Monte Carlo Transport Codes for use in the Space Radiation Environment

Lawrence Pinsky

Physics Department, University of Houston

Posted: 02/13/2015

Introduction

The recent article in Reviews of Modern Physics by Durante and Cucinotta[1] gives a review of the general situation with respect to both deterministic and Monte Carlo codes. This article is intended to supplement that with information about the strengths and weaknesses of the Monte Carlo codes in particular, as well as documenting how potential users can access them. This article in not intended as an exhaustive review of the detailed properties of those codes, but as a rough guide for individuals wanting to survey the field. The codes that are considered include GEANT4, FLUKA, PHITS, and MCMP-X, (all of which are available for free for academic non-commercial uses). It should be mentioned that this is not an exhaustive list of particle transport codes, a number of which are focused on particular applications. The focus of the codes chosen here is their applicability for use in simulating space radiation shielding situations and the salient feature that allows the application to this problem is their ability to simulate a broad range of the interactions of "heavy" ions (Z > 1) as they traverse matter.

All of the Monte Carlo codes discussed here function in a very similar manner. Generally they require an input, which specifies the initial source of the particles to be transported in terms of their starting position within a 3-dimensional geometry, as well as their initial direction and energy (or momentum), and of course particle type. Each of the codes offers a format that can be used to specify the details of the geometry to be considered, which includes not only the physical shapes and locations of the structure of the entity through which the radiation is to be transported (e.g. a spacecraft or a habitat on the lunar surface, etc.), but also crucially the specific detailed composition and density of the material in each region. The codes all "propagate" each initial particle through the geometry, calculating the local effects both on the particle itself (e.g. energy loss by ionization of the medium, multiple scattering, etc.) and on the medium (e.g the energy absorbed by the medium, in some cases the activation of the medium, etc.). The physics processes simulated in detail attempt to include all of those that the authors deem to be relevant (e.g. ionization, Cerenkov radiation, inelastic nuclear collisions, etc.). In many cases the individual interactions are simulated in great detail, and generally, those processes that are not expressly simulated in detail are usually accounted for in approximate or integrated ways (e.g. individual atomic ionizations, multiple scattering, etc.).

Once the initial particle type, starting point and momentum are known, along with the nature of the material through which it is propagating, the stochastic interactions can be simulated with physics-based (either theoretical or empirical) models for these processes and interactions through the use of random number generators. The use of random number generators is a science unto itself, and will not be dealt with here, but suffice it to say that the authors of all these codes have generally made careful choices in how to employ them, typically isolating that issue from the typical user. While these codes are thought of as being essentially stochastic in terms of their output, in actuality they are deterministic in the sense that if one starts with the same random number seed under the same computational conditions (computer, operating system, compiler, etc) the exact same result will be produced. In fact, often the codes are distributed with reference input files and associated expected outputs to be used for verification that the installation has been properly achieved. Provisions are also usually made to save the ending random number seed for use in starting the next run so that the results of multiple runs can be added to increase the statistical accuracy properly.

Clearly, the major hurdle faced by potential users is to represent complex geometries accurately. As noted by Durante and Cucinotta[1] in many past applications, simple global approximations to such entities as the International Space Station have been used in lieu of detailed geometric modeling. In part this is due to the mismatch between the methods used to define geometries in the Monte Carlo codes and the standard Engineering CAD (Computer Aided Design) formats that may be available for very complex structures. The Monte Carlo codes all function in a similar fashion with respect to the geometry when transporting particles.

At the most basic level, as a particle is transported through the geometry, the code needs to know two specific facts: (1) what the material and its density is in the present region; and (2) how far the particle can move in its current direction before it leaves that region and enters another region with a different material. By far the most common calculation (and therefore the most time consuming aspect of the codes) being done during execution is the determination of that distance. For all of the codes, the boundaries of the volumes are generally limited to nothing more detailed than quadratic surfaces. This is because the distance to a quadratic (or linear, i.e. planar) surface is analytically solvable in a simple enough form to minimize calculation time. Unfortunately, virtually all of the CAD formats use more complex surfaces such as 5th degree polynomials, and tend to provide only surfaces rather than primitive volumes. As such, at the present time there is no universal convertor for CAD formats into any of the individual Monte Carlo formats. Likewise, a number of 3-dimensional graphics formats have been considered for use (e.g. Open GL) as universal Monte Carlo transport geometry, but none has been adopted yet. Some of the codes have auxiliary tools to aid in geometry construction, but it still remains a major challenge to provide a geometry file with a high fidelity replication of the features of something as complex as the ISS or the Space Shuttle. However, that is not to say that it is impossible to do, as examples like the massive detectors deployed at the Large Hadron Collider LHC have all been modeled in Monte Carlo geometries, some with > 10 million individual primitive volume elements.

Another related geometry issue is the provision for transport through the tissues of the human body. There are a number of approaches that are used by the Monte Carlo codes including numerical phantoms made out of quadratic surfaces for such structures as the internal organs, and voxel phantoms with as small as mm-sized voxels to represent human bodies. The use of CT-scans to assemble relatively accurate current voxel models for particular individuals has been employed with Monte Carlo codes in the radiation therapy field for some time. This capability of the Monte Carlo codes means that simulating the detailed transport of space radiation through and within putative astronauts' bodies within a spacecraft or a space suit is more or less available. It is the spacecraft structure external to the astronaut that typically presents the difficulty.

The most obvious challenge beyond the representation of the geometry of the spacecraft with sufficient detail and accuracy is the computation time. Monte Carlo codes are all limited by statistics, which in turn are related to calculation times. However these Monte Carlo codes are known as being "embarrassingly parallelizable" because of the ability in most cases to superimpose the results of multiple runs with the same conditions, but different initial random number seeds. So, with the advent of multiprocessor CPUs and massive supercomputers, the prospects of being able to obtain the results from a very large number of incident particles impinging on a spacecraft seems impressive. However, if one considers that the free space interplanetary solar minimum integral fluence of particles at all significant energies in the Galactic Cosmic Rays (GCR) is over 3,000 per m^2/ster s, if one postulates an interplanetary spacecraft of modest size (say a sphere of radius 5 m), then its surface is penetrated by almost 2 million incident GCR particles every second. So, even a modest attempt to simulate the full-unbiased transport of all incident GCR particles over a single 10-minute period would require the transport of over a billion particles (10^9).

Having said that, there are still very tractable uses for Monte Carlo simulations of space radiation situations using biasing. Biasing can be done in many ways both within the code and by adding the effects of a number of separate runs with carefully chosen incident particle selections. Internal biasing in a Monte Carlo code means that the code artificially increases the probability of rare events occurring and then automatically compensates for that bias in the accounting of the expected physical outcome (e.g. Increasing the probability of neutrino interactions to make them common, and then adjusting the reported rate back to the correct relative values in the final tallies, but with far greater significance in the statistical error sense). Perhaps a more relevant example is the biasing of abundance of the relatively rare heavy ions in the GCR flux due to their substantially greater radiobiological importance. Using a code that provides for biasing internally is preferable because one can be more confident in the final compensated numbers than when it is done artificially, external to the code by the user who has to self-correct for the ratios.

Finally, as also noted by Durante and Cucinotta, perhaps the most important use of Monte Carlo codes in assessing the radiation environment in space applications will be in the simulation and evaluation of neutron fluences. Neutrons are particularly hard to provide shielding from as they tend to bounce around much like the steel ball in an arcade pinball machine, making many elastic scatters causing major changes in direction while losing relatively little energy when the scattering nucleus is heavy. Some nuclei do have large neutron absorption cross sections, but those are typically energy dependent, and scattering off of hydrogen (i.e. protons) typically results in thermalization of the neutrons, wherein the capture cross sections can become large, resulting in the production of unstable isotopes. This is particularly problematic when it occurs within the astronaut's body itself.

There are very few if any primary neutrons, but there can be externally produced neutrons such as the albedo neutrons seen in Earth orbit that result from the collision of higher energy GCR nuclei with the Earth's atmosphere followed by fortuitous backscatters. A similar neutron albedo situation exists on the lunar surface. However, most of the neutrons of interest inside spacecraft will be produced in the interactions of the charged particles that are incident on the spacecraft, particularly after relatively large amounts of shielding have been traversed. Thus, the Monte Carlo codes will need to simulate the nuclear interactions accurately in 3-dimensions including the production of neutrons in the final state as well as accurately representing the propagation of the neutrons after they are produced.

Each of these codes also provide their own "scoring" capabilities, in terms of what things the user can require to be reported and to some extent the format in which it is provided. This is typically done, either in the form of output data files, which must be processed analogous to real data in order to produce the typical plots and figures used to display the properties of the data. In some cases, the outputs can be directly piped into plotting programs or those connections may be built into the codes themselves from the user's perspective.

Besides the format of the geometry that it is necessary to input to each of the codes addressed in this summary, the major differences are in the choices that the authors have made regarding the modeling of the fundamental physics processes. In some cases particular aspects are just not included and in others there may be simple approximations. Where more detailed modeling is employed the approach can be phenomenological (e.g. polynomial fits to the existing data), or it can be "physics-based" wherein fundamental physics models that have been adjusted to fit the data reasonably well are employed. The advantage of the latter approach where there are reasonable theories available is that they stand a better chance of being extensible to regions where data are lacking. Some codes include multiple models covering the same phenomena, allowing the users to choose their favorite one. In some cases the fundamental parameters employed by individual models are accessible to the users and references are provided to guide them in choosing the values most appropriate for their specific applications.

GEANT4 ("Giant" in French—Available at: http://geant4.cern.ch/)

Strictly speaking, GEANT4 as distributed is not a ready-to-run integrated code, but rather is self-styled as a "toolkit" for assembling a custom Monte Carlo code that provides numerous options that can be included in the final code to tailor it to a specific application. GEANT4 is a CERN-developed code (CERN is the European Laboratory for Nuclear Physics in Geneva, Switzerland). GEANT4 was initially an evolution from the earlier version, known as GEANT 3.21, a Fortran-based code. The earlier version is still available and actually is still being used surprisingly often. That earlier version is closer to a stand-alone code rather than a toolkit, but it also has some options that can be selected in terms of physics models. GEANT4 began as a modernization project of GEANT 3.21 and initially focused on converting the GEANT 3.21 code to C++, an object-oriented language. From that early beginning GEANT4 has evolved into a large collection of libraries that include updated models comparable to what exists in the other major codes in most cases and as is the case with all of the codes, it has a few unique features along with some limitations.

With respect to space radiation applications, the European Space Agency (ESA) has invested considerable funding over the years in the GEANT4 project to enable it to provide support for specific space radiation applications. One unique capability in GEANT4 is the extension of the emission of electrons and photons by excited atoms down to an energy of 100 eV, where most codes terminate individual emission tracking at around 1 KeV, simply summing everything below that into a common channel. The motivation for this effort was the potential for applications in remote sensing of the properties of asteroids, for example.

Perhaps the greatest limitation of GEANT4's capabilities with respect to simulating the space radiation environment is the lack of coherent heavy ion event generators. GEANT4 does include separate capabilities to simulate total cross sections and independently various aspects of heavy ion interactions such as ionization energy losses and fragmentation, but what is lacking at the time of this writing is one coherent model that actually and accurately simulates all of the correlations within each individual heavy ion interaction, such as has been included in all of the other codes discussed here.

FLUKA (FLUctuating KAskade—Available at: http://www.fluka.org/fluka.php)

FLUKA is an INFN (Italy) & CERN supported code developed by the FLUKA Collaboration. The current version of FLUKA is a multi-decade evolution of a code that was originally developed by Johannes Ranft. FLUKA, like the rest of the Monte Carlo codes described here, presents the user with a relatively "turn-key" code wherein the code itself is delivered basically as a ready to run executable. Like FLUKA, All of them require input files that specify the parameters that control the options available,(e.g. lower limits to energy propagation of particles) as well as the physical conditions, such as the geometry and the initial conditions for the particles to be transported. FLUKA is one of the codes that broadly employs physics-based models for the interactions, and is continually updated and refined as new data become available. FLUKA also has the capability to model extended induced radioactivity due to incident radiation on materials in the geometry employed. This is useful in predicting long-term latent dose effects from activation of nuclei in the body of the astronauts themselves, as well as from the decay of such products in the surrounding spacecraft environment.

Of relevance to space radiation simulation, FLUKA has a collection of different heavy-ion event generators that are each employed over different ranges of the energy spectrum. At the highest energy, DPMJET3 is used, followed by a heavily modified version of RQMD at energies down to ~100 MeV/A, at which point a model based on the Boltzmann Master Equation approach is used. One of the advantages of using complete coherent event generator models is that the interactions are usually simulated in the Center of Mass frame, so that all resulting fragments as well as any newly created particles are predicted not only by particle type (or specific fragment isotope) but also in terms of the specific direction of each one of them. This allows the accurate evaluation not only of neutrons as described earlier, but also to take into account any very low energy "target-fragments" (in the "lab" frame) that may result, which is of significant interest when the interactions occur within the astronaut's body.

FLUKA provides the ability to employ biasing in a number of ways to enable the user to reduce the computing time needed to simulate rare events with high relative statistical significance. FLUKA also provides capabilities to simulate the interactions in common detector systems that allow the user to compare its predictions directly with experimental results by simulating to actual raw data files produced by the experimental apparatus. This allows the comparison to take place at a point where the vagaries of possible systematic differences that might occur during the analysis of the raw data do not exist . Also, the availability of "history-files" from the Monte Carlo outputs can be useful to the experimenter to understand the source of at least some of the "outlier" events that are typically "cut" during the final analysis of the raw data.

FLUKA is written in Fortran, and as such is sensitive to the choice of Fortran complier used. Where users employ a compiled version that is compatible with their operating system and CPU, this is typically not an issue. However, as the operating systems are updated and new processors become available, the need to have a Fortran compiler that is capable of implementing the details of the Fortran included in FLUKA is sometimes a challenge for the authors. This can be an issue for the user when the user takes advantage of the ability within FLUKA to write and compile into the code a separate custom user scoring subroutine. Such a capability is useful when implementing the process of producing output tailored specifically to the users' needs.

MCMP(X) (Monte Carlo Multi-Purpose)

The current versions of MCMP are an evolution of one of the earliest integrated transport codes, and a number of its embedded features have been incorporated into a number of the other codes described here. One potential difficulty is that because it is a US-developed code (by Los Alamos National Laboratory) its distribution is restricted due to ITAR regulations. Currently the code is distributed by Oak Ridge National Laboratory in optical disc format rather than being available for download as are the other codes described in this article.

PHITS (Particle and Heavy Ion Transport code System)

PHITS (Particle and Heavy-Ion Transport code System) [1] is a general purpose particle and heavy-ion MC transport code which can transport neutrons from thermal energies up to 200 GeV, and the same method as in the MCNP4C code [2] is employed for neutrons with energies between 1 meV and 20 MeV based on the Evaluated Nuclear Data such as the ENDF-B/VI [3], JENDL-3.3 [4,5]; and for p and n up to 3 GeV for the JENDL-HE [6,7] file. Above 20 MeV, the Bertini model with free p-p and n-n cross sections parameterized according to Niita et al. [8] is used up to 3 GeV, while the simulation model JAM (Jet AA Microscopic Transport Model) developed by Nara et al. [6] is used above 3 GeV for nucleons, above 2.5 GeV for pions, and for all energies for all other baryons. JAM is a hadronic cascade model, which explicitly treats all established hadronic states including resonances with explicit spin and isospin as well as their anti-particles. For protons and other hadrons, JAM is used above 1 MeV, but for charged particles below 1 MeV only the ionization process is considered until the particles are stopped. PHITS also uses Evaluated Nuclear Data for photon and electron transport below 1 GeV in the same manner as in the MCNP4C code based on the ITS code, version 3.0 [9]. The energy range of electrons and photons is restricted to the energy region 1 keV - 1 GeV at the present, but the extension of the maximum energy of these particles is in progress. PHITS can also transport nuclei in any solid, gas or liquid material. Below 10 MeV/n, only the ionization process for the nucleus transport is taken into account, but above 10 MeV/n the nucleus-nucleus collisions up to 100 GeV/n is described by the simulation model JQMD (JAERI Quantum Molecular Dynamics) developed by Niita et al. [10]. In the QMD model, the nucleus is described as a self-binding system of nucleons, which are interacting with each other through the effective interactions in the framework of molecular dynamics. One can estimate the yields of emitted light particles, fragments and excited residual nuclei resulting from the heavy ion collision. The QMD simulation, as well as the JAM simulation, describes the dynamical stage of the reactions. At the end of the dynamical stage, excited nuclei are created and must be forced to decay in a statistical way to get the final observed state. In PHITS the GEM model [11] (Generalized Evaporation Model) is default employed for light particle evaporation and fission process of the excited residual nucleus.

When simulating the transport of charged particles and heavy ions, the knowledge of the magnetic field is sometimes necessary to estimate beam loss, heat deposition in the magnet, and beam spread. PHITS can provide arbitrary magnetic fields in any region of the setup geometry. PHITS can simulate not only the trajectory of the charged particles in the field, but also the collisions and the ionization process at the same time. For the ionization process of the charged particles and nuclei, the SPAR code [12] is the default used for the average stopping power dE/dx, the first order of Molière model for the angle straggling, and the Gaussian, Landau and Vavilov theories for the energy straggling around the average energy loss according to the charge density and velocity. In addition to the SPAR code, the ATIMA package, developed at GSI [13, 14], has been implemented as an alternative code for the ionization process.

The total reaction cross section, or the lifetime of the particle for decay, are essential quantities in the determination of the mean free path of the transported particle. According to the mean free path, PHITS chooses the next collision point using the MC method. To generate the secondary particles of the collision, we need the information of the final states of the collision. It is therefore very important that reliable data of total non-elastic and elastic cross sections is used for the particle and heavy ion transport. In PHITS, the Evaluated Nuclear Data is used for neutron-induced reactions below 20 MeV. For neutron-induced reactions above 20 MeV a parameterization is used. As for the elastic cross sections, the Evaluated Nuclear Data is also used for neutron-induced reactions below 20 MeV, and a parameterization is used above 20 MeV [8]. Parameterizations are also used for proton induced reactions for all energies, and for the double differential cross sections of elastic nucleon-nucleus reactions [8]. We have also adopted the NASA systematics developed by Tripathi et al., [15-17] for the total nucleus-nucleus reaction cross section, as an alternative to the Shen formula [18]. PHITS has been extensively used and benchmarked for many different applications, including various space applications, e.g. [19-29].

When estimating the biological damage of high energetic photons and charged particles, the contribution from the neutrons created both outside and inside the human must also be considered. It is therefore important to be able to calculate the kinetic energy distributions of the created secondary charged particles from photonuclear and neutron induced reactions. For low energetic neutrons, normally nuclear data is used. However, based on the one-body Bolzman equation, energy and momentum is not conserved in an event during the transport calculations. They are only conserved as an average over many randomly calculated events since the Bolzman equation only include mean values of the one-body observables in the phase space and cannot give two-body and higher correlations. A feature has therefore been included in PHITS to treat low energy neutron collisions as "events" which means that the energy and momentum is conserved and makes it possible to extract kinetic energy distribution of the residual nuclei, two particle correlations, etc. In PHITS, the transport algorithm has been changed for the low-energy neutrons from that on solving Bolzmann equation to an algorithm based on an event generator. By using this event generator mode, energy and LET distributions for all charged particles, created by all charged particles and neutrons, can be calculated.

When estimating the direct biological effects of radiation, microdosimetric quantities, such as the lineal and specific energy, are better indexes for expressing the RBE of the primary and secondary particles in comparisons to the conventionally often used LET. Although the use of microdosimetric quantities in macrosscopic transport codes is limited, because of the difficulty in calculating the provability distributions on macroscopic matter. Therefore mathematical functions, for calculating the microdosmetric probability densities in macroscopic material, have been incorporated in PHITS. This makes it possible to instantaneously calculate the probability densities of lineal and specific energies around the trajectories of high energetic primary and secondary charged particle tracks. A method for estimating the biological dose, the product of physical dose and RBE, for charged particles has also been established by using the improved PHITS coupled to a microdosimetric kinetic model [30, 31]. Since the energy distributions of the secondary charged particles from neutron induced reactions can be estimated by using the "event generator" in PHITS, an estimation of the RBE of the neutrons can also be made. The accuracy of this will be evaluated in the near future.

References

[1] Niita, K. et al. JAEA-Data/Code 2010-022, 2010; http://phits.jaea.go.jp

[2] J. F. Briesmeister, et al., MCNP General Monte Carlo N-Particle Transport Code, Los Alamos National Laboratory report; LA-12625-M (1997).

[3] V. McLane, et al., ENDF/B-VI Summary Documentation, BNL-NCS-17541 (1996).

[4] K. Shibata, et al., Japanese Evaluated Nuclear Data Library Version 3 Revision-3: JENDLE-3.3, J. Nucl. Sci. Technol. 39, 1125 (2002).

[5] Watanabe, Y. et al. Nuclear data evaluations for JENDL high-energy file. Proceedings of International Conference on Nuclear Data for Science and Technology, Santa Fe, USA, Sep.26-Oct.1, 2004; AIP CP769, p326-331, 2005.

[6] Nara, Y. et al. Relativistic nuclear collisions at 10A GeV energies from p+Be to Au+Au with the hadronic cascade model. Phys. Rev. C61:024901, 1999.

[7] T. Fukahori, et al., JENDL High Energy File. J. Nucl. Sci. Technol., Suppl. 2, 25-30 (2002).

[8] K. Niita, H. Takada, S. Meigo, Y. Ikeda, High-energy particle transport code NMTC/JAM, Nucl. Instrum. Methods, B184, 406 (2001).

[9] J. A. Halbleib, et al., ITS Version 3.0: The Integrated TIGER Series of Coupled Electron/Photon Monte Carlo Transport Codes, SAND91-1634 (1992).

[10] K. Niita, et al., Analysis of the (N,xN') reactions by quantum molecular dynamics plus statistical decay model, Phys. Rev. C52, 2620 (1995).

[11] S. Furihata, Statistical analysis of light fragment production from medium energy proton-induced reactions, Nucl. Instr. and Meth. B171, 251 (2000).

[12] T. W. Armstrong, and K. C. Chandler, A Fortran program for computing stopping powers and ranges for muons, charged pions, protons, and heavy ions, ORNL-4869, Oak Ridge National Laboratory, (1973).

[13] H. Geissel, C. Scheidenberger, Slowing down of relativistic heavy ions and new applications, Nucl. Instr. Meth. B136-138, 114-124 (1998).

[14] C. Scheidenberger, H. Geissel, Penetration of relativistic heavy ions through matter, Nucl. Instr. Meth. B136-138, 114-124 (1998).

[15] R. K. Tripathi, F. A. Cucinotta, J. W. Wilson, Accurate universal parameterization of absorption cross sections, Nucl. Instr. and Meth. B117, 347 (1996).

[16] R. K. Tripathi, J. W. Wilson F. A. Cucinotta, Accurate universal parameterization of absorption cross sections II – neutron absorption cross sections, Nucl. Instr. and Meth. B129, 11 (1997).

[17] R. K. Tripathi, F. A. Cucinotta, J. W. Wilson, Accurate universal parameterization of absorption cross sections III – light systems, Nucl. Instr. and Meth. B155, 349 (1999).

[18] W. Shen, et al., Total reaction cross section for heavy-ion collisions and its relation to the neutron excess degree of freedom, Nucl. Phys. A 491 (1) 130-146 (1989).

[19] T. Sato et al., Applicability of particle and heavy ion transport code PHITS to the shielding design of spacecrafts, Rad. Meas. 41, 1142-1146 (2006).

[20] L. Sihver, D. Mancusi, K. Niita, T. Sato, L. Townsend, C. Farmer, L. Pinsky and I. Gomes, Benchmarking of calculated projectile fragmentation cross sections using the 3-D, MC codes PHITS, FLUKA, and MCNPX", Acta Astronautica 63, 865-877 (2008).

[21] T. Sato, L. Sihver, K. Gustafsson, D. Mancusi and K. Niita, Shielding Design of Spacecrafts Using PHITS, American Nuclear Society Transactions, 99, 592 (2008).

[22] L. Sihver, T. Sato, K. Gustafsson, V.A. Shurshakov, and G. Reitz, Simulations of the

MTR-R and MTR Experiments at ISS, and Shielding Properties Using PHITS,

IEEEAC paper 1015 (2009).

[23] L. Sihver, et al., An update about recent developments of the PHITS code, Adv. Space Res. 45, 892-899 (2010).

[24] L. Sihver, T. Sato, M. Puchalska, and G. Reitz, Simulations of the MATROSHKA experiment at the International Space Station using PHITS, Radiat. Environ. Biophys. 49:351-357 (2010).

[25] **K. Gustafsson, L. Sihver, D. Mancusi, T. Sato**, PHITS simulations of the Matroshka experiment", Adv. Space Res.46, 1266–1272 (2010).

[26] L. Sihver, M. Puchalska, T. Sato, T. Berger and G. Reitz, Monte Carlo simulations of MATROSHKA experiment outside ISS, IEEEAC paper 1585, ISSN: 978-142447350-2 (2011).

[27] T. Sato, et al., "Evaluation of Dose Rate Reduction in a Spacecraft Compartment due to Additional Water Shield", ISSN 0010_9525, Cosmic Research, Vol. 49, No. 4, pp. 319–324. © Pleiades Publishing Ltd. (2011).

[28] L. Sihver et al., Simulations of absorbed dose on the phantom surface of MATROSHKA-R experiment at the ISS, Adv. Space Res. 49, 230-236 (2011).

[29] M. Puchalska, L. Sihver, T. Sato, T. Berger, and G. Reitz, Simulations of MATROSHKA experiments at ISS using PHITS, Adv. Space Res. (accepted).

[30] T. Sato, Y. Kase, R. Watanabe, K. Niita, and L. Sihver, Biological Dose Estimation for Charged-Particle Therapy Using an Improved PHITS Code Coupled with a Microdosimetric Kinetic Model", Rad. Res. 171, 107-117 (2009).

[31] T. Sato et al., Analysis of Cell-Survival Fractions for Heavy-Ion Irradiations Based on Microdosimetric kinetic Model Implemented in the Particle and Heavy Ion Transport code System, Rad. Prot. Dos., 1-6, 2010.

---

[1] M. Durante and F. Cucinotta, "Physical basis of radiation protection in space travel," Rev. Mod. Phys. 83, 1245–1281 (2011).