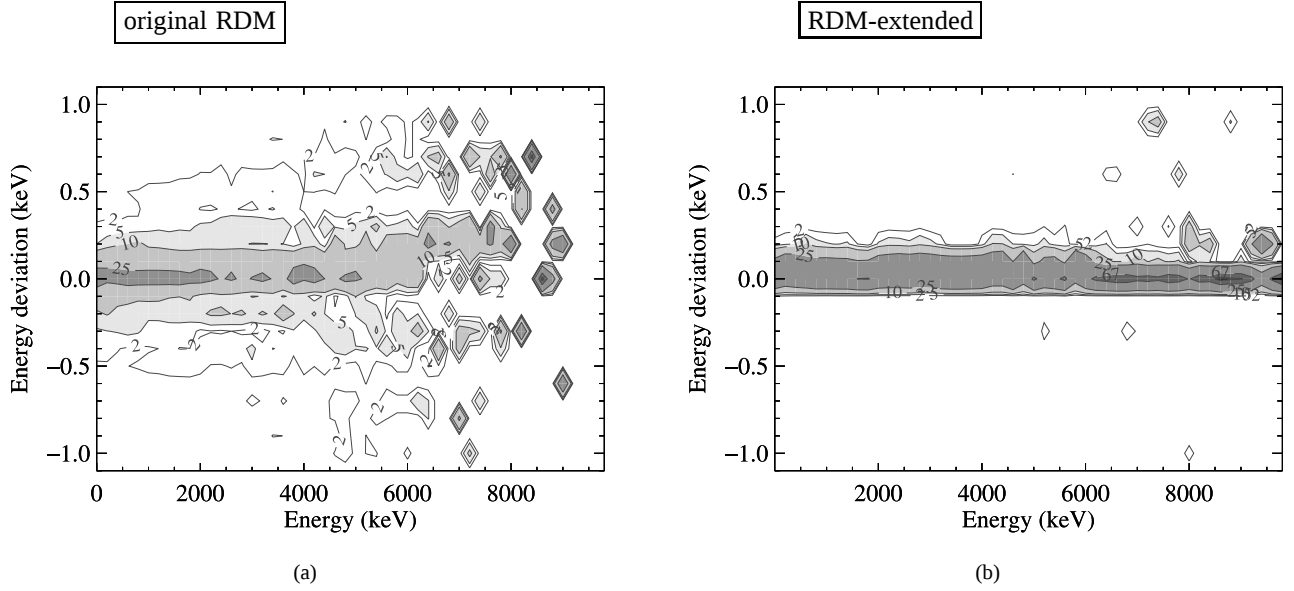


Fig. 9. Distribution of absolute energy deviations with respect to the radiation energy for all level energies. The contour levels correspond to the percentile of values at a given energy deviation with respect to the number of values at a given radiation energy. At higher energies evaluated values are more sparse resulting in fractured contours. Simulations using the per-decay approach of the original Geant4 RDM are shown on the left; simulations using the RDM-extended statistical approach are shown on the right.

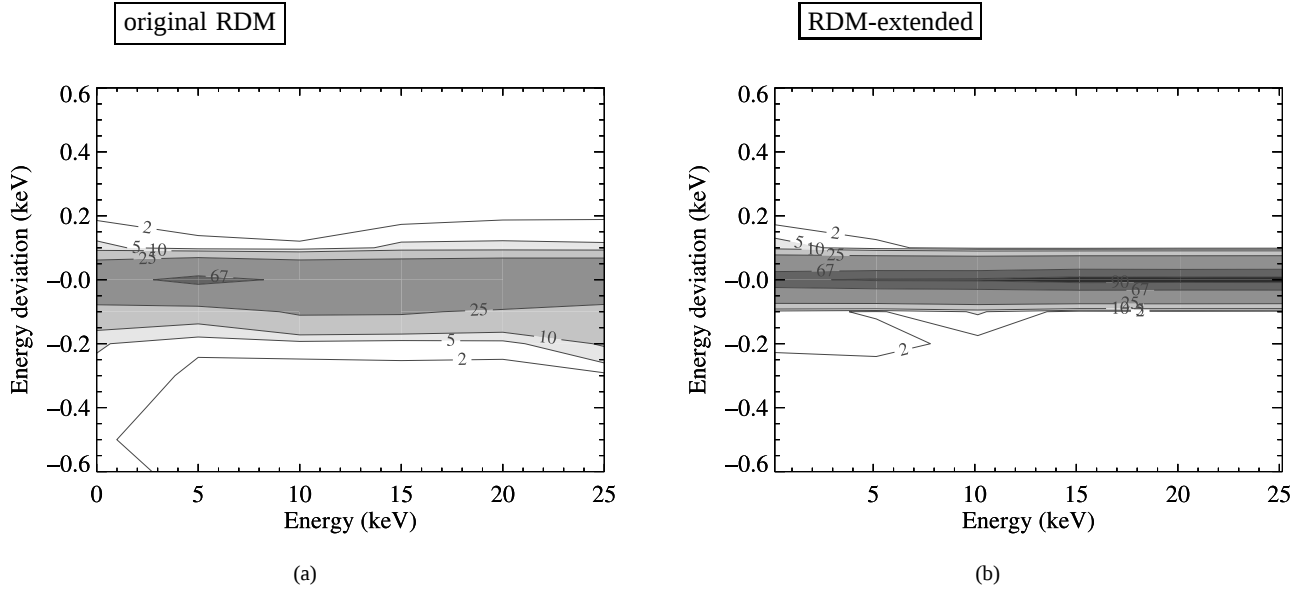


Fig. 10. Distribution of absolute energy deviation with respect to the radiation energy for level energies below 25 keV. The contour levels correspond to the percentile of values at a given energy deviation with respect to the number of values at a given radiation energy. Simulations using the per-decay approach of the original Geant4 RDM are shown on the left; simulations using the RDM-extended statistical approach are shown on the right.

All performance tests were done on a 12-core XEON machine at 2.93 GHz running Ubuntu 10.10 *Maverick Meerkat* and an identical application based on Geant4 9.4p04, which was unaltered except for the decay code. The *gcc* 4.4.5 compiler with Geant4 standard compiler flags and the extensions of the C++11-standard enabled was used for compilation. The radioactive decay was the only physics simulation process included in the test environment, thus guaranteeing that no other processes, which the involved particles might be subject to during further tracking, would influence the results.

The relative performance of the RDM-extended statistical sampling method shown in Fig. 12 asserts that this method is

– in almost all cases more than 20% – faster than the per-decay approach implemented in the original Geant4 RDM. An exception are isotopes which have a large number of deexcitation emissions such as ^{229}Th . Here the linear increase of sampling time with the number of emissions results in a performance penalty.

When comparing the refactored per-decay approach decay emission production, i.e. delegation to *G4PhotonEvaporation* and *G4AtomicDeexcitation*, two scenarios have been distinguished. In the first case, labeled “per-decay” in Fig. 12, the vacant shell index, which may be output by the *G4PhotonEvaporation* class at the end of deexcitation, is

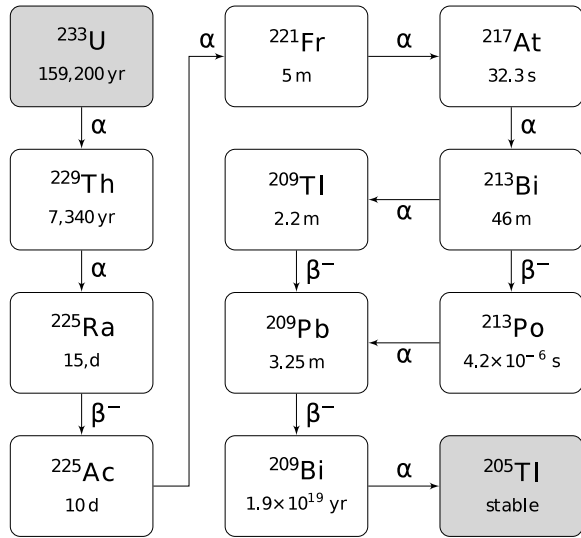


Fig. 11. The ^{233}U decay chain simulated for performance testing. The initial isotope of the chain was varied from ^{233}U to ^{213}Bi to test different chain lengths.

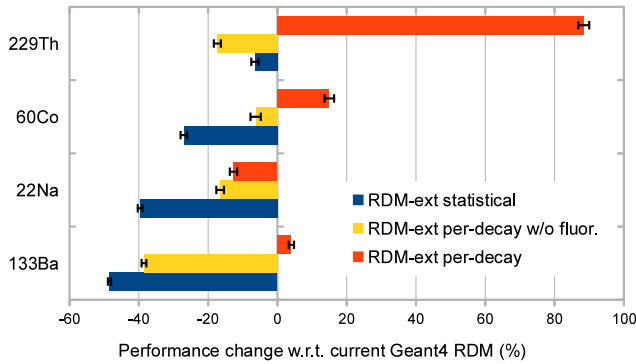


Fig. 12. Computational performance change when using the RDM-extended package with the statistical approach in comparison to the original Geant4 RDM and the per-decay approach. Each simulation consisted of 100 000 decays and was repeated 200 times. Three different sampling techniques are shown: statistical approach, classical approach and classical approach without post-decay fluorescence production. This last approach resembles the original Geant4 RDM implementation.

passed to *G4AtomicDeexcitation* for X-ray and Auger-electron production. The original Geant4 RDM code does not pass the atomic shell index, a scenario also simulated with the RDM-extended package and labeled "per-decay w/o fluor." in Fig. 12. It is apparent from the figure, especially for ^{229}Th , that the inclusion of *G4AtomicDeexcitation* results in a severe performance penalty. If a full treatment of X-ray and Auger-electron emission is needed and simulation performance is critical, it is thus recommended to use statistical sampling, if the applications scenario permits it.

For the decay chain performance comparison shown in Fig. 13 one has to consider that the per-decay approach of the original Geant4 RDM does not take the duration of the time period to be sampled into account. Instead it is up to the user to filter for the relevant emission according to the application scenario. This explains the increase in computing

time needed by the RDM-extended between ^{233}U as the initial isotope and ^{229}Th . As is shown in Fig. 11, thorium has a much shorter half life time than uranium. Regardless of which isotope is taken as the initial one in the chain, the simulated time duration for which decays are sampled stays the same at $3 \times 10^{13} \text{ s} = 95120 \text{ yr}$. During this time, much less uranium than thorium will have decayed. Accordingly, when uranium is chosen as an initial isotope, less emission has to be sampled, than is the case for thorium. This reflects itself in the performance increase of $\sim 55\%$.

Similarly, the statistical sampling in the RDM-extended package is faster, if only the emission from selected isotopes in the chain is of interest. In the RDM-extended code only the emissions from these isotopes are actually sampled and passed to tracking, reducing the number of particles which need to be simulated. In the original RDM the user has to filter for the emission of interest after it has been produced by the decay code and has been passed to tracking. This is inefficient and computing time intensive, as can be seen from Fig. 13. An extreme case for such a scenario is when only the emission of the final isotope in a chain is of interest. This case is documented in Fig. 13 by the two "end of chain" data values.

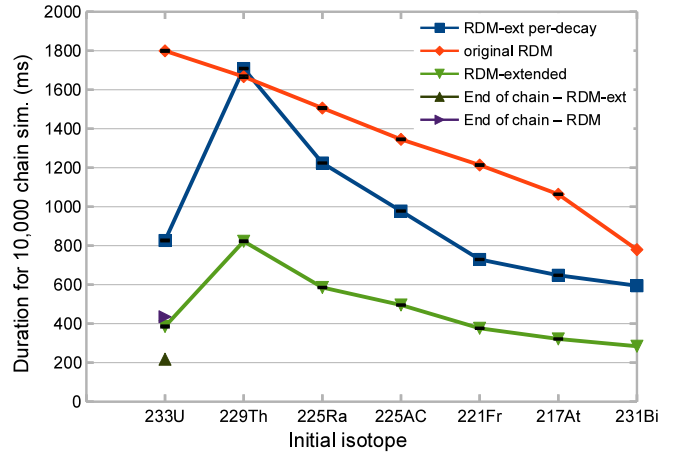


Fig. 13. Computational performance of the RDM-extended package and the original RDM when decaying the ^{233}U decay chain. The chain length was varied by setting different initial nuclei. The two data points labeled end of chain are the performance values for when only the emission of the last isotope in the chain is of interest.

XIV. CONCLUSION

Experimental requirements concerning radioactive decay simulation have been analyzed and evaluated against radioactive decay models and functionality available in present Monte-Carlo codes. It was found that none of the available codes offers the possibility to correctly simulate physics on the per-decay level and to correctly simulate the statistical outcome of many decays without unnecessary overhead within one framework.

A software package, which addresses these requirements and allows simulations using both approaches, has been designed, implemented and verified with respect to the established reference of the ENSDF evaluated data library. The

RDM-extended package described in this paper reproduces the functionality of the original Geant4 RDM package implementing per-decay sampling, although with improved software design and generally faster computational performance. In addition, it encompasses functionality for statistical sampling of radioactive decays: with respect to the per-decay approach of the pre-existing Geant4 RDM, this approach has been verified to achieve better consistency with ENSDF data, and better computational performance. Significant consistency improvements have been verified especially in the X-ray regime.

The RDM-extended package can be used transparently in Geant4-based simulation applications. Its experimental validation is documented in a distinct dedicated paper [8].

ACKNOWLEDGMENT

The authors would like to acknowledge the financial support from the Deutsche Zentrum fuer Luft- und Raumfahrt (DLR) under Grant number 50QR902 and 50Q1102.

REFERENCES

- [1] S. Agostinelli *et al.*, “Geant4 – a simulation toolkit,” *Nucl. Instrum. Methods.*, vol. 506, no. 3, pp. 250 – 303, 2003.
- [2] J. Allison *et al.*, “Geant4 developments and applications,” *IEEE Trans. Nucl. Sci.*, vol. 53, pp. 270–278, 2006.
- [3] J. F. Briesmeister, *MCNP : a general Monte Carlo code for neutron and photon transport*. Los Alamos National Laboratory. Los Alamos, 1986, no. 3.
- [4] J. S. Hendricks, “MCNPX version 2.5.c,” ser. American Nuclear Society, 2003.
- [5] A. Fassò, A. Ferrari, P. R. Sala, and J. Ranft, “FLUKA: Status and Prospects for Hadronic Applications,” in *Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications*, F. B. A. Kling, Ed., 2001, p. 955.
- [6] P. Truscott, “Treatment of Radioactive Decay in Geant4,” *Qinetiq. Tech. Rep.*, 2002.
- [7] J. Tuli, “Evaluated Nuclear Structure Data File (ENSDF),” Dec. 2010. [Online]. Available: <http://www.nndc.bnl.gov/ensdf/>
- [8] S. Hauf *et al.*, “Validation of Geant4-based Radioactive Decay Simulation,” *IEEE Trans. Nucl. Sci.*, 2013, to be published.
- [9] “Systems and software engineering – life cycle processes –requirements engineering,” *ISO/IEC/IEEE 29148:2011(E)*, pp. 1–94, 2011.
- [10] O. Schwerer, “Nuclear Data Libraries and Services Provided by the IAEA,” Sep. 2001.
- [11] A. A. Sonzogni, “NuDat 2.0: Nuclear Structure and Decay Data on the Internet,” *AIP Conference Proceedings*, vol. 769, no. 1, pp. 574–577, 2005.
- [12] IAEA, “CINDA 2000, The Index to Literature and Computer Files on Microscopic Neutron Data,” Nov. 2000.
- [13] B. Pritychenko, E. Btk, M. Kellett, B. Singh, and J. Totans, “The Nuclear Science References (NSR) database and Web Retrieval System,” *Nucl. Instrum. Methods.*, vol. 640, no. 1, pp. 213 – 218, 2011.
- [14] T. W. Burrows, “The Program RADLST,” Brookhaven National Laboratory, Tech. Rep. BNL-NSC-52142, 1988.
- [15] J. A. Bearden and A. F. Burr, “Reevaluation of X-Ray Atomic Energy Levels,” *Rev. Mod. Phys.*, vol. 39, pp. 125–142, Jan. 1967.
- [16] W. Bambynek *et al.*, “X-Ray Fluorescence Yields, Auger, and Coster-Kronig Transition Probabilities,” *Rev. Mod. Phys.*, vol. 44, pp. 716–813, 1972.
- [17] A. G. Croff, “ORIGEN2: A versatile computer code for calculating the nuclide compositions and characteristics of nuclear materials,” *Nucl. Technol.*, vol. 62, no. 3, 1983.
- [18] W. B. Wilson, T. R. England, D. C. George, D. W. Muir, and P. G. Young, “Recent development of the CINDER’90 transmutation code and data library for actinide transmutation studies,” ser. Global ’95, Versailles (France), 1995.
- [19] M.-L. Giacri-Mauborgne *et al.*, “Status of the Photonuclear Data Library for CINDER’90,” in *International Conference on Nuclear Data for Science and Technology*, ser. American Institute of Physics Conference Series, vol. 769, 2005, pp. 195–198.
- [20] P. Truscott *et al.*, “Geant4-a new monte carlo toolkit for simulating space radiation shielding and effects,” in *Radiation Effects Data Workshop, 2000*, 2000, pp. 147 –152.
- [21] J. Rumbaugh, I. Jacobson, and G. Booch, *Unified Modeling Language Reference Manual, The (2nd Edition)*. Pearson Higher Education, 2004.
- [22] M. Fowler, *Refactoring: improving the design of existing code*, ser. Addison-Wesley Object Technology Series. Reading, MA: Addison-Wesley, 1999.
- [23] I. Band, M. Trzhaskovskaya, and M. Listengarten, “Internal conversion coefficients for atomic numbers $Z \leq 30$,” *Atomic Data and Nuclear Data Tables*, vol. 18, no. 5, pp. 433 – 457, 1976.
- [24] F. Rösel, H. M. Fries, K. Alder, and H. C. Pauli, “Internal conversion coefficients for all atomic shells,” *Atomic Data and Nuclear Data Tables*, vol. 21, no. 2-3, pp. 91 – 289, 1978.
- [25] R. Hager and E. Seltzer, “Internal conversion tables part I: K-, L-, M-shell conversion coefficients for $Z=30$ to $Z=103$,” *Nuclear Data Sheets. Section A*, vol. 4, no. 1-2, pp. 1 – 11, 1968.
- [26] S. Guatelli, A. Mantero, B. Mascialino, M. Pia, and V. Zampichelli, “Validation of Geant4 Atomic Relaxation Against the NIST Physical Reference Data,” *Nuclear Science, IEEE Transactions on*, vol. 54, no. 3, pp. 594 –603, Jun. 2007.
- [27] M. G. Pia, P. Saracco, and M. Sudhakar, “Validation of k and l shell radiative transition probability calculations,” *IEEE Transactions on Nuclear Science*, vol. 56, pp. 3650–3661, 2009.
- [28] S. T. Perkins, M. H. Chen, D. E. Cullen, and J. H. Hubbell, *Tables and graphs of atomic subshell and relaxation data derived from the LLNL Evaluated Atomic Data Library (EADL), Z=1-100*. Livermore, CA: Lawrence Livermore Nat. Lab., 1991.
- [29] M. G. Pia, H. Seo, M. Batic, M. Begalli, C. H. Kim, L. Quintieri, and P. Saracco, “Evaluation of atomic electron binding energies for monte carlo particle transport,” *Nuclear Science, IEEE Transactions on*, vol. 58, no. 6, pp. 3246–3268, 2011.
- [30] M. Batič *et al.*, “Algorithms and parameters for improved accuracy in physics data libraries,” *Journal of Physics Conference Series*, vol. 396, no. 2, p. 022039, Dec. 2012.
- [31] P. Venkataramaiah, K. Gopala, A. Basavaraju, S. S. Suryanarayana, and H. Sanjeeviah, “A simple relation for the Fermi function,” *J. Phys. G*, vol. 11, no. 3, p. 359, 1985.
- [32] M. E. Rose, *Beta-and gamma-ray spectroscopy*, K. Sieghahn, Ed. North Holland Publishing Company, Amsterdam, 1955.
- [33] D. H. Wright *et al.*, “Recent Developments and Validations in Geant4 Hadronic Physics,” in *Calorimetry in High Energy Physics: XII*, ser. American Institute of Physics Conference Series, S. R. Magill and R. Yoshida, Eds., vol. 867, Oct. 2006, pp. 479–486.
- [34] H. Bateman, “The solution of a system of differential equations occurring in the theory of radioactive transformations,” *Proc. Cambridge Philos. Soc*, vol. 15, p. 423, 1910.
- [35] M. Amaku, P. R. Pascholati, and V. R. Vanin, “Decay chain differential equations: Solution through matrix algebra,” *Comp. Phys. Comm.*, vol. 181, pp. 21–23, 2010.
- [36] R. J. Onega, “Radioactivity Calculations,” *Am. J. Phys.*, vol. 37, no. 10, pp. 1019–1022, 1969.
- [37] D. S. Pressyanov, “Short solution of the radioactive decay chain equations,” *Am. J. Phys.*, vol. 70, pp. 444–445, 2002.
- [38] L. Moral and A. F. Pacheco, “Algebraic approach to the radioactive decay equations,” *Am. J. Phys.*, vol. 71, pp. 684–686, 2003.
- [39] T. M. Semkow, “Comment on “Algebraic approach to the radioactive decay equations,” by L. Moral and A. F. Pacheco [Am. J. Phys. 71 (7), 684–686 (2003)],” *Am. J. Phys.*, vol. 72, pp. 410–411, 2004.
- [40] D. Yuan and W. Kernan, “Explicit solutions for exit-only radioactive decay chains,” *J. App. Phys.*, vol. 101, no. 9, p. 094907, 2007.
- [41] ISO JTC, “SC22/WG21. ISO/IEC 14882: 2011, Information Technology, Programming Languages, C++,” 2012.
- [42] M. Han, C.-H. Kim, L. Moneta, M. Grazia Pia, and H. Seo, “Physics data management tools: computational evolutions and benchmarks,” *ArXiv e-prints*, Dec. 2010.