

OPEN ACCESS

A Roadmap For Geant4

To cite this article: Makoto Asai 2012 *J. Phys.: Conf. Ser.* **396** 052007

View the [article online](#) for updates and enhancements.

You may also like

- [Development of a coupled simulation toolkit for computational radiation biology based on Geant4 and CompuCell3D](#)
Ruirui Liu, Kathryn A Higley, Maciej H Swat et al.
- [Backscattered electron emission after proton impact on gold nanoparticles with and without polymer shell coating](#)
F Hespeels, A C Heuskin, T Tabarrant et al.
- [Microdosimetry of the Auger electron emitting \$^{125}\text{I}\$ radionuclide using Geant4-DNA simulations](#)
H Fourie, R T Newman and J P Slabbert

ECS
The
Electrochemical
Society
Advancing solid state &
electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research

A Roadmap For Geant4

Makoto Asai, on behalf of the Geant4 Collaboration

SLAC National Accelerator Laboratory
2575 Sand Hill Road, Menlo Park, CA 94025, USA

E-mail: asai@slac.stanford.edu

Abstract. The Geant4 simulation toolkit is now in the 14th year of its production phase. Geant4 is the choice of most current and near future high energy physics experiments as their simulation engine, and it is also widely used in astrophysics, space engineering, medicine and industrial application domains. Geant4 is a “living” code under continuous development; improvement of physics quality and computational speed is still a priority for Geant4. It is evolving and being enriched with new functionalities. On the other hand, the simulation paradigm that prevailed during the foundation of Geant4 is now being rethought because of new technologies in both computer hardware and software. The Geant4 Collaboration has identified many options and possibilities. Geant4 has accommodated some of these by providing a multi-threading prototype based on event-level parallelism. In this article we discuss the past, present and future of the Geant4 toolkit.

1. Introduction

Geant4 [1, 2] is a toolkit for the Monte Carlo simulation of particles and radiation passing through and interacting with matter. It describes (1) the tracking of particles and radiation through a geometry composed of different materials, (2) their interactions with the electrons and nuclei encountered as well as with potential electromagnetic fields, (3) the creation of other particles in these interactions, and (4) the consequences in terms of detector responses or scored physics quantities such as dose. The toolkit is designed to model all the elements associated with detector simulation: the geometry of the system, the materials involved, the fundamental particles of interest, the physics processes governing particle interactions in the tracking of particles through materials and electromagnetic fields, the response of sensitive detector components, the storage of events and tracks, the visualization of the detector and particle trajectories, and the capture and analysis of simulation data at different levels of detail and refinement. It offers particle interaction codes ranging from fast, approximate parameterizations to precise and resource-intensive models incorporating the detailed current knowledge necessary for most publishable results.

The R&D project of developing Geant4 was initiated by two presentations at CHEP 1994 at San Francisco [3, 4], which resulted in the launch of CERN DRDC RD44 [5] in December 1994. Geant4 is the fully re-engineered, object-oriented, successor to GEANT version 3 [6]. It was a pioneering project that successfully adopted then-modern software engineering techniques to detector simulation in HEP and Nuclear Physics. RD44 made alpha-releases from April 1997, beta-releases from July 1998, and then the first public production release in December 1998. RD44 was successfully terminated and transformed into the international Geant4 Collaboration [7], which assumed the responsibilities of maintenance, further developments and user support of the Geant4 toolkit. Currently the Collaboration offers one full release and one beta release every year.

Geant4 and its derivative application codes have widespread uses in various domains of science. Geant4 is used by almost all running or planned experiments in experimental High Energy Physics (HEP) worldwide. The most prominent current HEP use is by the experiments at the Large Hadron Collider, which make continuous use of over one hundred thousand cores worldwide to run Geant4-based simulations. Geant4 has already been used over most of the lifecycle of these experiments, i.e. for detector design, calibration and alignment, and first analysis. Geant4 is a clear example of HEP software technology with transformational applications to other areas of scientific research. The Geant4 community of developers and users extends beyond HEP and nuclear physics into areas such as accelerator science, particle astrophysics, space engineering, medical physics, education, and industrial applications. In accelerator science, Geant4-based applications are used to design accelerators and their shielding structures. For example, the proposed Project-X muon collider is modelled by an application named G4beamline [8], and BDSim [9] for ILC/CLIC.

In astrophysics, Geant4's unique ability to simulate the acceleration of charged particles in dynamic electromagnetic fields enables the simulation of solar flares [10]. In addition, the magnetospheres of planets in our solar system are modelled in PlanetoCosmics [11] based on Geant4 to simulate the radiation environment of the planets. In space engineering, Geant4 has been used to evaluate the amount of radiation in a spacecraft, to calibrate on-board detectors when a craft is in operation, to estimate the radiation dose received by astronauts and electronic devices in the International Space Station and in future manned missions to Mars, and to simulate the radiation effects on semiconductor devices in space radiation environments [12, 13].

Throughout the world of medical physics, the use of Monte Carlo simulation is growing. In many innovative areas, such as particle therapy, mixed-mode imaging, and on-board imaging for dose verification, Geant4 has been the simulation tool of choice [14, 15, 16]. Medical users value Geant4 because of its ability to simulate the dynamic motion of the geometries of patients and the treatment apparatus. For developing educational software, Geant4 is an ideal tool not only because of its comprehensive physics coverage and flexible geometry description, but also because of its powerful visualization and interactivity. Several on-going projects are aimed at teaching the nature of fundamental particles and properties of radiation to college-level and even younger students. Industry has also adopted Geant4 as a tool to design non-destructive systems for inspection and testing. Widespread use of Geant4 is demonstrated by more than 5000 citations of one of its general papers [1] as of May 2012 [17].

2. Highlights of recent developments

2.1. Multi-threaded prototype

The current implementation of Geant4 relies on the '90s computing architecture paradigm of cheap and increasing memory resources serving limited computing power. It uses a single computational thread. In order to make more efficient use of modern multi-core shared memory hardware or even distributed memory clusters, particle simulation codes, such as Geant4, should be organized into multiple cooperating computational threads. Particle simulations easily map to models where each particle, or small groups of particles, is allocated to individual computational threads. This is the obvious method for parallelizing these types of applications and works well in situations where the entire simulation space can reside in a single shared memory.

To adapt Geant4 to many-core platforms, the Collaboration began in 2007 to prototype a multithreaded version of the code. Multi-threaded-Geant4 (below as Geant4MT) relies on an event-level parallelism approach similar to ParGeant4 [18] implemented for distributed-memory multi-processors on top of Task Oriented Parallel C/C++ (TOP-C) [19]. On a computer with k cores, Geant4MT can replace k independent copies of a Geant4 process with a single process with k threads, which have a reduced memory footprint and also attain scalability.

An application based on Geant4MT shares “relatively read-only data” among threads for memory footprint reduction, where relatively read-only data means data in memory space that is

written at the initialization phase of the program, but which is kept unchanged during the event loop. Such relatively read-only data reside in the shared central heap accessible from all threads. On the other hand, data, which are dynamically updated during the tracking phase, reside in individually dedicated thread-private heaps. For a benchmark of a simplified version of the full CMS detector simulation on Dunnington hexa-core with 24 local worker threads and 4000 electromagnetic events per each worker thread, each local thread required only 20 MB of additional memory while the master thread required 250 MB. This total size of 730 MB demonstrates a significant memory footprint reduction compared to the required memory size of 6 GB for 24 separate executions [20]. Figure 1 illustrates the use of memory space by a Geant4MT application.

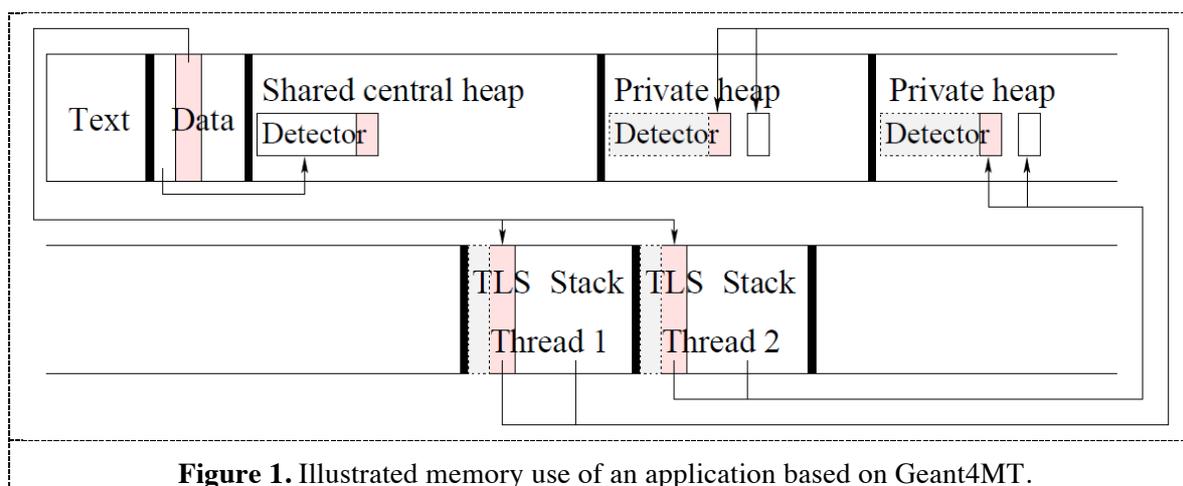


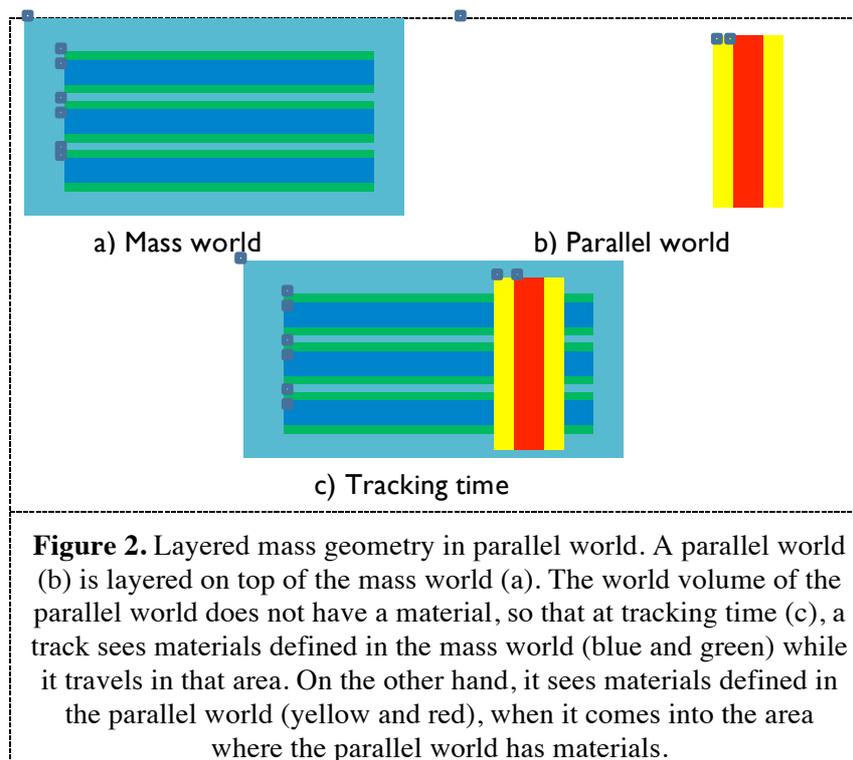
Figure 1. Illustrated memory use of an application based on Geant4MT.

The latest version of the Geant4MT prototype supports all manner of geometries and detector/scoring functionalities and all the physics models offered by Geant4. It has demonstrated excellent scalabilities for both Westmore and AMD chips. It still has limitations: it (1) currently runs only on Linux platforms, (2) does not yet support interactive GUI or visualization and runs only in batch mode, and (3) has been tested only with the most commonly-used physics lists. Migration to Geant4MT prototype requires straightforward and minimal changes in user code. Further details of this prototype and its performance are discussed in Ref. [20] and also elsewhere in these proceedings [21].

2.2. Layered mass geometry in parallel worlds

Since version 9.1 in December 2007, Geant4 has offered the possibility for a user to describe one or more parallel worlds in which artificial volumes are created to define sensitivity, dedicated production thresholds (so-called production cuts), shower parameterization, and so on [22]. A typical use-case of such a parallel world is defining a large scoring plane intersecting the detailed detector geometry to score for example neutron flux. Nevertheless, the user was not allowed to define any material in the parallel world; all volumes in parallel worlds were seen as ghost volumes just for defining conceptual volume boundaries.

In the latest version of Geant4 9.5, the “parallel world” functionality has been extended to allow the user to define materials in volumes defined in parallel worlds. Conceptually, a parallel world may be stacked on top of the ordinary world (so-called mass world or tracking world) or other parallel world. At tracking time, a track will see the material of the top-layered world, and if a material is not defined for the volume in that world (i.e. null pointer), material is taken from the world one layer beneath. This extension offers an alternative way of implementing a complicated geometry. For example, instead of employing a Boolean operation to make a hole for another volume, the volume representing the hole could be defined as a parallel geometry. Figure 2 illustrates such overlaid volumes.



A parallel world can be chosen to be associated only to some types of particles. In other words, the user may define geometries of differing levels of detail for different particle types. For the example of a sampling calorimeter, the mass world would define only the crude geometry with averaged material, while a parallel world would have all the detailed geometry with individual materials. Such a parallel world would then be associated to all particle types except electrons, positrons and gammas. Then electrons, positrons and gammas, being the most time-consuming particles for electromagnetic shower simulation, do not see volume boundaries and individual materials defined in a parallel world, and their steps are thus not limited by such boundaries. Shower parameterization may also have its dedicated geometry with only readout segmentation for the sake of rapidly locating randomly generated energy spots.

2.3. Improvements in electromagnetic physics

Electromagnetic physics is of primary importance for Geant4 physics, since the visible energy of hadrons is still due mainly to ionization and bremsstrahlung, and also because half of a hadronic shower is the electromagnetic component originating from π^0 decays into gammas. For the main observables, Geant4 electromagnetic physics describes the experimental data to within 1% agreement [23]. We are now concentrating on points of electromagnetic physics where the disagreement with data is above 1%. ATLAS, CALICE, and CMS report that Geant4 version 9.4 electron shower lateral profiles agree on the core but are slightly (1-2%) narrower in the tails [23]. These issues are mainly due to a fast and approximate description of our models, or because of medium and atomic physics effects. Details of Geant4 electromagnetic physics, including recent improvements and on-going validation such as the above-mentioned lateral profiles, are discussed in a dedicated presentation elsewhere in these proceedings [24].

2.4. Improvements in hadronic physics

In recent years, several improvements have been made to the various hadronic physics models Geant4 offers. These include the Fritiof (FTF) parton string model which has been extended to include antinucleon and anti-nucleus interactions with nuclei [25], the Bertini-style cascade with its improved

CPU performance and extension to include photon interactions [26], and the pre-compound and de-excitation models [27]. We have recently released new models and databases for low energy neutrons, and the radioactive decay process has been improved with the addition of forbidden beta decays and better gamma spectra following internal conversion [26].

As new and improved models become available, the physics lists we recommend for production use have evolved, starting from LHEP, QGSP, and QGSP_BERT to FTFP_BERT most recently. Now the spine of Geant4 hadronic physics consists of FTFP_BERT (Fritiof parton string model, Bertini cascade model and pre-compound model). FTFP_BERT looks to be on track for a “universal” response to hadronic physics needs. Details of Geant4 hadronic physics, including recent improvements and on-going validation, are discussed in a dedicated presentation elsewhere in these proceedings [26].

2.5. Improvements in usability

The Geant4 Collaboration puts continuous effort into improving the usability of the Geant4 toolkit. We are restructuring and polishing examples associated with the release code. Ease of physics list creation, combining existing physics builders, and adding processes to “pre-packaged” physics lists has been improved. Also, an automatic consistency check of the physics list is performed at run time. For example, if a user mistakenly assigns more than one process of the same kind to a particle, e.g. two multiple-scattering models to a particle, a warning will be issued.

Since version 9.5, all the warning and error messages Geant4 generates have the same banner and footer. This enables automated detection of warning/error messages embedded in the huge number of output files typical in the massive production runs of LHC experiments. Also, “*cout*”/“*cerr*” destinations become user-configurable depending on the severity of the error.

In version 9.5, the CMake system for installation has replaced the old Configure-based installation script. The use of CMake has been extended to cover all the installation procedures including the download of data sets. GNUMake build scripts are kept available for the sake of backward compatibility. Geant4 now comes with an embedded module for CLHEP, which includes the subset of the CLHEP library classes [28] required by Geant4. For users requiring the full CLHEP, there exists a choice at installation time to use the embedded CLHEP module or the CLHEP external library. With these changes for configuration, the installation of Geant4 libraries is now more straightforward.

3. Future and opportunities

Geant4 is “living” code under continuous development by about 100 scientists over the world. In this section we discuss highlights of the on-going developments and perspectives.

3.1. Improvements in performance

Improving computing performance is of importance to all Geant4 users and developers. Geant4 uses stochastic processes to model the interactions that take place when particles interact with matter. This is a time-consuming process and utterly vital to distinguishing between physics discoveries and unlikely fluctuations of known processes. For example, most particle physics experiments devote over half of their computing power to Geant4-based simulation. The HEP programs exploiting the CERN LHC in new computing hardware to run Geant4 invest more than \$10M per year. The total cost of this simulation (including power, cooling, housing, operation) is many tens of millions of dollars per year. Even with this investment, the much higher statistics of simulations needed by most experiments in the next ten years will require major improvements to execution efficiency in Geant4.

In the design of Geant4 kernel classes and their methods, we anticipate improvements in cache-hit-rate by redesign of data locality, reduction of virtual abstraction layers in class inheritance, and redesign to limit deep recursive calls, for example. Given its wide use and collaborative nature, Geant4 emphasizes the aspect of scientific productivity through intelligibility and maintainability, over that of raw computational efficiency. This is justified by the large potential costs of difficult-to-maintain

code. Still, the sentiment is that it is possible to articulate these two aspects and to keep a high standard of productivity and maintainability while adaptively exploiting the performance potential of architectural innovations.

At the code implementation level, in particular in physics, physics quality was generally emphasized over computing performance. We anticipate a code review and potential re-implementation. It is likely that examining and refactoring the sequences of method invocations by applying transformations such as specialization (partial evaluation of arguments) as well as data transformations for locality can bring a large gain. Given the nature of the Geant4 code, such work must be very sensitive to the overall software engineering concepts built into the code, so as not to adversely affect its overall design and hence maintainability. Also, we must not lose physics performance; massive verifications will be part of the process.

These changes are essentially independent of the adaptation to multi-threading. If we could gain performance by any of these changes for a single-thread application, they will similarly benefit the multi-threaded version. In addition, these changes must be transparent to user code at least for average users, so that they would not cause a massive migration effort.

3.2. Event/track level full reproducibility

Recently the ATLAS experiment identified a bug in one of the Geant4 physics models, which caused unphysical simulation results 20 times in 1 billion events (each event has millions of interactions). To pinpoint the cause of such a rare problem, event/track-level full reproducibility is required. That is, as long as a program is restarted with the same random number engine status, it should regenerate exactly the same result regardless of other conditions. In particular, as we shift to multi-threading application, we cannot re-create the problem without track-level reproducibility. Many of the causes of irreproducibility have already been identified and resolved. Recent developments in Geant4MT brought us a new tool for identifying points of divergence [20].

3.3. Multi-threaded Geant4

We anticipate a few more releases of Geant4MT prototypes corresponding to every Geant4 patch release in 2012. In these prototypes, one of the major revisions will involve improvement of usability in particular for external “frameworks”. Migration of a standalone application to Geant4MT is already straightforward and simple, but we still need to identify the interfaces relevant to external frameworks, which are used by large experiments in particular at the LHC. We also need to catch up on all design and implementation changes described in section 3.1 for Geant4MT. We plan that Geant4MT based on the next Geant4 public release (version 9.6 at the end of 2012) will be the final prototype release. In 2013 we will merge Geant4MT into the main development code base. We anticipate version 9.6 at the end of 2012 will be the final minor release of Geant4 version 9 series. The 2013 release will then be a major release with full multi-thread capability. Since the 2013 release is a major release, it may cause some minimal migration cost for user code. We plan to make the first beta release of this major upgrade in June 2013.

3.4. Extending Geant4 functionalities

Several development activities to extend Geant4 functionalities are underway. Following are some of the developments we expect to deliver in coming releases.

3.4.1. Phonon transport in crystal and electron/hole drift in a semiconductor

Recent dark matter detection experiments such as CDMS (Cryogenic Dark Matter Search) [29] use crystals at sub-Kelvin temperatures. In such a low temperature crystal, individual phonons travel almost independently of one another, and thus can be treated as particles. This development introduces the concept of crystal structure to Geant4. In addition, this introduction of crystal structure enables Geant4 to simulate the drifting motion of electrons and holes in a semiconductor crystal [30].

3.4.2. Material activation

Simulation of activation of irradiated material has been a long-demanded feature in Geant4. It has many use-cases including high luminosity HEP experiments, satellites passing through space radiation environments, and shielding devices for accelerators. A new convolution mechanism is being developed so that the radiation spectrum of activated material at a certain time from a given irradiation period could be predicted. Early results show good agreement with dedicated beam tests [31].

3.4.3. Very low energy electromagnetic physics and physico-chemical processes

New physics processes are being introduced for electrons, protons and some ions for the modelling of early biological damage induced by ionising radiation at the DNA scale down to the eV regime [32, 33]. Applicability of these physics models is primarily limited to liquid water, with the ambition of extending it to some other biological and semiconductor materials. Physico-chemical and chemical processes that allow the modelling of molecular species production, diffusion and mutual interactions in liquid water are currently under development [34].

3.5. Future opportunities in computing

Extrapolating five years, we anticipate more cores per chip. Intel's MIC (Many Integrated Core) is about to be delivered. In addition, many of these new cores will likely be a heterogeneous mix of processors, with a few optimized to maximize single thread throughput, while most are designed to maximize energy efficiency with wide SIMD (Single Instruction, Multiple Data) data paths. The AMD Fusion family of Accelerated Processing Units (APU), blending Opteron CPUs and Radeon GPUs, is just the beginning. This trend will not only exacerbate today's performance optimization challenges, but also simultaneously promote the issues of energy consumption and resilience to the forefront. On the programming side, new languages such as CUDA and OpenCL have appeared to address this new hardware era. Also, analysis of the overall structure of Geant4 and its application code in terms of computational flows, algorithms and models may enable the development of code generation schemes and execution strategies targeting future high-performance architectures. Domain-Specific Languages (DSLs) may specifically address the problem of simultaneously achieving programmer productivity and machine performance.

The Geant4 Collaboration is quite aware of these opportunities. We will keep our eyes on new trends, keep in touch with and/or participate in projects that explore them, and incorporate useful elements into our design iterations. The Geant4 Collaboration continues to improve the computing performance of the toolkit, while preserving its broad range of functionality, physics performance and flexibility. We expect such improvements to be evolutionary rather than revolutionary so that they will not cause the rewrite-from-scratch of user code. We will also keep these changes free from any specific hardware or programming paradigm so that Geant4 will persist over the long timescale of modern HEP experiments.

4. Summary

Geant4 is now in the 14th year of its production phase. Despite its age, it is still evolving and being enriched with new functionalities. This demonstrates the advantage of the use of OO technologies, and shows the appropriateness of the early design adopted during the RD44 era.

Improvement of physics quality and speed remains a priority for Geant4. Tackling the percent level inaccuracies in EM physics is the current challenge. Large investments in hadronic physics have lead to a "Darwinian" emergence of the FTFP_BERT physics configuration, which looks to be on track for a "universal" response to hadronic physics needs.

New technologies trigger in-depth rethinking of simulation paradigms. The Geant4 Collaboration is quite aware of these opportunities. We have accommodated some of these by providing a multi-threading prototype based on event-parallelism. This will become the Geant4 baseline by 2013. We continue to keep our eyes on other opportunities.

5. References

- [1] S. Agostinelli *et al.*, “Geant4 – A Simulation Toolkit”, Nuclear Instruments and Methods A 506 (2003) 250-303
- [2] J. Allison *et al.*, “Geant4 Developments and Applications”, IEEE Transactions on Nuclear Science 53 No. 1 (2006) 270-278
- [3] R. Brun *et al.*, “GEANT steps into the future”, CHEP 1994
- [4] K. Amako *et al.*, “Object oriented analysis and design of GEANT based detector simulator”, CHEP 1994
- [5] RD44 Collaboration, CERN/LHCC 98-44 (1998)
- [6] <http://wwwasdoc.web.cern.ch/wwwasdoc/geant/geantall.html>
- [7] <http://www.geant4.org/>
- [8] <http://www.muonsinc.com/muons3/G4beamline>
- [9] <http://www.pp.rhul.ac.uk/twiki/bin/view/JAI/BdSim>
- [10] J. Kotoku *et al.*, “Effects of Compton scattering on the Gamma Ray Spectra of Solar flares”, arXiv:0708.0057 (2007)
- [11] <http://cosray.unibe.ch/~laurent/planetocosmics/>
- [12] <http://www.spennis.oma.be/>, also
D. Heynderickx *et al.*, “Space Environment Information System (SPENVIS): a WWW interface to models of the space environment and its effects”, AIAA-2000-0371 (2000)
- [13] <https://creme.isde.vanderbilt.edu/>, also
R.A. Weller *et al.*, “Monte carlo simulation of single event effects,” IEEE Trans. Nucl. Sci., vol. 57, no. 4, pp. 1726-1746 (2010)
- [14] <http://www.opengatecollaboration.org/>
- [15] T. Aso *et al.*, “Geant4 Based Simulation Framework for Particle Therapy System”, IEEE/NSS Conf. Rec. Vol.4 2564-2567 (2007)
- [16] J. Perl *et al.*, “TOPAS - An innovative proton Monte Carlo platform for research and clinical applications”, Medical Physics (in press, 2012)
- [17] <http://scholar.google.com/scholar?hl=en&q=Geant4>
- [18] <http://www.ccs.neu.edu/home/gene/pargeant4.html>
- [19] G. Cooperman, “TOP-C: Task-Oriented Parallel C for distributed and shared memory”, In Workshop on Wide Area Networks and High Performance Computing, volume 249 of Lecture Notes in Control and Information Sciences, 109-118 (1999)
- [20] X. Dong *et al.*, “Multithreaded Geant4: Semi-automatic Transformation into Scalable Thread-Parallel Software”, Proc. of Euro-Par 2010 - Parallel Processing, Lecture Notes in Computer Science 6272, Springer, pp. 287-303 (2010)
- [21] X. Dong *et al.*, “Improving Geant4 Multi-core’s Performance and Usability”, Proceedings of CHEP 2012 to be published
- [22] J. Apostolakis *et al.*, “Parallel Geometries in Geant4: Foundation and Recent Enhancements”, Proceedings of the 2008 IEEE NSS/MIC/RTSD Conference
- [23] <http://indico.cern.ch/conferenceOtherViews.py?view=standard&confId=144956>
- [24] J. Allison *et al.*, “Geant4 electromagnetic physics for high statistic simulation of LHC experiments”, Proceedings of CHEP 2012 to be published
- [25] V. Uzhinsky *et al.*, “Development of the Fritiof Model in Geant4”, Proceedings of the SNA+MC2010 Monte Carlo Conference, Tokyo, 2010
- [26] J. Yarba, on behalf of the Geant4 Hadronics Group, “Recent Developments and Validation of Geant4 Hadronic Physics”, Proceedings of CHEP 2012 to be published
- [27] J.M. Quesada *et al.*, “Recent Developments in Pre-equilibrium and De-excitation Models in Geant4”, Proceedings of the SNA+MC2010 Monte Carlo Conference, Tokyo, 2010
- [28] <http://proj-clhep.web.cern.ch/proj-clhep/>
- [29] <http://cdms.berkeley.edu/>

- [30] D. Brandt *et al.*, “Cryogenic Charge Transport on Crystals Using Geant4”, presentation at 8th Geant4 Space Users’ Workshop (2011)
- [31] T. Sato *et al.*, “In-Orbit Activation Study of ASTRO-H X-ray Observatory Using Geant4”, submitted to IEEE/NSS 2012
- [32] <http://geant4-dna.org/>
- [33] S. Incerti *et al.*, “Comparison of Geant4 very low energy cross section models with experimental data in water”, *Med. Phys.* 37 (2010) 4692-4708
- [34] M. Karamitros *et al.*, “Modeling radiation chemistry in the Geant4 toolkit”, *Prog. Nucl. Sci. Tec.* 2 (2011) 503-508