PHITS

Particle and Heavy Ion Transport code System

User's Manual

Ver. 3.35

English version

Preface

This manual is the Particle and Heavy Ion Transport code System (PHITS) user's guide. PHITS is a general-purpose Monte Carlo particle transport simulation code that is used in many studies in the fields of accelerator technology, radiotherapy, space radiation, etc. For details on the physical models and important functions implemented in PHITS, see the main article [1], benchmark studies [2] [3], and papers citing them. This manual explains how to execute PHITS and which parameters should be used in the system.

The contents of this manual correspond to the PHITS version number shown on the title page and are subject to change without notice. If you have any question or comment regarding this manual, please contact the PHITS office (phits-office@jaea.go.jp).

[1] T. Sato, Y. Iwamoto, S. Hashimoto, T. Ogawa, T. Furuta, S. Abe, T. Kai, Y. Matsuya, N. Matsuda, Y. Hirata, T. Sekikawa, L. Yao, P.E. Tsai, R.N. Hunter, H. Iwase, Y. Sakaki, K. Sugihara, N. Shigyo, L. Sihver and K. Niita, Recent improvements of the Particle and Heavy Ion Transport code System - PHITS version 3.33, J. Nucl. Sci. Technol. 61, 127-135 (2024).

[2] Y. Iwamoto, T. Sato, S. Hashimoto, T. Ogawa, T. Furuta, S. Abe, T. Kai, N. Matsuda, R. Hosoyamada, and K. Niita, Benchmark study of the recent version of the PHITS code, J. Nucl. Sci. Technol. 54:5, 617-635 (2017).

[3] Y. Iwamoto, S. Hashimoto, T. Sato, N. Matsuda, S. Kunieda, Y. çelik, N. Furutachi and K. Niita, Benchmark study of particle and heavy-ion transport code system using shielding integral benchmark archive and database for accelerator-shielding experiments, J. Nucl. Sci. Technol. 59, 665-675 (2022).

Contents

1 Recent Improvements and Development Members **1**
- 1.1 Recent Improvements . . . 1
- 1.2 Development members . . . 21
- 1.3 Reference of PHITS . . . 21

2 Installation and execution of PHITS **23**
- 2.1 Installation and execution on Windows OS . . . 23
  - 2.1.1 Installation on Windows OS . . . 23
  - 2.1.2 Execution on Windows OS . . . 24
    - 2.1.2.1 Execute using "send to → PHITS" . . . 24
    - 2.1.2.2 Execute via Command Prompt . . . 24
  - 2.1.3 Recommended software . . . 24
- 2.2 Installation and Execution on Mac OS . . . 25
  - 2.2.1 Installation on Mac OS . . . 25
  - 2.2.2 Execution on Mac OS . . . 25
    - 2.2.2.1 Execute by drag and drop . . . 25
    - 2.2.2.2 Execute via Terminal . . . 26
  - 2.2.3 Installation procedure using `Manual` mode in "PHITS_Installer" . . . 27
  - 2.2.4 Installation procedure without using "PHITS_Installer.command" . . . 28
- 2.3 Installation and Execution on Linux OS . . . 28
  - 2.3.1 Installation on Linux OS . . . 28
  - 2.3.2 Execution on Linux OS . . . 29
    - 2.3.2.1 Execute by specifying an executable file without using a shell . . . 29
- 2.4 How to change the executable file . . . 30

3 Restart and terminate PHITS calculations **31**
- 3.1 Restart PHITS calculations . . . 31
- 3.2 Terminate the PHITS code . . . 31

4 General format of input and output files **33**
- 4.1 Sections . . . 34
- 4.2 Reading control . . . 35
- 4.3 Inserting files . . . 36
- 4.4 User-defined variables . . . 36
- 4.5 User-defined characters . . . 37
- 4.6 Using mathematical expressions . . . 38
- 4.7 Particle identification . . . 38
- 4.8 Summary output file (phits.out) . . . 41
- 4.9 Tally output file . . . 44

5 Section format **46**
- 5.1 [Title] section . . . 46
- 5.2 [Parameters] section . . . 47
  - 5.2.1 Calculation mode . . . 47
  - 5.2.2 Number of history and Bank . . . 48
  - 5.2.3 Cut-off and switching energies . . . 51
  - 5.2.4 Cut-off time, cut-off weight, and weight window . . . 56
  - 5.2.5 Stopping power models . . . 57
  - 5.2.6 Elastic scatterings, total cross sections, and reaction cross sections . . . 58
  - 5.2.7 Decays and de-excited process models . . . 59
  - 5.2.8 Event generator mode and kerma approximation . . . 60
  - 5.2.9 Adjoint (backward-in-time) mode . . . 61
  - 5.2.10 Nuclear reaction model options . . . 62
  - 5.2.11 Options for low energy neutrons . . . 63
  - 5.2.12 Photon transport based on the PHITS original model . . . 63

5.2.13 Parameters for EGS5 . . . 64
5.2.14 Photon- and muon-induced reaction models . . . 67
5.2.15 Muon production model . . . 68
5.2.16 Neutrino-induced reaction model . . . 69
5.2.17 Options for Coulomb diffusion, gravitational field, and electromagnetic field . . . 70
5.2.18 Output options (1) . . . 72
5.2.19 Output options (2) . . . 73
5.2.20 Output options (3) . . . 74
5.2.21 Output options (4) . . . 75
5.2.22 Voxel and tetrahedron geometry options . . . 76
5.2.23 About geometrical errors . . . 78
5.2.24 Input-output file name . . . 79
5.2.25 Others . . . 80
5.2.26 Dumpall option . . . 81
5.2.27 Event Generator Mode . . . 85
5.3 [Source] section . . . 86
5.3.1 <Source> : Multi-source . . . 87
5.3.2 Common parameters . . . 88
5.3.3 Cylinder distribution source . . . 90
5.3.4 Rectangular solid distribution source . . . 90
5.3.5 Gaussian distribution source (x,y,z independent) . . . 91
5.3.6 Generic parabola distribution source (x,y,z independent) . . . 91
5.3.7 Gaussian distribution source (x-y plane) . . . 92
5.3.8 Generic parabola distribution source (x-y plane) . . . 92
5.3.9 Sphere and spherical shell distribution source . . . 93
5.3.10 Phase-space source based on accelerator beam emittance . . . 94
5.3.11 s-type = 12 . . . 96
5.3.12 Cone shape . . . 97
5.3.13 Triangle prism shape . . . 98
5.3.14 xyz-mesh distribution source . . . 99
5.3.15 Tetra-mesh source . . . 101
5.3.16 Surface definition source . . . 102
5.3.17 Reading dump file . . . 103
5.3.18 User definition source . . . 106
5.3.19 Definitions for energy distribution . . . 109
5.3.19.1 Continuous energy distribution with integral value . . . 110
5.3.19.2 Continuous energy distribution with differential value . . . 112
5.3.19.3 Discrete energy distribution . . . 113
5.3.19.4 Energy distribution by free format . . . 114
5.3.19.5 Gaussian and Maxwellian energy distributions . . . 116
5.3.19.6 Energy distribution defined by analytical functions . . . 117
5.3.19.7 Energy distribution of RI source . . . 118
5.3.19.8 Energy distribution of result of tally . . . 121
5.3.19.9 Cosmic-ray source mode . . . 122
5.3.20 Definitions for angular distribution . . . 126
5.3.21 Definition for time distribution . . . 128
5.3.22 Example of multi-source . . . 130
5.3.23 Duct source option . . . 134
5.4 [Material] section . . . 137
5.4.1 Formats . . . 137
5.4.2 Element (nuclide) definition . . . 137
5.4.3 Composition ratio definition . . . 137
5.4.4 Material parameters . . . 138
5.4.5 Direct specification of stopping power . . . 138
5.4.5.1 Format of stopping-power database . . . 139
5.4.6 $S(\alpha,\beta)$ settings . . . 139
5.4.7 Chemical form specification for track structure calculation . . . 139

5.4.8 Examples . . . 140
5.5 [Surface] section . . . 141
5.5.1 Formats . . . 141
5.5.2 Examples . . . 144
5.5.3 Surface definition by macro body . . . 154
5.6 [Cell] section . . . 155
5.6.1 Formats . . . 155
5.6.2 Description of cell definition . . . 156
5.6.3 Universe frame . . . 158
5.6.4 Lattice definition . . . 159
5.6.5 Repeated structures . . . 162
5.6.5.1 LIKE *n* BUT cell parameter . . . 162
5.6.5.2 Nesting structure with lattice . . . 164
5.6.5.3 Voxel phantom . . . 166
5.6.5.4 Use of tetrahedron geometry . . . 167
5.7 [Transform] section . . . 171
5.7.1 Formats . . . 171
5.7.2 Mathematical definition and usage of transformation . . . 171
5.7.2.1 The case of $M = \pm 1$ . . . 171
5.7.2.2 The case of $M = \pm 2$ . . . 172
5.7.2.3 The case of $M = \pm 3$ . . . 173
5.7.3 The case of tally . . . 174
5.8 [Temperature] section . . . 175
5.9 [Mat Time Change] section . . . 176
5.10 [Magnetic Field] section . . . 177
5.10.1 Charged particle . . . 177
5.10.2 Neutron . . . 178
5.10.3 Implementation of the magnetic field map . . . 179
5.10.4 Format of magnetic field map file . . . 180
5.10.4.1 Header format . . . 180
5.10.4.2 Data format . . . 181
5.11 [Electro Magnetic Field] section . . . 183
5.11.1 Uniform electric and magnetic fields . . . 183
5.11.2 Use of electric field map . . . 183
5.12 [Delta Ray] section . . . 185
5.13 [Track Structure] section . . . 186
5.14 [Super Mirror] section . . . 188
5.15 [Elastic Option] section . . . 189
5.16 [Data Max] section . . . 190
5.17 [Frag Data] section . . . 191
5.18 [Importance] section . . . 194
5.19 [Weight Window] section . . . 195
5.20 [WW Bias] section . . . 196
5.21 [Forced Collisions] section . . . 199
5.22 [Repeated Collisions] section . . . 200
5.23 [Volume] section . . . 201
5.24 [Multiplier] section . . . 202
5.24.1 Multiplier subsection . . . 203
5.25 [Mat Name Color] section . . . 206
5.26 [Reg Name] section . . . 208
5.27 [Counter] section . . . 209
5.28 [Timer] section . . . 211
5.29 [User Defined Interaction], [User Defined Particle] section . . . 212
5.29.1 Applications Examples of User-Defined Code . . . 213

6 General format of tally sections **215**
6.1 Mesh type . . . 215
6.1.1 Region mesh . . . 215
6.1.2 Definition of the region and volume for repeated structures and lattices . . . 216
6.1.3 r-z mesh . . . 218
6.1.4 xyz mesh . . . 219
6.1.5 tet mesh . . . 219
6.2 Energy mesh . . . 220
6.3 LET mesh . . . 220
6.4 Time mesh . . . 220
6.5 Angle mesh . . . 220
6.6 Mesh definition . . . 221
6.6.1 Mesh type . . . 221
6.6.2 `e-type=1` . . . 221
6.6.3 `e-type=2,3` . . . 222
6.6.4 `e-type=4` . . . 222
6.6.5 `e-type=5` . . . 222
6.7 Other tally parameters . . . 222
6.7.1 Particle parameter . . . 222
6.7.2 axis parameter . . . 223
6.7.3 samepage parameter . . . 224
6.7.4 file parameter . . . 224
6.7.5 resfile parameter . . . 224
6.7.6 unit parameter . . . 225
6.7.7 factor parameter . . . 225
6.7.8 output parameter . . . 225
6.7.9 info parameter . . . 225
6.7.10 title parameter . . . 225
6.7.11 ANGEL parameter . . . 226
6.7.12 Sangel parameter . . . 227
6.7.13 2d-type parameter . . . 227
6.7.14 gshow parameter . . . 228
6.7.15 rshow parameter . . . 228
6.7.16 x-txt, y-txt, z-txt parameter . . . 229
6.7.17 volmat parameter . . . 229
6.7.18 epsout parameter . . . 229
6.7.19 counter parameter . . . 230
6.7.20 resolution and line thickness parameters . . . 230
6.7.21 trcl coordinate transformation . . . 230
6.7.22 dump parameter . . . 230
6.8 Function to sum up two (or more) tally results . . . 232
6.9 Function to analyze two (or more) tally results . . . 234
6.9.1 User-defined anatally function . . . 235
6.9.2 Analysis function . . . 235
6.9.2.1 PHITS input file for transport calculations (phits.inp) . . . 236
6.9.2.2 input file for scripts of analysis function (autorun.inp) . . . 236
6.9.2.3 Procedure of scripts of analysis function . . . 236
6.9.2.4 Output format . . . 236
6.10 Extended statistical indicators . . . 238
6.10.1 How to use . . . 238
6.10.2 Calculation method and output examples for each statistical indicator . . . 239
6.10.3 Statistical check sheet . . . 243

7 Tally section format **244**
7.1 [T-Track] section . . . 244
7.2 [T-Cross] section . . . 249
7.3 [T-Point] section . . . 255
7.4 [T-Adjoint] section . . . 258
7.5 [T-Deposit] section . . . 261
7.6 [T-Deposit2] section . . . 267
7.7 [T-Heat] section . . . 270
7.8 [T-Yield] section . . . 274
7.9 [T-Product] section . . . 280
7.10 [T-DPA] section . . . 285
7.11 [T-LET] section . . . 288
7.12 [T-SED] section . . . 291
7.13 [T-Time] section . . . 295
7.14 [T-Interact] section (formerly named [T-Star]) . . . 298
7.15 [T-Dchain] section . . . 302
7.16 [T-WWG] section . . . 309
7.17 [T-WWBG] section . . . 314
7.18 [T-Volume] section . . . 317
7.19 [T-Userdefined] section . . . 319
7.20 [T-Gshow] section . . . 324
7.21 [T-Rshow] section . . . 326
7.22 [T-3Dshow] section . . . 328
7.22.1 Box definition . . . 331
7.22.2 3dshow example . . . 332
7.23 [T-4Dtrack] section . . . 335

8 Processing dump file **338**

9 Region error check **339**

10 Compilation of PHITS **340**
10.1 Compilation on Windows OS . . . 340
10.1.1 Compilation using Microsoft Visual Studio with Intel Fortran . . . 340
10.1.2 Compilation using makefile . . . 340
10.2 Compilation on Mac OS . . . 341
10.3 Compilation on Linux OS . . . 341
10.4 Compilation with KURBUC . . . 342
10.5 Array sizes . . . 343

11 Additional explanation for parallel computing **344**
11.1 Distributed memory parallel computing . . . 344
11.1.1 Installation of MPI protocol . . . 344
11.1.2 Execution of MPI version of PHITS on Windows . . . 344
11.1.3 Execution of MPI version of PHITS using command line . . . 345
11.1.4 Adjustment of maxcas and maxbch . . . 345
11.1.5 Treatment of abnormal ending . . . 345
11.1.6 Output file name of dump, dumpall, [t-userdefined] . . . 345
11.1.7 Read-in file definition in PHITS . . . 345
11.2 Shared-memory parallel computing . . . 345
11.2.1 Execution . . . 345
11.2.2 Important notices for shared-memory parallel computing . . . 346

12 FAQ **347**
12.1 Questions related to parameter setting 347
12.2 Questions related to errors occurred in compiling or executing PHITS 348
12.3 Questions related to Tallies 350
12.4 Questions related to source generation 352
12.5 Questions related to PHIG-3D 352
12.6 Other questions 352

APPENDIX **354**
A List of physical processes that cannot be handled by PHITS 354
B Evaluated nuclear data libraries 355

index **357**

1 Recent Improvements and Development Members

1.1 Recent Improvements

Major upgrade features after version 3.02 are described below.

From ver. 3.35, the following function has been implemented, and some bugs have been fixed. (2025/03/28)

- The activation cross-section of JENDL-5 has been converted into the DCHAIN and `ndata` formats, and they are stored in the "dchain-sp/data" and "XS/yield" folders, respectively. Additionally, in accordance with this update, the default values of `ndata` in `[t-yield]` and `[t-dchain]` have been changed to 2. This improvement was performed under the support of Dr. N. Iwamoto of Japan Atomic Energy Agency.
- A particle navigation function and a low-energy unbiased method have been introduced to `[t-wwg]`. Furthermore, the introduction of a new parameter named `pedestal` helps prevent excessive particle splitting.
- The `#all` command has been added to the `[cell]` section, enabling the easy definition of the region that serves as the background for particle transport, such as vacuum and air. Consequently, some lecture notes, including lec01 and snowman, have been revised.
- The sum-over values for almost all tallies are correctly derived and made it possible to output their statistical errors.
- When performing two-step calculations using a dump source, a new option has been introduced in the dump parameter so that the dump data IDs are automatically read from the header file. These three improvements were performed under support of NAIS Inc.
- PHIG-3D has been improved to be capable of visualizing the tally results with `xyz` mesh. Additionally, a rendering function based on the Polygon Boolean method has been introduced, which reduces memory usage (excluding geometries with many curved surfaces). These functions were developed under the support of Dr. S. Ohnishi of National Maritime Research Institute (NMRI).
- A chemical simulation code dedicated to PHITS (PHITS-Chem) has been developed. It enables the calculation of the G values for various types of radiation used in radiotherapy.[4] The code is distributed at "utility/usrtally/ChemCode." We also developed a function to visualize the kinetics of calculated radicals using PHIG-3D. Furthermore, by adopting a space partitioning algorithm, calculation time has been reduced compared to the first version.
- Sample input files for reproducing neutron sources based on α-emitters has been added to "sample/source/NeutronSource/Preci model".
- ExPORT-PHITS[5] (Excel Program for Integration of Organ dose rates calculated by RT-PHITS) has been developed and stored in "utility/RT-PHITS" folder.
- The source code has been revised to be compatible with the new Intel compiler (ifx). In addition, Windows project files for each executable file (single, OpenMP, MPI, and debug version) have been developed and stored in the bin folder. Please see "document/Install-IntelFortran-OneAPI-en.pdf" in more detail.
- The compiler used to create executable files for Mac has been changed from ifort to gfortran, due to the transition from Intel CPUs to Apple Silicon processors.
- The track structure model for protons and ions, ITSART has been updated. In addition to ionization handled in earlier version, Rutherford scattering and electronic excitation are considered. Moreover, accuracy for ion transport has been significantly improved.
- The `mother` parameter has been introduced for `[t-deposit]` and `[t-dpa]`, making it possible to calculate kerma and DPA for each target nucleus. Accordingly, we have significantly modified the recommendation settings and lecture notes for BNCT dosimetry.

[4] Matsuya et al., Phys. Chem. Chem. Phys. (2025), DOI: 10.1039/d4cp04216f
[5] T. Sato et al., EJNMMI Physics (2025), DOI: 10.1186/s40658-025-00743-6

- In the cosmic-ray source mode, we have added a mode that generates a source from a plane (`s-type`=1 and 2). We have also improved the accuracy in calculating the angular distribution of ground-level muon flux (from 1 to 45 degrees) and the proton flux at low altitudes (above 300 g/cm^2). Additionally, we have ensured that the Oct. 1989 event is correctly reproduced when `environ`=4 in SPE mode (previously, the Sep. 1989 event was being reproduced).
- The limitation on the number of elements in a single material when using the EGS mode (previously set to 20) has been removed. In addition, electron and positron transport using EGS in the elements Cm, Bk, Cf, and Es (atomic numbers 96, 97, 98, and 99, respectively) becomes available. These improvements were carried out with the cooperation of Dr. K. Furutaka from Japan Atomic Energy Agency and Dr. Y. Namito and Dr. H. Hirayama from High Energy Accelerator Research Organization.
- Interpolation method using 4-point (cubic) Lagrangian interpolation has been implemented for muliplier function in `[multiplier]`. This work was performed under the support of Dr. Y. Uwamino and Simplified Shielding Calculation Code Review Working Group, Radiation Science and Technology Subcommittee, Atomic Energy Society of Japan.
- A new function to import tetrahedral-mesh geometry in HDF5 format has been introduced to seamlessly treat data converted from CAD in near future.
- INC-ELF has been improved to calculate α-induced reactions. This work was performed under the support of Dr. Y. Uozumi and Mr. T. Furuta of Kyushu University.
- Changed the default output format in the `[ T-4Dtrack ]` tally to the T4D format and added several new parameters.
- Some activation cross section libraries included in DCHAIN have been updated to include the data for (n, n') reactions.
- Added application examples for `[User Defined Interaction]` (Muon pair production, Neutrino reaction, and Energy cutoff for each region).
- A new parameter, `lionprd`, has been introduced, allowing control over the types of secondary light ions generated from neutron-induced nuclear reaction simulated using nuclear data library without e-mode. Also, the `kerma` parameter, which had a similar function to this, has been removed.
- The default setting for checking input file consistency in restart, sumtally, and anatally calculations has been set to OFF (`ireschk`=1).
- A new function to check zero-volume tetrahedron has been included in the tetrahedral geometry check `itgchk`=1. OpenMP parallelization becomes available with tetrahedral-mesh geometry check mode `itgchk`=1.

- Introduction of `file(30)` parameter to freely specify the file name of the binary file for tetrahedron geometry with `itetra`=1 and 2.
- File specification for BDF format tetrahedral-mesh geometry has been changed with the file extension. The binary format has been also revised so the old binary files with `itetra`=2 needs to be recreated.
- A option to extract some tetrahedrons for tallies with `mesh=tet` has been introduced. This option is useful to save computational time and file sizes when tally results are requested only part of the tetrahedrons for tetrahedral mesh geometry composed by a large number of tetrahedrons.
- A new parameter `primary` has been introduced to `[T-product]` tally. When primary particles remain at the final state of atomic & nuclear reactions, you can choose to include or exclude them.
- In `[t-product]`, `part` parameter accepts isomers in a format, for example, `part=Na-24m In-116n` where `Na-24m` is first isomer and `In-116n` is second isomer. When nucleus with `g` is specified (e.g. `part=Na-24g`), only ground state products excluding isomers are tallied.
- A mesh option `mesh=tet` becomes available in `[weight window]` for tetrahedral-mesh geometry. Additionally the mesh option mesh=tet becomes available in `[T-WWG]` and `[T-WWBG]` for tetrahedral-mesh geometry. A mesh option `mesh=xyz` becomes available in `[T-WWBG]`.

- When defining a cone with a vertex using a TRC surface (a truncated conical surface), the value of R_2 is now automatically set to 1.0e-5.
- The anatally function has become applicable to [t-dchain].
- Added two angel parameters, cmap and ndis. This allows selecting colormaps for 2D plots and using discrete color bars.
- The lecture notes for ParaView have been updated and stored in the "lecture/advanced/paraview" folder. These materials were created with the cooperation of Dr. T. Fujibuchi of Kyushu University.
- The following bugs have been fixed.
  - Deposition energy calculated using kerma approximation for fissile nuclei.
  - Wrong frame transform of neutron emission angle and energy in photo-nuclear reactions has been corrected thanks to assistance by Dr. Cédric Jouanne, Mr. Louis Garnaud of CEA France, and Ms. Nathalie Labonnote of INSTN France.
  - Electron trajectory in electro-magnetic fields.
  - Angular distribution in ground-level muon fluxes for relatively small angle (1–45 deg).
  - 2-D geometry drawing function (seldomly occurred only in version 3.341).
  - [source] output function in the RI source format in DCHAIN.
  - Nuclear reaction induced by deuterons below 1 MeV/n.
  - Various bugs in track-structure modes.
  - Production in high-energy deuteron and α particle production above their dmax.
  - Restart calculation using [t-deposit] with unit=5.
  - Angular distribution in s-type=9 source.
  - Function to read nuclear data for meta-stable nuclides.
  - Event generator mode coupled with dir=-1 neutron source.
  - [t-cross] with samepage=z.
  - Function to read dump source data written in free ASCII format.
  - Several minor bugs in PHITS-Pad.
  - A bug where flux passed to Dchain was wrongly divided by volume when mesh=tet was selected in [T-Dchain] tally without [Volume] section in the PHITS input file has been fixed.
  - Erroneous unit conversion and formulization in NRF cross section calculation have been corrected thanks to assistance by Dr. Toshiyuki Shizuma of QST.

From ver. 3.34, the following function has been implemented, and some bugs have been fixed. (2024/03/26)

- The installer for Windows has been updated. Please see Sec. 2.1.1 for details. Support for 32-bit versions of Windows has been discontinued.
- An automatic download and setup program for JENDL-5 has been developed and included in "phits/XS/jendl5" folder. These two improvements were performed under support of AdvanceSoft Corporation.
- The files named "natural_abundance_20c.dat" (for JENDL-5) and "natural_abundance_50c.dat" (for JENDL-4) that specify stable isotopes to be expanded when natural elements are specified in the [material] section were included in the data folder. These files are selectively used according to the neutron library written at the beginning of the address file ("xsdir.jnd"). The mass table included in "xsdir.jnd" has also been updated. This work was performed under the support of Dr. N. Iwamoto of Nuclear Data Center, JAEA.
- A function to read neutron libraries for meta-stable nuclides has been implemented, though it is applicable only to JENDL-5.
- The executable files of PHITS in the debug mode, which checks for memory violations and uninitialized variables, have been included in the bin folder. They can be used when $DBG=1 is written before the 1st section of the input file.

- Output function of extended statistical indicators has been implemented. In addition to statistical errors, variance of variance (VOV), figure of merit (FOM), and probability density function (PDF) can be output for tally results of `[t-track]` and `[t-point]`. Please see Sec. 6.10 for details. This development was performed by the commissioned work from Nuclear Regulation Authority to Japan Atomic Energy Agency.

- PHITS-Chem code, which allows the simulation of the behavior of radicals generated by water radiolysis, has been included to the package. This code can be used in combination with the `[track structure]` section (currently only available for electron beams). For details, please see "utility/usrtally/ChemCode."

- A new track-structure modality, Electron Track Structure model applicable to ARbitrary Targets (ETSART)[6], has been implemented.

- A new function has been introduced in `[t-wwg]` to create `[weight windows]` tuned to reduce statistical errors in specific regions using the concept of the history counter.

- A new parameter `deltxyz` has been introduced to automatically fine-tune min/max values of `xyz`-mesh in `[weight window]` and `[t-wwg]` when they are defined as integers.

- An adjoint mode applicable to charged particles has been implemented. Please see "phits/utility/adjoint/charged" folder in more detail.

- A text editor specialized for making PHITS input file (PHITS-Pad) has been developed. Please see "phits/phitspad/manual" in more detail. This function was developed by Mr. K. Sakamoto and Mr. T. Imaki under the support of Center for Computational Science & e-Systems, JAEA.

- A new tally `[T-4Dtrack]` has been introduced. Particle track files output by this tally can be loaded in the "Particle tracks" tab of PHIG-3D, allowing for the creation of images and videos of the tracks.

- `[User Defined Interaction]` section has been introduced, allowing users to relatively easily implement their own interactions into PHITS.

- `[User Defined Particle]` section has been introduced, enabling users to relatively easily implement their own particles into PHITS. Using this feature, it is also possible to modify the properties of existing particles, such as their lifetime or decay patterns.

- The `rshow` option in tallies for `mesh=tet` has been implemented.

- A new option to output numbers defined as user-defined variables `c`*i* to `file(6)(D=phits.out)` has been introduced. The number is output as many times as specified by `ncvalout` in `[paraemters]`. It has also been extended to allow definitions up to `c999`.

- The output parameter of `[t-cross]` is now automatically determined based on the presence or absence of `a-type` and `e-type`.

- Reaction categories (`output`) for track-structure modes in `[t-interact]` have been reorganized.

- `[Forced collisions]` has become applicable to thick targets with variable energy within the target, even when charged-particle nuclear data libraries are used.

- A new parameter `mtinfo` has been introduced to output the multiplier values used in the `multiplier` subsection.

- MT numbers 1–3 (total reaction cross sections, elastic scattering cross sections, and inelastic scattering cross sections) are now available in the multiplier subsection even when charged particle or high-energy neutron libraries are used.

- Cross-section output mode (`icntl=1`) with `inucr=100` has become feasible to output data for all MT numbers (including negative values) that can be defined in the multiplier subsection. In addition, a new parameter `epsout` has been added to `[libout]` to generate an image (EPS) file of the output cross section data.

[6] Y. Hirata et al., Development of an electron track-structure mode for arbitrary semiconductor materials in PHITS, Jpn. J. Appl. Phys. 62, 106001 (2023).

- A sample input file for `[frag data]` for d-T reaction has been added to "phits/sample/fragdata/d-T" folder, which was developed based on Nishitani et al. [7]
- New ANGEL parameters (`xdec, xexp, ydec, yexp`) have been introduced to specify decimal and exponential types for representing axis values in EPS files.
- The following bugs have been fixed. The first 4 items occurred only after version 3.31 or 3.33.
  - The normalization factor was wrong in the case of `factor`< 0 for `[t-deposit]` with `unit=0`.
  - Positrons and their associated annihilation γ-rays were double counted when N-13, O-15, or F-18 was specified in the RI source with `proj=all`.
  - An error occurred when two or more string variables were used in the same line of an input file.
  - KURBUC mode did not work properly.
  - `[t-interact]` did not work properly when ITSART was used.
  - A scattered electron was counted as a produced particle in `[t-product]` when electron scattering occurs in EGS or track-structure mode.
  - The electron trajectory was shifted slightly after the electrons pass through the magnetic field region.
  - A bug in reflection boundary with tetrahedral-mesh geometry.

From ver. 3.33, the following function has been implemented, and some bugs have been fixed. (2023/09/01)

- The latest nuclear data library JENDL-5[8] for neutrons, protons, and deuterons has been included in the PHITS package, while JENDL-4.0/HE and JENDL/DEU-2020 were removed. Consequently, the default values of `lib(i)` parameter (suffix ID of the primary nuclear data library) have been changed to those for JENDL-5, including photo-nuclear and alpha-particle-induced nuclear reactions. In the case of proton and neutron libraries, the data for approximately 30 nuclei were selected to be included in the PHITS package because the file size of the complete dataset of JENDL-5 is too large - more than 100 GB. Please see "Appendix B Evaluated nuclear data libraries" in more detail. This work was performed under the support of Dr. C. Konno and Dr. K. Tada of the research group for Research Group for Reactor Physics and Thermal Hydraulics Technology and Dr. S. Nakayama of Nuclear Data Center, JAEA.
- With the partial introduction of JENDL-5, carbon is also expanded to natural abundance ratios. Previously, only carbon was not expanded to natural abundance ratios because JENDL-4 evaluated cross sections as natural carbon.
- A new function to scale the tally output value maximum to be *factor* when the `factor` parameters of corresponding tally is negative, has been implemented.
- A function to consider the detector resolutions has been implemented in `[t-deposit2]`.
- PHIG-3D has been updated to be applicable to input file with user-defined character variable. The opacity of each cell can be defined. These functions were developed under the support of Dr. S. Ohnishi of National Maritime Research Institute (NMRI).
- The maximum character length of the installation folder name (environmental variable PHITSPATH or `file(1)` parameter) has been extended to approximately 170.
- The default value of `iannih` has been changed to 1 for avoiding the double count of the contribution from annihilation gamma in the case of `proj=all`.
- `jmout` has been changed to be automatically set to 1 when `[t-dchain]` is defined.
- The maximum flight path for charged particles when their energy/angular straggling is considered, `deltc`, has become independent of the material density in the case that the density is below 1 g/cm^3. Owing to this change, the magnitude of energy/angular straggling in the air (or other gases) becomes slightly smaller in comparison to the previous versions.

[7] T. Nishitani, M. Ishikawa, M.-S. Cheon, S.-H. Hong, R. Rodionov, S. di Sarra, V. Krasilnikov, D. Gin; B. Coriton, "Neutronic simulations of in-vessel neutron calibrations for ITER neutron diagnostics by using simplified ITER model", J. Fusion Eng. Des., DOI: 10.1016/j.fusengdes.2023.113548

[8] Iwamoto O, Iwamoto N, Kunieda S, et al. Japanese evaluated nuclear data library version 5: JENDL-5. J Nucl Sci Technol. 2023;60(1):1-60.

- Nuclear reactions are disregarded when the track structure model KURBUC is used in order to precisely reproduce the results of original KURBUC.
- The following bugs have been fixed. Except for the first 2 items, these bugs occurred only after version 3.31.
  - Restart calculation did not work when more than two `multiplier` subsections were defined in `[t-cross]`.
  - Source locations were wrong when tetra-mesh source (`s-type=24`) was used in transformed tetrahedron geometry.
  - CPU time written in "phits.out" was two-fold overestimated.
  - Access violation occurred when a certain surface in microbody was used (see 5.85).
  - Access violation occurred when a negative value of `nx, ny, or nz` was specified in xyz-mesh type source (`s-type=22`).
  - Access violation occurred when KURBUC was used.
  - `jmode=-1` did not work in DCHAIN 3.31.
  - `reg` parameter in `[source]` did not work properly when two different values were assigned to the parameters in different multisource.
  - A bug for the geometry with large number repeated lattice structure (>10000 in one dimension) showing lost particles has been fixed. Also a bug showing lost particles for tetrahedral-mesh geometry has been fixed.
  - A bug that caused dotted lines to not be displayed on the page after the first page of eps files.

In version 3.32, the following bugs have been fixed. Note that the first 4 bugs were occurred only in version 3.31. (2023/05/16)

- A bug in generating neutrons using the RI source function has been fixed.
- A bug that occurred when `infl` command was used in `[surface]` and `[cell]` sections has been fixed.
- A bug in `[t-cross]` with `axis=t` has been fixed.
- A bug that rarely occurred when a lot of transformations are used in an input file has been fixed.
- A bug that occurred when `dmax(1)` was set to below 20 has been fixed.
- A bug that occurred when the incident deuteron or alpha particle energy is exactly the same as their dmax has been fixed.
- A bug in reading neutron data libraries for meta-stable nuclei has been fixed.
- A bug in calculating CPU time has been fixed. This bug was occurred only when the CPU time became longer than 24 hours.

From ver. 3.31, the following function has been implemented, and some bugs have been fixed. (2023/04/07)

- The gamma de-excitation model EBITEM has been upgraded to Ver.2. It uses EGAF(Evaluated Gamma Activation File) to calculate the gamma-rays produced by neutron capture. It also takes into account for atomic de-excitation subsequent to internal conversion. Unless `|igamma|` $\leq$1 is explicitly specified, Ver.2 is automatically activated.
- A new function to use character variables anywhere in PHITS input file has been introduced. Please see 4.5 User-defined character in more detail.
- A new parameter `samapage` has been introduced for selecting the data types shown in the same page of image output files. Please see 6.7.3 samapage in more detail. These two improvements were performed under support of NAIS Inc.
- A new option (`inucr=100`) to directly output the cross sections contained in nuclear data libraries has been implemented in the cross-section and kerma-factor output mode (`icntl=1`). Please see "phits/sample/icntl1" in more detail.

- RI source function (`e-type=28,29`) has become applicable to self-fission neutron sources.
- Computational time for transporting charged particles has been reduced by storing the maximum nuclear-reaction cross section of charged particles in memory. These three improvements were performed under the support of Dr. N.Furutachi of RIST.
- A new model has been developed in the `[t-sed]` tally, allowing the calculation of microdosimetric quantitates based on the track-structure mode incorporated in PHITS.
- User Defined Model (PHITS-UDM) has been developed, which Users can implement their own "interactions" and "particles."Please see `https://github.com/sakaki-y/PHITS-UDM` in more details.
- A new option `itgchk` has been introduced to check irregular intersections among tetrahedrons in tetrahedron geometries.
- Anatally function becomes feasible to analyze the results from different tallies.
- `[t-produce]` becomes feasible to score secondary particles generated using nuclear data libraries. These two improvements were performed by Mr. K. Sakamoto and Mr. T. Miura under the support of Center for Computational Science & e-Systems, JAEA.
- A bug that caused electron behavior to be affected by the electromagnetic field even after exiting the field has been fixed. This bug had occurred since version 3.25.
- A bug that occurred when using JENDL4 and other nuclear data (except JENDL-4.0/HE) simultaneously for the same nuclide has been fixed.
- A bug has been fixed in which the thermal motion of target nuclei was not calculated correctly when using `[t-point]`. This bug was occurred when thermal neutrons were scattered, and the tally result was off by up to 20% in the calculation of the energy-integrated quantities such as the effective dose.

From ver. 3.30, the following function has been implemented, and some bugs have been fixed. (2023/01/10)

- A new option, `negs=2`, has been introduced for the `negs` parameter. The maximum energies `dmax(12)`, `dmax(13)`, and `dmax(14)` for electrons, positrons, and photons are automatically set to 10 TeV when `negs=2`.
- The gamma-ray emission subsequent to neutron capture is simulated using EGAF(Evaluated Gamma Activation File). Previously, the evaluated data was used for nuclei with atomic number lower or equal to 32(Ge) whereas the theoretical calculation model EBITEM was used for the other nuclei. With this upgrade, PHITS uses the evaluated data for all the nuclei from Z=33 (As) to Z=83 (Bi), U-235 and U-238, in addition to the nuclei up to Ge, to precisely predict gamma-ray spectra of neutron capture reactions.
- A new option of `[t-cross]` and `[t-product]` to output the results as a function of LET has been implemented. This option was developed by Mr. K. Sakamoto and Mr. T. Miura under the support of Center for Computational Science & e-Systems, JAEA.
- The areal portion of DCHAIN's dose conversion factor units, controlled with the `idosunit` parameter (when using a nondefault value), has been changed to be more intuitive.
- The calculation methods for the ionization and excitation cross sections in ITSART have been improved to reproduce the stopping powers obtained from ATIMA.
- A new option (`inucr=16`) has been implemented in `icntl=1` to output the stopping power of any ions and material combination.
- The allocation method of the memory for storing the geometry and source information has been changed from static to dynamic. Due to this change, re-compiling PHITS with enlarged mdas parameter is not necessary anymore even when the PHITS simulation consuming extremely large memory size is performed. This change was mostly programmed by Mr. A. Murofushi and Dr. N. Furutachi of RIST.

- Sample input files for the ICRP adult voxel- and mesh-type reference computational phantoms are prepared and included in "sample/icrp" folder. These input files were created under supervisions of Dr. Akira Endo of JAEA for the voxel phantoms and Dr. Chan Hyeong Kim and Mr. Bangho Shin of Hanyang University for the mesh phantoms. These phantoms are distributed based on the approval from ICRP. You have to cite ICRP Publication 110/145 when you publish the results using the voxel/mesh phantoms, respectively.

- A function to detect geometry errors such as double definition and undefined region has been implemented in PHIG-3D. This function was developed under the support of Dr. S. Ohnishi of National Maritime Research Institute (NMRI).

- ANGEL has been improved to convert a tally output file to the eps file even when the vertical axis of the tally result has exceeds 20 digits. The ANGEL executable file has been updated.

- Instruction for importing CAD geometry into PHITS via tetrahedral mesh has been prepared and included in "utility/CADimport" folder.

- Two bugs in `[t-point]` for neutron transport have been fixed. The first one was occurred when γ-rays having anisotropic angular distribution (e.g. 4.4 MeV γ-rays emitted from carbon target) were tallied. In such cases, the intensities of the corresponding γ-rays were overestimated by several orders of magnitudes. The second one was occurred when the secondary neutrons produced by inelastic scattering of non-thermal neutrons were tallied. This bug disturbed the neutron spectra to some extent, but it had almost no influence (less than 1%) on the energy-integrated quantities such as the effective dose.

- A bug in calculating the *n*-*n* and *n*-*p* elastic scattering cross sections above 1 GeV in the case of `icxnp=1` has been fixed. This bug was introduced in version 3.25.

- A bug in geometry drawing of tetrahedral-mesh geometry with the `gshow` option has been fixed.

From ver. 3.29, the following function has been implemented, and some bugs have been fixed. (2022/09/12)

- The angular distribution of ground-level muons above 1 TeV has been improved in the cosmic-ray source mode.

- The sampling algorithm of secondary electron energies in the electron track structure mode has been improved.

- A new function to reconstruct dose distribution in carbon ion radiotherapy (CIRT) using treatment planning data has been developed in RT-PHITS. An original graphical user interface (GUI) has been also developed for RT-PHITS, which enables intuitively parameter setting, visualization of geometry of beam devices and patient body, evaluation of dose distributions. Please see "/phits/utility/RTphits" in more details.

- A new function to detect and show double-defined or undefined regions has been developed in PHIG-3D.

- A new function to tally a result without contribution from the specified particle in `part` parameter by putting a minus sign before the particle name has been developed.

- Installation and compilation manual of PHITS using OneAPI Intel Fortran Compiler ("Install-IntelFortran-OneAPI-en.pdf") has been prepared and included in "document" folder. Since OneAPI Intel Fortran Complier is freely available and it can build PHITS with much shorter computational time in comparison to gfortran, we recommend to use Intel Fortran for compiling PHITS and stop the support of compiling PHITS using gfortran.

- A Bug in the tallies for calculating history-by-history information, i.e. `[t-deposit]` with `output=deposit`, `[t-deposit2]`, and `[t-interact]` with `MorP=Prob`, have been fixed. This bug did NOT influence the tally results when the weights of particles were always 1.0. For `[t-deposit]` and `[t-deposit2]`, only the uncertainties of the tally results were wrong, while both tallied quantities and uncertainties were wrong in `[t-interact]`.

- A bug in generating dump source using Windows OpenMP version has been fixed. This bug did NOT influence the tally results when PHITS calculation properly finished.

- A bug in the RI source mode by setting `part=all` for alpha emitter has been fixed.

- A bug in the 2D and 3D geometry drawing with `gshow` with tetrahedral-mesh geometries has been fixed.

From ver. 3.28, the following function has been implemented, and some bugs have been fixed. (2022/05/09)

- `[transform]` has become effective for ellipse torus and torus surfaces.
- ITSART, the track-structure model for arbitrary materials, considers molecular excitation of water.
- ITSART calculates Rutherford scattering by Thomas-Fermi model, which effectively considers scattering angle cut-off by taking into account for screening by orbital electrons. The scattering angle cut-off specified by `ruth_min` (0.01 radian by default) is not necessary anymore.
- A bug in reading the new format of thermal scattering libraries has been fixed.
- Several bugs in `[t-yield]` (the problems in the MPI version, isomer production calculations, and in the cases of axis ≠ chart or dchain) introduced in version 3.27 have been fixed.
- A bug in reading fission yield data in DCHAIN-PHITS introduced in version 3.27 has been fixed.
- Erroneous energy deposition and ion range overprediction, happened when both `[t-deposit]` and track-structure mode was used, has been corrected.
- Several bugs in the anatally mode have been fixed.

From ver. 3.27, the following function has been implemented, and some bugs have been fixed. (2022/03/22)

- Neutral Kaons decay as K-long (K_L^0) or K-short (K_S^0).
- The generalized track structure mode has been upgraded and renamed as ITSART (Ion Track Structure calculation model for Arbitrary Radiation and Targets). ITSART (1) is about 100 times faster than ITS, (2) consumes less memory in thick targets by flipping particle transport sequence, (3) considers Auger electrons and atomic de-excitation X-rays. These improvements are applied to cells whose `mID` is set as -1 in the `[track structure]` section.
- A function to read photo-nuclear data libraries in ACE format has been implemented. A new parameter to specify the maximum energy of the photo-nuclear data library, `dpnmax`, has been introduced in `[parameters]` section. This development was performed under support of NAIS Inc.
- A function to automatically determine the maximum library energy of each nucleus from the information given in address file ("xsdir") has been implemented. In addition, a new parameter `lib(i)` has been introduced in `[parameters]` to specify the extension name of the default nuclear data library file for particle type `i`. Owning to these improvements, it is not necessary to specified `[data max]` section by users if they want to use all high-energy nuclear data libraries such as JENDL-4.0/HE defined in their xsdir file.
- High-energy event generator mode has been developed, which determines the reaction point using the total cross section contained in the high-energy nuclear data libraries while the information on secondary particles is evaluated using the nuclear reaction models.
- A new parameter `mxnuclei` has been introduced in `[t-yield]`, which specifies the maximum number of nuclides whose yields are scored in the case of `axis=chart` or `dchain`. The default value of this parameter is 3,000, which is much lower than that actually assigned in the previous versions, i.e., 96,288. Consequently, the allocated memory size and computational time are dramatically reduced when `[t-yield]` or `[t-dchain]` with large mesh sizes are defined.
- The limitation of the number of particle types (`part`) specified in a tally was removed. Note that this limitation in the other sections such as `[counter]` does not change. Above five improvements were performed under the support of Dr. N. Furutachi of RIST.
- Treatment of thermal scattering libraries has been improved, with support for continuous energy thermal scattering libraries, mixed elastic thermal scattering libraries and libraries containing SANS models such as nanodiamonds. This improvement has been performed under the support of Dr. José Ignacio Marquez Damian at European Spallation Source. More information is available from elsewhere[9] , [10].

[9] J.I. Marquez Damian, D. DiJulio, V. Santoro, L. Zanini, G. Muhrer. "Improvements in thermal neutron scattering data sampling in PHITS", UCANS9 Meeting, Riken, Japan (2022).

[10] `https://git.esss.dk/spallation-physics-group/phits-tsl`

- The Low-Earth Orbit (LEO) mode has been added to the cosmic-ray source mode. A function to generate the terrestrial cosmic-rays to a certain direction has been implemented.
- A new option (`M=±3`) has been implemented in `[transform]`.
- The parameter `iMeVperU` became effective in the cross-section output mode (`icntl=1`).
- A new option for ignoring the peak from Auger electrons has been implemented in `[t-sed]`.
- The parameter for specifying the source direction such as `dir` has become effective when the direction of each source particle is not specified in the case of `s-type=17` (dump source mode).
- The anatally function has become applicable to all tallies except for `[t-cross]` with `mesh=r-z`, `[t-adjoint]`, `[t-deposit2]`, `[t-heat]`, `[t-dchain]`, `[t-wwg]`, `[t-wwbg]`, `[t-volume]`, `[t-userdefined]`, `[t-gshow]`, `[t-rshow]`, `[t-3dshow]`. This function was developed by Mr. Takamitsu Miura of RIST, and was supported by Center for Computational Science & e-Systems, JAEA.
- The limitation of the shell numbers used in Doppler broadening of Compton scattering energies was extended from 79 to 500. Consequently, `iprofr` can be set to 1 (default) even when the number of elements per material is large.
- The deuteron nuclear data library JENDL/DEU-2020 in fragdata format was replaced by that in the ACE format. This development was performed under the support of Dr. S. Nakayama of Nuclear Data Center, JAEA. Please refer the paper[11] in your publication using the data files.
- Many sample input files have been added to "phits/sample" folder, such as cosmic-ray sources for various conditions and several neutrons sources including that generated by d-D reaction based on its fragdata prepared by Nishitani et al.[12]
- Bugs in `[t-yield]` occurred when `axis=xy` (restart and sumtally functions did not work properly) or `ndata=3` & `e-mode=0` have been fixed.
- The RI source generation function was improved to generate all kinds of radiation from decay (i.e., α-, β-, and γ-rays) when `proj=all` is set. (Note that neutrons from spontaneous fission are not considered.) This function was partially developed by Mr. T. Miura of RIST under the support of Center for Computational Science & e-Systems, JAEA.
- A new parameter `ndata` was added in `[counter]`. In the case of `ndata=2,3` in `[t-yield]` or `[t-dchain]`, double-counting of yields can be avoided by the `[counter]` setting.
- A rare bug when many transform matrixes were used in an input file has been fixed.
- A bug that stopped negative pions were not absorbed even in the case of `npidk=0` has been fixed. This bug was due to the change of cut-off energy of negative pions from 1 MeV to 1 keV in version 3.13.
- A bug in the cosmic-ray source mode for generating neutrons in the blackhole mode and ground-level muons has been fixed.
- A bug in `[elastic option]` introduced in version 2.85 has been fixed.
- Several bugs in PHIG-3D have been fixed.
- Electron track structure mode (etsmode) has been slightly modified. We changed the cross sections above 100 keV. Please be careful to simulation results of ionization and excitation events. The results obtained from this version are different from previous ones.
- A bug in the positron transport simulation has been fixed. In the previous versions, some positrons do not emit annihilation gamma-rays when they are stopped in a material.

From ver. 3.26, the following function has been implemented, and some bugs have been fixed. (2021/09/01)

[11] S. Nakayama, O. Iwamoto, Y. Watanabe, and K. Ogata, JENDL/DEU-2020: deuteron nuclear data library for design studies of accelerator-based neutron sources, J. Nucl. Sci. Technol., 805-821 (2021), DOI:10.1080/00223131.2020.1870010.

[12] T. Nishitani, S. Yoshihashi, K. Ogawa, M. Miwa, S. Matsuyama, A. Uritani, "Neutron yield calculation of thin and thick d-D targets by using PHITS with frag data table", J. Nucl. Sci. Technol., DOI:10.1080/00223131.2021.1981475

- A function to read nuclear data libraries in ACE format for deuteron and alpha particle has been implemented. Note that `[data max]` section is not applicable to those libraries. Thus, the nuclear data libraries for all nuclides used in your input file must be prepared even when you want to use the library for a certain nuclide. The libraries can be download from the websites of nuclear data, such as JENDL[13] and TENDL[14]. This improvement has been performed under the support of ASTOM R&D.

- A problem of "Insufficient virtual memory", which was observed only in version 3.25, has been fixed. In addition, a parameter to control the relative size of memory space allocated for nuclear data, `xsmemory`, was introduced in `[parameters]` section.

- The user defined tally for calculating DNA damage yields contained in "/phits/utility/usrtally/DNAdamage" has been improved to be capable of estimating the complex DSB yield.

- Track structure mode for arbitrary materials has included Rutherford scattering. This improvement applies to the cells whose `mID` is -1 in `[track structure]` section.

- Treatment of escaping protons has been changed in the SCINFUL mode.

- A bug when the dump source mode was used in the OpenMP version has been fixed.

- Calculation method and output unit of arc-DPA in `[t-DPA]` tally has been revised.

- A bug of unnecessary memory check in the anatally mode has been fixed.

- A bug when the cosmic-ray source mode was used in the MPI version has been fixed.

- Several bugs in PHIG-3D have been fixed.

From ver. 3.25, the following function has been implemented, and some bugs have been fixed. (2021/07/09)

- The source codes of PHITS have been revised to be compatible with NVIDIA TOOL KIT (former PGI fortran). If you want to compile PHITS with NVIDIA TOOL KIT, please set `ENVFLAGS=LinNVIDIA` in "makefile". This revision was made by Dr. M. Yamaguchi of QST Takasaki and Mr. N. Shirai of Hewlett Packard Japan Inc.

- The memory for containing nuclear data has been changed from static to dynamic one so that PHITS can read extremely large nuclear data without increase `mdas` parameter. This improvement has been performed under the support of ASTOM R&D.

- DCHAIN-PHITS has been improved to be capable of calculating the fission yields induced by neutrons below 20 MeV. For this improvement, several fission yield data libraries are added in "dchain-sp/data" folder. In addition, a new parameter `ilchain` has been introduced to adjust the maximum chain length independently from the maximum chain number. Please see `dchain-sp\manual` in more detail.

- A new parameter `icxnp` has been introduced to select the database of n-p scattering cross sections up to 1 GeV. In the default setting, a new database taken from JENDL/HE-2007 is selected. This improvement has been performed under the support of Dr. D. Satoh of Research Group for Radiation Transport Analysis, JAEA.

- A track structure mode applicable to arbitrary incident ion/material combination has been developed. This mode is applied by assigning `mID` to be -1 in `[track structure]` section. By default, this mode assumes that the target comprises a pure element or mixture of pure elements. By specifying the constituent molecules in the `[material]` section (e.g. `chem=CO2`), this mode takes into account for the molecular structure of the medium.

- New output options named `a-flux` and `oa-flux` have been implemented in `[t-cross]` for calculating angular differential fluxes.

- The OpenMP parallelization becomes available in the anatally calculation.

[13] `https://wwwndc.jaea.go.jp/ftpnd/jendl/jendl-deu-2020.html`
[14] `https://tendl.web.psi.ch/tendl_2019/tendl2019.html`

- Computational time of PHITS simulation using `[t-yield]` or `[t-dchain]` has been dramatically reduced.

- The maximum number of characters to be read in a line of "xsdir" has been extended from 80 to 130.

- The evaluated high-energy nuclear data libraries JENDL-4.0/HE for twenty-two nuclides stored in the PHITS package have been replaced with the files officially released by the research group for Reactor Physics and Standard Nuclear Code System, JAEA. The proton data files of ^{6}Li and ^{7}Li were added in the package. Please see B for more detail. This work and the modification of PHITS related with reading files were performed under the support of Dr. C. Konno of the research group for Reactor Physics and Standard Nuclear Code System.

- A bug in reading proton-Li data of JENDL-4.0/HE has been fixed.

- A bug when spatial size of `xyz` mesh is too large in `[weight window]` setting has been fixed.

- A bug in controlling particle weight when a source with weight $\neq 1$ is generated in a void cell has been fixed.

- A bug that rarely causes double defined region when a microbody surface is rotated has been fixed.

- Several bugs in PHIG-3D have been fixed.

From ver. 3.24, the following function has been implemented, and some bugs have been fixed. (2021/03/12)

- Estimation of doses from induced-activities by connecting calculation of PHITS using `[t-dchain]` with xyz mesh → DCHAIN-PHITS → PHITS becomes feasible. Please see "/phits/dchain-sp/sample/3-step_dose_xyz" in more detail. This function was partially developed by Mr. T. Miura of RIST under the support of Center for Computational Science & e-Systems, JAEA.

- The concept of history-counter, which is the maximum counter value of all particles generated in one history, has been introduced in tallies. When you define a constraint using this history-counter, you can tally your interested events selectively by tracing back to the generation of their source. Please see "/phits/sample/misc/history_counter" in more detail. This function was developed under the support of Dr. N. Furutachi of RIST.

- Functions of independently controlling source generations (`ibatch` in `[source]`) and summarizing tally results (`itall=4`) of each batch have been implemented. Using these functions, you can reduce the computational time when you need to run many simulations with different source conditions. These functions were developed under the support of Dr. N. Furutachi of RIST.

- A function to output the line number, the subroutine and file names where error or warning message is written in the source program has been implemented. This function is activated by setting `ierrout=1` in `[parameters]`. This function was developed by Mr. T. Imaki under the support of Center for Computational Science & e-Systems, JAEA.

- Parameters and equations become usable in the definition of multiplier subsections, except for equations using brackets.

- Sample programs of user-defined anatally for calculating the biological dose for targeted radionuclide therapy and the dose-mean LET have been provided in "/phits/utility/usranatal folder".

- `[frag data]` files of JENDL Deuteron Reaction Data File 2020 (JENDL/DEU-2020) [15] were contained in "/phits/XS/fragdata/". Deuteron-induced reactions on $Li^{6,7}$, Be^{9}, $C^{12,13}$ can be described well by the data files. This development was performed under the support of Dr. S. Nakayama of Nuclear Data Center, JAEA. Please refer the paper [16] in your publication using the data files. Furthermore, a sample input file for accelerator-based neutron source using the data file was contained in "/phits/recommendation/NeutronSource".

[15] `https://wwwndc.jaea.go.jp/ftpnd/jendl/jendl-deu-2020.html`

[16] S. Nakayama, O. Iwamoto, Y. Watanabe, and K. Ogata, JENDL/DEU-2020: deuteron nuclear data library for design studies of accelerator-based neutron sources, J. Nucl. Sci. Technol., 805-821 (2021), DOI:10.1080/00223131.2020.1870010.

- RT-PHITS (RadioTherapy package based on PHITS) has been developed by introducing various new functions into DICOM2PHITS, such as a module for converting PET-CT data to `[source]` section. Please see "/phits/utility/RTphits" in more detail. This development was performed under the support of Dr. Y.Koba, Dr. S.Yonai, Dr. S.Matsumoto, Dr. W.Chang of QST and Dr. T.Watabe and Mr. H.Sasaki of Osaka University.

- PHIG-3D (PHITS Interactive Geometry viewer in 3D) has been developed. PHIG-3D can visualize geometry in 3D by reading PHITS input file, and interactively enlarge, reduce, and rotate geometry by mouse operation. This software was mainly developed by Dr. S. Ohnishi of National Maritime Research Institute (NMRI), based on Gxsview [17]. NMRI holds the copyright of this software, and JAEA has the right to freely use, change, distribute, and relicense to third parties.

- Several bugs are fixed, including the problems in cosmic-ray source mode, `[frag data]`, electron transport in complex electro-magnetic fields. Bugs related to OpenMP and multiple lattice definitions introduced in version 3.23 are fixed. However, a bug in reading dump source file occurred in the Linux-OpenMP version has not been fixed yet.

- Bugs in sample user-defined tallies included in "/phits/utility/usrtally" have been fixed. These bugs occured only after version 3.22. In addition, your own user-defined tally may not work in tandem with the latest PHITS. In that case, please revise your user-defined tally program by referring "/phits/utility/usrtally/readme-en.docx" .

From ver. 3.23, the following function has been implemented, and some bugs have been fixed. (2021/01/12)

- A new source option for reproducing the cosmic-ray environments in space and in the atmosphere has been implemented. Please see Section 5.3.19.9 in more detail.

- The user-defined anatally function has been implemented for analyzing multiple tally results, using user-defined program written in "usranatal.f." As examples, the biological dose calculation programs based on the stochastic microdosimetric kinetic model1[18] have been developed. Please see Sec. 6.9.1 and `utility\usranatal` folder in more detail.

- Sources with energy distribution (`e-type`) can be set in the accelerator source mode (`s-type=11`).

- The photon reaction cross sections can be outputted using `icntl=1`. When `ipnint=1,2`, processes of photo-nuclear reactions are outputted.

- The maximum number of isotopes defined in `[material]` has been changed from 92 to 300.

- A warning message is shown when a particle is split into more than 10 particles by importance or weight window functions.

- The athermal-recombination-corrected dpa(arc-dpa) can be outputted using `idpa=1` in the `[t-dpa]` section. Details are in the paper[19].

- DCHAIN-PHITS has been improved by introducing several new parameters and increasing maximum number of materials in an xyz mesh from 10 to 100. Please see `dchain-sp\manual` in more detail. Furthermore, the new main DCHAIN reference paper[20] has been published.

- OpenMP parallelization performance has been improved by the modification of the source code such as reducing critical statements and the overhead of OpenMP thread private variables.

- Memory assignment of lattice strucutre data of geometry has been changed to dynamical allocation. Owing to this improvement, the re-compiling of PHITS to extend the memory space becomes not necessary for the calculation with a large memory voxel data.

[17] S. Ohnish, Gxsview: Geometry and cross section viewer for calculating radiation transport, SoftwareX, 14, 100681 (2021).
[18] T. Sato and Y. Furusawa, Radiat. Res. 178, 341-356 (2012).
[19] Y. Iwamoto, S. Meigo, and S. Hashimoto, J. Nucl. Mater. 538 (2020) 152261. `https://doi.org/10.1016/j.jnucmat.2020.152261`
[20] H. N. Ratliff, N. Matsuda, S. ichiro Abe, T. Miura, T. Furuta, Y. Iwamoto, and T. Sato, "Modernization of the DCHAIN-PHITS activation code with new features and updated data libraries," Nucl. Instrum. Methods Phys. Res., B, vol. 484, pp. 29-41, 2020. doi: 10.1016/j.nimb.2020.10.005. [Online]. Available: `https://doi.org/10.1016/j.nimb.2020.10.005`

- A bug when two (or more) lattice structures are used in an input file has been fixed. This bug occurs only in version 3.22.

- A bug in calculating particle trajectories in the magnetic or electromagnetic fields defined in the data list type with non-uniform meshes has been fixed.

From ver. 3.22, the following function has been implemented, and some bugs have been fixed. (2020/10/09)

- A new option has been introduced in the phase-space source based on accelerator emittance (`s-type=11`). Please see Section 5.3.10 in more detail. This development was performed under the support of Dr. Y. Sakaki of National Institutes for Quantum and Radiological Science and Technology (QST).

- A bug for ignoring the relativistic theory in the calculation of particle trajectory using magnetic field map has been fixed. This bug fix was supported by Mr. N.Tokunaga of Kyushu University.

- A bug in drawing the geometry with cell number has been fixed. This bug occurs only in version 3.21.

From ver. 3.21, the following function has been implemented, and some bugs have been fixed. (2020/09/14)

- A function to use the stopping power of liquid water, graphite, and dry air given in ICRU90 has been developed. For protons and carbon ions, the dedx files for those stopping powers were prepared and contained in `data\dedx`. For electrons, the database of the density correction factors used in the evaluation of ICRU90 was prepared, and it can be used by setting `epstfl=1` in the `[parameters]` section. Please see `utility\icru90` in more detail. This development was performed under the support of Dr. Y. Namito of KEK and Dr. M. Shimizu of AIST.

- A new function for calculating the response function of organic scintillator based on SCINFUL-QMD [21] has been developed. Please see `utility\usrtally\scinful-qmd` in more detail. This development was performed under the support of Dr. D. Satoh of Research Group for Radiation Transport Analysis, JAEA.

- An algorithm for adjoint Monte Carlo simulation has been implemented for efficiently estimating the photon fluences in a small region from sources distributed in a large area. Please see Section 5.2.9 as well as `utility\adjoint` in more detail. This development was performed under the support of Dr. A. Malins of the Center for Computational Science & e-Systems, JAEA.

- The evaluated high-energy nuclear data libraries JENDL-4.0/HE for twenty-two nuclides have been included in the PHITS package. Please see APENDIX for more detail.

- A function to output cross sections and kerma factors used in PHITS has been developed as a part of `icntl=1`. Please see `utility\icntl1` in more details.

- Iwamoto's fission model has been implemented for fission of sub-actinide (nuclide with atomic number less than 89). This function was developed under the support of Dr. H. Iwamoto of Japan Atomic Energy Agency, Research Group for Nuclear Transmutation System. For the future improvement of actinide fission model, the fission barrier table `Fission_barrier_table_PhysRevC_91_024310.dat` [22] is placed at `phits\data`.

- Gamma evaporation model EBITEM is upgraded to use up-to-date RIPL(Reference Input Parameter Library) -3 [23] nuclear structure data placed at `phits\data\RIPL`. In the old versions, the ENSDF nuclear structure data taken in 2012 was built in to the executable. The merits of this upgrade are, improvement of calculation accuracy, downsizing of PHITS executable, saving runtime memory requirement, and reduction of compilation time. The new version of EBITEM is automatically used without any extra operations.

- The weight biasing option was introduced in the xyz-mesh source (`s-type=22`). Using this option, PHITS generates equal numbers of particles from each mesh and expresses the source intensity by adjusting the particle weights.

[21] D. Satoh, T. Sato, N. Shigyo and K. Ishibashi, SCINFUL-QMD: Monte Carlo based computer code to calculate response function and detection efficiency of a liquid organic scintillator for neutron energies up to 3 GeV, JAEA-Data/Code, 2006-023 (2006).

[22] Peter Moller, Arnold J. Sierk, Takatoshi Ichikawa, Akira Iwamoto, and Matthew Mumpower, Fission barriers at the end of the chart of the nuclides, Phys. Rev. C 91, (2015) 024310.

[23] R. Capote, M. Herman, P. Oblozinsky, P.G. Young, S. Goriely, T. Belgya, A.V. Ignatyuk, A.J. Koning, S. Hilaire, V.A. Plujko, M. Avrigeanu, O. Bersillon, M.B. Chadwick, T. Fukahori, Zhigang Ge, Yinlu Han, S. Kailas, J. Kopecky, V.M. Maslov, G. Reffo, M. Sin, E.Sh. Soukhovitskii and P. Talou, Reference Input Parameter Library (RIPL-3), Nuclear Data Sheets, 110, Issue 12, (2009) 3107-3214

- DCHAIN-PHITS has been improved to output the statistical uncertainties by propagating those of the PHITS simulation. In addition, the mesh option `mesh=tet` becomes available in `[t-dchain]`.

- Analysis function using autorun was improved to estimate systematic uncertainties of results of `[t-deposit]` (`output=dose`). Furthermore, this function can be applied for two (or more) tally sections written in an input file. This function was developed by Mr. Takamitsu Miura of RIST, and was supported by Center for Computational Science & e-Systems, JAEA.

- An option to select share/non-share of tally variables for OpenMP shared-memory parallel computing has been implemented. A setting to reduce required memory by sharing tally variables among multi threads was chosen in older version of PHITS. However there were cases that computational speed became slow due to a bottle neck of a process required to prevent memory corruption. Thus a new option to select share/non-share of tally variables are introduced.

- A new parameter `gsline` was introduced in the `[parameters]` section to control `gslat` parameter in each tally in a lump. In addition, new options for `gslat` were added.

- A new parameter `icells` was introduced in the `[parameters]` section to read and write binary files of the entire `[cell]` section for reducing the computational time for reading complex geometry.

- For PHITS simulation using voxel data, echo of the voxel data is excluded from the input echo in the "phits.out" file to reduce output data file size and computational time. By setting an option `ivoxel=3` in the `[parameters]` section, you can obtain the voxel data echo as before.

- A new parameter `ichkmat` has been introduced to check material numbers and densities of each cell when different densities are defined for a given material number. When incorrect drawing of the geometry by gshow option occurs, this option may work well. Note that it increases the loading time of complex geometry with a lot of cells.

- Bugs in the use of proton nuclear data library have been fixed. In addition, a new parameter `epseudo` to control the energy for calculating pseudo reaction cross section has been introduced in `[parameters]` section. These revisions were performed under the support of Dr. N.Furutachi of RIST.

- Bugs in the electron motions in magnetic fields have been fixed. A bug in the consideration of the magnetic field map for z-direction has been fixed. A bug in reading proton nuclear data libraries has been fixed. If you used these functions in your PHITS simulations, we recommend to re-perform them. A bug in t-deposit tally for tetrahedral-mesh geometry using the next region material by mistake.

From ver. 3.20, the following function has been implemented, and some bugs have been fixed. (2020/03/19)

- Proton ($E < 300$ MeV) and carbon ion ($E < 10$ MeV/n) track structure codes, KURBUC[24], have been implemented in this version of PHITS code system. The implementation was performed under collaboration with Prof. Hooshang Nikjoo and Dr. Thiansin Liamsuwan who developed the KURBUC codes. The copyright of KURBUC codes belongs to Prof. Hooshang Nikjoo, and its source code is not included in the PHITS package.

- A function to consider the photon-induced muon-pair production has been implemented. This function was developed under the support of Dr. Y. Sakaki of KEK. See the paper [25] in more detail.

- A new section `[repeated collisions]` has been implemented to improve the statistics of rarely produced secondary particles. Please see Section 5.22 for more details. This function was developed under the support of Dr. H. Iwamoto of Japan Atomic Energy Agency, Research Group for Nuclear Transmutation System.

[24] Uehara S, Nikjoo H, Goodhead DT, 1993, Cross sections for water vapour for Monte Carlo electron Track structure from 10 eV to 10 MeV region. Phys Med Biol 38: 1841-1858;

Hooshang Nikjoo, Shuzo Uehara, Dimitris Emfietzoglou, Interaction of Radiation with Matter, CRC Press, Published June 11, 2012;

Liamsuwan T, Uehara S, Emfietzoglou D and Nikjoo H 2011, Physical and biophysical properties of proton tracks of energies 1 keV to 300 MeV in water, Int. J. Radiat. Biol. 87 141-60;

Liamsuwan T and Nikjoo H 2013 A Monte Carlo track structure simulation code for the full-slowing-down carbon projectiles of energies 1 keV u^{-1} - 10 MeV u^{-1} in water, Phys. Med. Biol. 58 673-701

[25] Y. Sakaki et al., Implementation of muon pair production in PHITS and verification by comparing with the muon shielding experiment at SLAC, Nucl. Instr. & Meth. A 977, 164323 (2020).

- A new parameter iangform has been introduced in [T-Cross] section to choose the reference for formed angle in a-type mesh. In addition to the normal vector to the surface (iangform=0), x-axis (iangform=1), y-axis (iangform=2) and z-axis (iangform=3) can be chosen.

- A new option opt=5 in [frag data] has been introduced: pointwise data of differential cross sections can be read by this option.

- A function to read the stopping power of each material from the user-supplied table has been implemented. As for the sample data, the stopping power of protons and α particles in various materials calculated by PSTAR and ASTAR[26] were prepared and included in "/phits/data/dedx"folder. This development was performed under support of NAIS Inc.

- A function to read electro-magnetic field maps written in xyz or r-z grid has been implemented. This development was performed under support of Advanced Software Inc.

- A new pseudo-random number generator xorshift64 has been implemented. This generator was used in the default settings. The period length of random numbers was extended from 2^{46} to $2^{64}-1$ by the new generator.

- 2-dimensional geometry drawing function, gshow, has been improved to properly depict complicated geometry defined by voxel or tetrahedron. This improvement was performed under support of NAIS Inc.

- Several activation cross section libraries used for DCHAIN-PHITS have been developed, and their hybrid library has been selected as the default library. In addition, the decrease of target nuclides due to nuclear reaction has been automatically considered by setting the default values of itdecs and itdecn to 1. Please see Section 7.15 for more detail. This development was performed under the support of principal researcher C. Konno of Japan Atomic Energy Agency, Research Group for Reactor Physics and Standard Nuclear Code System.

- A bug of providing particle energy in MeV/n instead of MeV/u with ATIMA was fixed. Owing to this revision, the range of charged particle slightly changed - approximately 1% at the maximum. In addition, the unit of ion energy in PHITS is not MeV/u but MeV/n, and the PHITS manual as well as input echo were revised accordingly.

- The power index of the Maxwellian energy distribution source (e-type=3, 7) has become changeable, and its default value was set to 0.5. Note that the former versions of PHITS set the power index to 1.0, but they could not properly reproduce the Maxwellian distribution.

- A new function was implemented to estimate systematic uncertainties of tally results based on analysis variance (ANOVA). Effects caused by errors of input information can be estimated by this function. Furthermore, script files autorun (autorun.bat & autorun.sh) were improved for analysis function in PHITS. The scripts successively execute PHITS by slightly varying input values. Please see Section 6.9.2 for more detail. This function was developed by Mr. Takamitsu Miura of RIST, and was supported by Center for Computational Science & e-Systems, JAEA.

- Documents for the connection calculation between PHITS and FLUENT were made.
  See files in "/phits/utility/FLUENT" in detail.

From ver. 3.17, the following function has been implemented, and some bugs have been fixed. (2019/10/29)

- A new option foamout=2 to output the tetrahedral-mesh tally results in csv format has been implemented.

- The bug in the [t-deposit] calculation using EGS5 was fixed. This bug was due to uninitialization of a certain variable in PHITS, so it occasionally causes a serious influences on the tally results, but in most case not.

- The calculation procedures for 'sum over' in many tallies have been revised.

- The bug in the motion of cut-off particles with energy straggling when nedisp=1 has been fixed.

[26] https://www.nist.gov/pml/stopping-power-range-tables-electrons-protons-and-helium-ions

- The parameters used in the biological dose calculation using "usrdfn2.f" were changed from those given in T.Sato `et al.` Radiat. Prot. Dosim. 143, 491-496 (2010) to T.Sato `et al.` Radiat. Res. 171, 107-117 (2009).

From ver. 3.16, the bug for `sangel` parameter has fixed. The other bug in ver. 3.15 that `[t-track]` tally does not output "`_err`" files has also been fixed. (2019/09/26)

From ver. 3.15, the following bugs have been fixed. (2019/09/12)

- A bug that `[t-deposit]` overestimates contributions of the energy cut-off in EGS5 has been fixed. Furthermore, another bug when using EGS5 and lattice structure has been fixed. The former and latter bugs occurred from version 3.13 and 3.10, respectively.
- A bug regarding the function when setting `itall=3` has been fixed. Segmentation errors occurred depending on input files. This bug occurred from version 3.13.
- A bug of DCHAIN, which occurred only in version 3.14, has been fixed.

From ver. 3.14, the following functions have been implemented, and some bugs have been fixed. (2019/08/18)

- DCHAIN-SP has been improved in terms of the following two features; (1) several neutron activation libraries and decay-data libraries have been developed based on the latest evaluated data, and (2) statistical uncertainties of the induced activities can be evaluated considering those of the production yields calculated by PHITS. Owing to these improvements, some parameters have been added to `[t-dchain]` section (see section Sec. 7.15 in detail). The new version of DCHAIN will be called DCHAIN-PHITS to be distinguished with the previous versions.
- A new function to calculate the DNA damage yields has been developed, using the track structure mode and user-defined tally. Their sample data are contained in "/phits/utility/usrtally/DNAdamage."
- Two bugs occurred only in version 3.13 have been fixed, which are related to `[t-interact]` and the track structure mode.

From ver. 3.13, the following functions have been implemented, and some bugs have been fixed. (2019/08/02)

- The default value of the cut-off energy for proton, pion, muon, and ions has been decreased from 1 MeV/n to 1 keV/n.
- A new source type (`s-type=26`) has been introduced to define a surface source which generates source particles on the surface(s) that has/have been pre-defined in the `[surface]` section. Please see Section 5.3.16 for more details.
- A new option, `itall=3`, has been implemented to plot the trend of tally results and their statistical uncertainties for each batch with the ongoing PHITS calculation. This option is available only for `[t-track]` and `[t-point]` at this moment.
- A new parameter, `istdcut`, has been introduced to the `[parameter]` section. It enables the termination of the tally calculations of which the statistical uncertainties have reached the `stdcut` setting. This function is particularly useful when the types of tallies which consume much longer computational time, such as `[t-point]` and `[t-sed]`, are written in the same input file with other tallies. Another new parameter, `istdbat`, has been introduced to inactivate `stdcut` at the early stage of the PHITS simulation.
- In stopping power calculations using ATIMA, the ionization potential of water is expected to automatically adjust to 75 eV. However, this function was valid only when the material of water was explicitly specified by the isotope 1H, i.e. material consists of natural abundant hydrogen is not treated as water in this particular respect. We modified the program, so that the material of water which is defined by naturally-abundant hydrogen isotopes, such as in the format of `H 2 O 1` and `H -0.1111 O -0.8889`, is now regard as water for stopping power calculations. Owing to this modification, the ranges of charged particles in water increase slightly (less than 1% in most cases) compared to the results obtained from the previous versions. The valid condition of `ih2o` parameter has been also revised in the same way.
- Some bugs have been fixed, such as ignorance of `stdcut` in the MPI version, strange behavior of particles when mirror surfaces are used in lattice structure, and a rare bug in EGS5.

From ver. 3.12, the following functions have been implemented, and some bugs have been fixed. (2019/06/20)

- The new parameters `cnt(1)`, `cnt(2)`, and `cnt(3)` have been introduced in `[source]` section to specify the initial counter values for each multi-source subsection <source>. The maximum number of <source> subsections is extended from 200 to 500.

- A function to consider the decrease of target radioactive nuclides due to nuclear reactions has been implemented in `[t-dchain]`. DCHAIN has also been revised in correspondance to the `[t-dchain]` improvement. Note that former versions of DCHAIN cannot read the outputs from the improved [t-dchain].

- The `itall = 1` option has become available in the Windows OpenMP version of PHITS.

- Some bugs regarding the FLUENT-PHITS coupled calculations have been fixed, including the incorrect translation of the material densities between FLUENT and PHITS.

From ver. 3.11, the following bugs have been fixed. (2019/05/16)

- The bug of MPI parallel calculations with tetrahedral-geometry stopped abruptly in ver. 3.10 has been fixed.

- The bug that the magnetic field maps given in `r-z` grid (`type = -2 or -4`) cannot be properly considered has been fixed.

- Annihilation photons are double counted in the RI source generation function when the production of both positron and photon are considered. To avoid the problem, a new parameter `iannih` has been introduced to exclude the annihilation photons in the case of `proj = photon`.

From ver. 3.10, the following functions were implemented, and some bugs were fixed. (2019/03/13)

- Procedure for coupling PHITS with thermal analysis software such as ANSYS Fluent is established. For this purpose, a function to read tetrahedral-mesh geometry written in bulk-data format of NASTRAN is implemented. A new mesh option, `mesh=tet`, is introduced in `[t-track]`, `[t-deposit]`, `[t-yield]`, `[t-product]`, and `[t-dpa]` to output the tallied quantities in each tetrahedral mesh. Field-data format used in OpenFoam, which can be directly read by thermal analysis software, has become selected as an output format.

- As a part of the improvements for coupling with thermal analysis software, a new option, `unit=5`, is introduced in `[t-deposit]` to output the deposition energy in J/m^3/source.

- Sample files for using high-energy nuclear data library JENDL-4.0/HE are provided in the PHITS package. Please see `\phits\recommendation\jendlHE` in more details.

- A function to read magnetic field maps written in xyz or r-z grid is implemented. This development was performed under support of Advanced Software Inc.

- `[t-dchain]` has become applicable to `mesh=xyz`. In reply to this improvement, DCHAIN is also updated. Note that the function to visualize the induced activities in `xyz` mesh has not been developed yet.

- A nuclear reaction model to handle the reactions by electron-, mu-, tau-neutrino and their antiparticles up to 150 MeV of incident energy is incorporated. Neutrinos can interact with orbital electrons, ^{1}H, and ^{2}H whereas the nuclei heavier than ^{2}H are currently out of the range of this model.

- INC-ELF, an intra-nuclear cascade model developed by Kyushu University, is updated to consider collective excitation of the target nucleus and to precisely calculate charged particle emission barriers.

- Electron track-structure mode is improved to consider the production of Auger electrons.

- User-defined energy resolution function has been implemented to `[T-deposit]` in order to reproduce energy resolution which cannot be realized by `dresol` and `dfano`. Users can define the functional forms of energy resolution in "usresol.f," which is activated by defining negative `dresol`. As a sample, the default "usresol.f" contains a numerical model proposed by Meleshenkovskii et al. to reproduce asymmetric peaks of a CdZnTe detector.

- A new option, `gshow=5` and `[t-gshow]` with `output=10`, is introduced to visualize the geometry in the pixel format. This option is useful for properly visualize the geometries with fine structure such as voxel and tetrahedral phantoms.
- Time for reading lattice structure such as voxel phantom data is reduced by introducing compressed format.
- Some parameters in the muonic atom cascade program have become adjustable.
- The third user-defined function, "usrdfn3.f," is introduced for weighting the results of `[t-deposit]`. The default program of usrdfn3.f can convert the absorbed dose to the water equivalent dose, which is frequently used in medical physics.
- Remaining batch number has become adjustable using "batch.out."
- In `[t-cross]` with `mesh=xyz` or `r-z`, a new option, `enclose=1`, is introduced for calculating incoming and outgoing particle fluxes (or currents) from each mesh.
- Deposition energies in the cells with the same ID but placed in different lattice coordinates can be separately analyzed in the case of `[t-deposit]` with `output=deposit`.
- Nuclear data libraries containing more than 1000 γ-ray spectra such as the latest version of ENDF and JEFF have become acceptable in PHITS.
- The fixed charge mode, `ifixchg=1`, is introduced for calculating the stopping power of particles with a certain charge state. Note the charge exchange reaction has not been implemented yet, so the fixed charge mode is appropriate only when target material is extremely thin at this moment.
- Several bugs are fixed, including the problem in positron transport when `emin(12)` is specified without specifying `emin(13)`, and that in electron and positron transports in the magnetic field defined at a lattice structure.

From ver. 3.08, the following changes have been made. (2018/08/20)

- The user-defined activation cross section data can be used in the `[t-yield]` and `[t-dchain]` sections. Please refer to Section 7.8 for further detail. This new function was supported by the R&D program entitled "Reduction and Resource Recycling of High-level Radioactive Wastes through Nuclear Transmutation" in the Impulsing Paradigm Change through Disruptive Technologies Program (ImPACT).
- For RI sources, characteristics x-ray and internal conversion electron are now considered.
- Tetrahedral geometries that are converted from CAD can now be used, even when the surface of tetrahedron does not perfectly match the boundary surfaces.
- The parameters (`c1-c99`) specified in the input file can be used in the user-defined source file ("usrsors.f").
- The high-speed mode of ANGEL, which does not check numerical data while plotting the results, has become the default.
- For cylindrical sources (`s-type=1`), a few bugs, such as the bug occurring when a source particle is sampled between 0 and `r1` with the setting of `r0=r1`, have been fixed.

From ver. 3.07, `mesh=xyz` became available in `[weight window]` for EGS5. Conversion coefficients for soft error rates on semiconductor devices were added to default data of `[multiplier]`. In addition, some bugs for JAM were fixed. (2018/07/05)

From ver. 3.06, the following functions were implemented, and some bugs were fixed. (2018/05/29)

- By defining `igamma` as negative, Doppler effect owing to the motion of the emitter nuclei is disregarded. Specific gamma-rays are observed as mono-energetic peak in the energy spectrum.
- GEM Ver.2 including nucleon-gamma emission competition was developed. This version is available by setting `ngem=2` in `[parameters]` section.
- New format of `[transform]` was implemented to define rotation around z,y,x axes simply.

- The distributed-memory parallel computing using a MPI protocol became available on Windows. This development was performed under support of NAIS Inc.

From ver. 3.05, the following functions were implemented, and some bugs were fixed. (2018/03/14)

- The function to calculate the deposition energy using kerma approximation, which used to be implemented only in `[t-heat]`, became available in `[t-deposit]`. Necessity of the use of kerma approximation in the energy deposition calculation is automatically judged based on the status of `e-mode` and `negs` specified in `[parameters]` section. Owing to this improvement, we recommend to use `[t-deposit]` instead of `[t-heat]` in all situations of deposition energy calculations.

- `[t-deposit]` with `output=deposit` can be used in the case that weight is not always 1 unless weights of energy depositing particles change within one history. Then, this mode works in the cases that you set `s-type=9` and `dir=-all` in `[source]` section, or you use `[forced collisions]` for neutral particles.

- The graph style for `[t-cross]` with `axis=z` is changed from histogram to dot & line.

- Specification of `z-type=1` and `nz=0` is allowed for `[t-cross]` to calculate fluences of particles passing through a certain surface.

- Numbers of delta-rays produced by ions (including protons) and knock-on electrons produced by electron scattering can be separately calculated in `[t-interact]` section. In addition, they can be distinguished in `[counter]` section too.

- RI source function was improved to produce Auger electrons even when the Auger electron spectrum data are not included in `RIsource.ack`.

- A bug in `[t-product]` in the case of `iMeVperu=1` was fixed.

From version 3.04, the probability density of the number of interactions by individual history can be outputted from `[t-star]`. The number of interactions occurred in the track-structure mode can be also counted in the tally. In addition, the tally name was changed to `[t-interact]` in order to explicitly state that it is the tally for counting the number of interactions, though the former name `[t-star]` can still be accepted. (2018/02/16)

From ver. 3.03, the following functions were implemented, and some bugs were fixed. (2018/02/01)

- A new function to generate source particles from tetrahedron geometry was implemented. See Section 5.3.15 in more detail.

- `mesh = xyz` became available in `[weight window]` and `[t-wwg]` sections.

- Coefficients depending on particle type can be defined in `[multiplier]` section, using `part` parameter. Owing to this improvement, several types of radiation doses can be directly calculated using pre-defined `[multiplier]` sections.

- `[t-deposit]` with `output=deposit` can be used in the case that source weight is not 1.

- A bug in `[t-deposit]` in the cases of `output=deposit`, `dresol`$\neq 0$, and `part`$\neq$`all` was fixed.

- A bug in the range calculation of charged nuclei in some tallies when `iMeVperu=1` was fixed.

From ver. 3.02, the following functions were implemented, and some bugs were fixed. (2017/12/01)

- New parameters `nudtvar` and `udtvar`(i) were introduced for `[t-userdefined]`. After variables `udtvar`(i), ($i = 1, \cdots,$ `nudtvar`) was set in an input file, these can be used in `subroutine usrtally`. There is no upper limit of the number of `udtvar`(i) unlike `udtpara` before ver. 3.01. `udtpara` can be also set after this version.

- New options were implemented for the `unit` of `[t-let]` and `[t-sed]`.

- Position of the source generation was adjusted to just on the spherical surface for `s-type=9` with `dir=iso`. This revision influences only when some materials are placed outside the source sphere.

- A bug related to `[counter]` section was revised. Before this revision, occurrence of atomic interactions were ignored in calculating counter when `negs` was set to `-1` (only photon transport mode).
- A bug in the calculation of the restricted stopping power when δ-rays are generated by `[delta ray]` was fixed.
- The bug in the calculation of angular straggling using `nspred = 2` was fixed. This bug was introduced in PHITS2.96, and calculation results for charged particle beam using the versions between 2.96 and 3.01 might be strange.

1.2 Development members

Koji Niita,
Research Organization for Information Science & Technology (RIST).

Tatsuhiko Sato, Yosuke Iwamoto, Shintaro Hashimoto, Tatsuhiko Ogawa, Takuya Furuta, Shinichiro Abe, Takeshi Kai, Yusuke Matsuya[27], Norihiro Matsuda, Yuho Hirata, Takuya Sekikawa, Lan Yao, Pi-En Tsai[28], and Hunter Ratliff[29]
Japan Atomic Energy Agency (JAEA).

Hiroshi Iwase, Yasuhito Sakaki, and Kenta Sugihara
High Energy Accelerator Research Organization (KEK).

Nobuhiro Shigyo,
Kyushu University.

Lembit Sihver,
Vienna University of Technology, Austria.

The following members also contributed to the development of PHITS.

Hiroshi Takada, Shin-ichro Meigo, Makoto Teshigawara, Fujio Maekawa, Masahide Harada, Yujiro Ikeda, Yukio Sakamoto, Hiroshi Nakashima, Tokio Fukahori, Keisuke Okumura, Tetsuya Kai, and Shusaku Noda,
Japan Atomic Energy Agency (JAEA).

Satoshi Chiba,
Tokyo Institute of Technology (TITech).

Takashi Nakamura,
Tohoku University.

Davide Mancusi,
Chalmers University, Sweden.

1.3 Reference of PHITS

Please refer the following document in context of using any version of PHITS.

- T. Sato, Y. Iwamoto, S. Hashimoto, T. Ogawa, T. Furuta, S. Abe, T. Kai, Y. Matsuya, N. Matsuda, Y. Hirata, T. Sekikawa, L. Yao, P.E. Tsai, R.N. Hunter, H. Iwase, Y. Sakaki, K. Sugihara, N. Shigyo, L. Sihver and K. Niita, Recent improvements of the Particle and Heavy Ion Transport code System - PHITS version 3.33, J. Nucl. Sci. Technol. 61, 127-135 (2024).

[27] Hokkaido University
[28] Current affiliation: Zap Energy, USA
[29] Current affiliation: Western Norway University of Applied Sciences (HVL), Norway

This is an open access article can be downloaded from https://www.tandfonline.com/doi/full/10.1080/00223131.2023.2275736

Other articles that describe the features of PHITS are:

- H. Iwase, K. Niita, T.Nakamura, Development of general purpose particle and heavy ion transport Monte Carlo code. J Nucl Sci Technol. 39, 1142-1151 (2002).
- K. Niita, T. Sato, H. Iwase, H. Nose, H. Nakashima and L. Sihver, Particle and Heavy Ion Transport Code System; PHITS, Radiat. Meas. 41, 1080-1090 (2006).
- L. Sihver, D. Mancusi, T. Sato, K. Niita, H. Iwase, Y. Iwamoto, N. Matsuda, H. Nakashima, Y. Sakamoto, Recent developments and benchmarking of the PHITS code, Adv. Space Res. 40, 1320-1331 (2007).
- L. Sihver, T. Sato, K. Gustafsson, D. Mancusi, H. Iwase, K. Niita, H. Nakashima, Y. Sakamoto, Y. Iwamoto and N. Matsuda, An update about recent developments of the PHITS code, Adv. Space Res. 45, 892-899 (2010).
- K. Niita, N. Matsuda, Y. Iwamoto, H. Iwase, T. Sato, H. Nakashima, Y. Sakamoto and L. Sihver, PHITS: Particle and Heavy Ion Transport code System, Version 2.23, JAEA-Data/Code 2010-022 (2010).
- K. Niita, H. Iwase, T. Sato, Y. Iwamoto, N. Matsuda, Y. Sakamoto, H. Nakashima, D. Mancusi and L. Sihver, Recent developments of the PHITS code, Prog. Nucl. Sci. Technol. 1, 1-6 (2011).
- T. Sato, K. Niita, N. Matsuda, S. Hashimoto, Y. Iwamoto, S. Noda, T. Ogawa, H. Iwase, H. Nakashima, T. Fukahori, K. Okumura, T. Kai, S. Chiba, T. Furuta and L. Sihver, Particle and Heavy Ion Transport Code System PHITS, Version 2.52, J. Nucl. Sci. Technol. 50:9, 913-923 (2013).
- T. Sato, Y. Iwamoto, S. Hashimoto, T. Ogawa, T. Furuta, S. Abe, T. Kai, P.-E. Tsai, N. Matsuda, H. Iwase, H. Shigyo, L. Sihver, and K. Niita, Features of Particle and Heavy Ion Transport code System (PHITS) version 3.02, J. Nucl. Sci. Technol. 55, 684-690 (2018).

2 Installation and execution of PHITS

PHITS is a fortran-based code; it can be executed on Windows (7 or later), Mac (OS X v10.6 or later), Linux, and Unix operating systems. The installers are prepared for each environment. The execution environment can be easily setup through the use of installer. Note that an administrator account is required to use PHITS on Windows and Mac operating systems.

The recommended system requirements for PHITS are 2 GB of RAM and 6 GB (with at least 4 GB required) of available space on the hard disk.

The installation and execution methods for Windows, Mac, and Linux OS are respectively described in Sections 2.1, 2.2, 2.3. In addition, Section 3 describes how to resume PHITS calculations, and how to terminate the PHITS code during calculations.

2.1 Installation and execution on Windows OS

2.1.1 Installation on Windows OS

(1) If you have an older version of PHITS, please make a backup of your edited files and install the new version in the same folder (overwrite installation is recommended). During the installation, only the address file ("xsdir.jnd") can be selected to be overwritten or not, and if overwritten, a backup of the old file is automatically created.

(2) Double-click "phits_win_setup.exe" in the win folder of the PHITS package.

(3) Specify the installation folder ("c:/phits" is recommended).

(4) Select the required software. We encourage you to select `New Installation` if you are installing PHITS for the first time on your PC, or `PHITS Upgrade` if you are updating the software (see Sec. 2.1.3).

(5) Installation is finished.

(6) To check whether PHITS has been installed properly, right click "\phits\lecture\basic\lec01\lec01.inp" and select "send to" → "PHITS."

(7) Check whether "xz_track_all.eps" is created or not.

(Note #1) The names of the installation files and folders and must contain no space characters; otherwise installation and execution will fail.

The installer "phits_win_setup.exe" performs the following processes:

(1) If an editor is selected for installation, install that editor and associate it with "*.inp" and "*.out" files. Also, modify the configuration file so that PHITS and PHIG-3D can be executed directly from that editor.

(2) If Ghostscript is selected, run the Ghostscript installer.

(3) If SumatraPDF is selected, run the SumatraPDF installer.

(4) Install Visual C++ Redistributable Package required to run PHIG-3D.

(5) Extract phits.7z to the specified installation folder.

(6) Pass `PATH` to the folder "phits/bin" which contains the PHITS executable files.

(7) Add the environment variable `PHITSPATH` (this eliminates the need to specify `file(1)` in the input file).

(8) Create shortcuts of four batch files - "phits.bat," "angel.bat," and "phig3d.bat" in the "phits/bin" folder, and `dchain.bat` in the "phits/dchain-sp/bin" folder - to the "sendto" folder.

(9) Copy "app/7za.exe" to "phits/bin" folder for preparing the installation of JENDL-5.

2.1.2 Execution on Windows OS

The PHITS code can be executed either via File Explorer (previously known as Windows Explorer) using "sendto → PHITS" command, or via Command Prompt.

2.1.2.1 Execute using "send to → PHITS"

Right click the PHITS input file in a File Explorer, select "send to → PHITS" . As the shortcut for "phits.bat" in the "\phits\bin\" folder has been created during installation, the PHITS code will be initiated and start simulations.

2.1.2.2 Execute via Command Prompt

PHITS can be executed by command lines in Command Prompt (also known as cmd.exe or cmd), the command-line interpreter on Windows operating systems as shown in Figure 2.1.

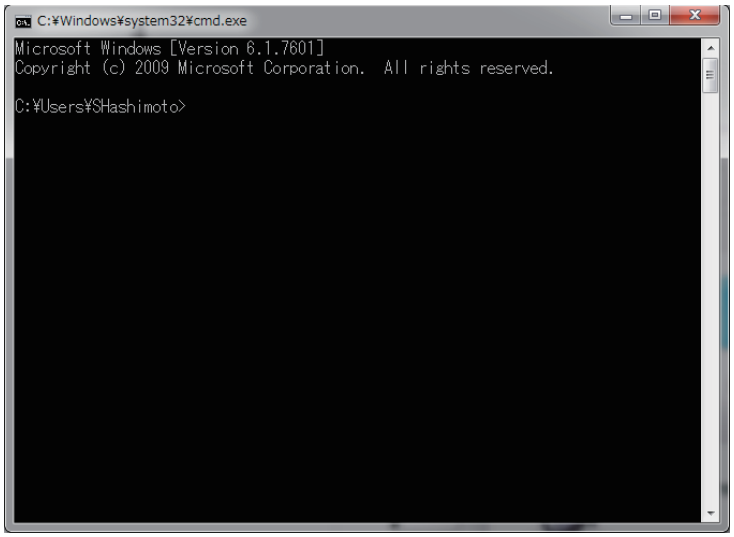


Figure 2.1: Example of how to execute PHITS via Command Prompt

To execute PHITS, use the "cd" command to go to the directory of where the input is, and then type the following commands in the terminal:

```
phits.bat your_input.inp
```

where "your_input.inp" is your input file name (e.g., "lec01.inp"). As command histories can be retrieved by pressing the ↑ key in the terminal, it is convenient to use the same input file name when you want to repeatedly execute PHITS.

2.1.3 Recommended software

There is no software that needs to be pre-installed before using PHITS. Nevertheless, it is recommended to install a text editor that can display line numbers, as the line number is specified when an error occurs in the PHITS input file. Furthermore, installation of Ghostscript and SumatraPDF. which are written in the EPS format, is necessary to display image files created by PHITS.

Free text editors for Windows:

- Notepad++ (`https://notepad-plus-plus.org/`).

For details of the installation of Ghostscript and SumatraPDF, see the following web pages:

- Ghostscript (`https://www.ghostscript.com/releases/gsdnld.html`)

- SumatraPDF（`https://www.sumatrapdfreader.org/free-pdf-reader`）

Please install this software after installation of Ghostscript.

2.2 Installation and Execution on Mac OS

2.2.1 Installation on Mac OS

(1) Right-click “phits_installer” in the mac folder of the PHITS package and seslect “Open” in the shortcut menu.

(2) Select `Automatic` in installation mode (See Fig. 2.2). In the failure case, try `Manual` described later. please see Sec. 2.2.3.

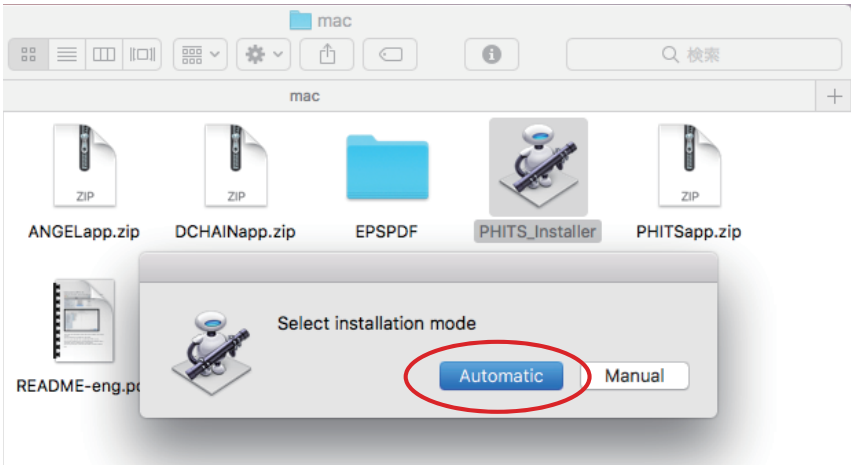


Figure 2.2: Selection of installation mode.

(3) Specify the folder to install PHITS in. The folder with the account name (ex: iwamoto) is recommended to be selected (See Fig. 2.3). After installation of PHITS, please reboot your system (or logout and re-login).

(4) The “phits” folder is then automatically created. All the contents of PHITS, such as execution files, source files, and nuclear data libraries, are installed in the “phits” folder. It may take several minutes.

(Note #1) If you move the “/phits/” folder to another place after installation, PHITS does not work properly.

(Note #2) If “/phits/” folder already exists in the specified folder, the old “/phits/” folder is automatically renamed to “phits[date].[time]” during the installation.

(Note #3) If the name of PHITS installation folder contains a space character, installation of PHITS may be failed.

2.2.2 Execution on Mac OS

PHITS can be executed by dragging and dropping or through the use of Terminal.

2.2.2.1 Execute by drag and drop

Drag the icon of the PHITS input file and drop it on the blue “PHITS” icon on the Dock (See Fig. 2.4). To test whether the installation is successful, click the icon Finder on the Dock and open the folder of “phits/lecture/basic/lec01.” Then, drag “lec01.inp” and drop it on the PHITS icon. A new terminal window appears, and the calculation condition is output on it. The calculated results will be shown in the same folder as the input file. As histories of commands are shown when you press the ↑ key in your Terminal, it helps to execute PHITS many times with same input file.

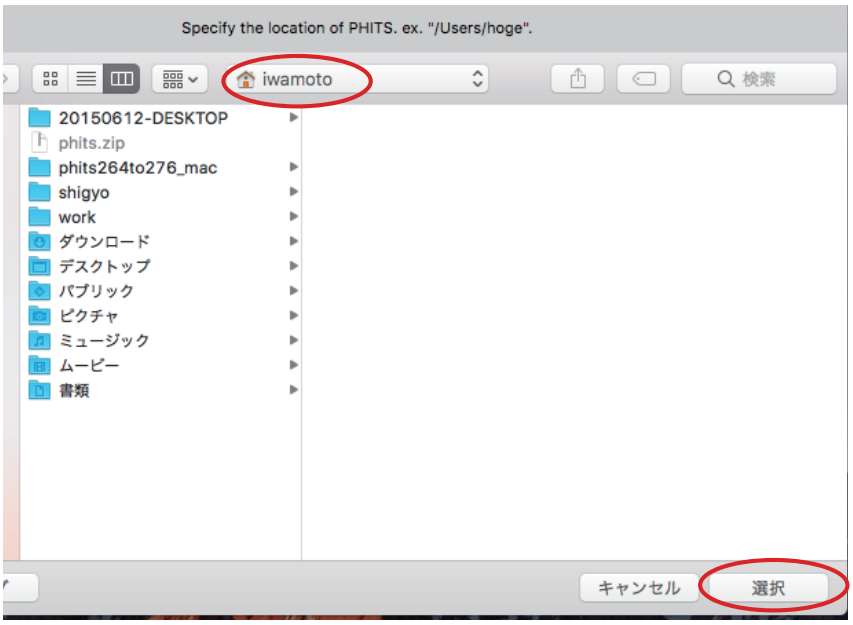


Figure 2.3: Screenshot of what folder is recommended for PHITS installation.

Figure 2.4: Execution by drag and drop.

To execute ANGEL and DCHAIN, drag the icon of their input file generated by the PHITS tally, and drop it on the red "ANGEL" or the green "DCHAIN" icon on the Dock.

To execute PHIG-3D, drag the icon of the PHITS input file on the robot icon "PHIG3D_launch" icon on the Dock.

To execute PHITS-Pad, drag the icon of the PHITS input file on the "PHITSPAD_launch" icon on the Dock.

2.2.2.2 Execute via Terminal

After the PHITS installation using "PHITS_Installer.command," you can use PHITS via Terminal with command lines. To execute in Terminal, select the following items (See Fig. 2.5).

Finder → Applications → Utilities → Terminal

To execute PHITS, go to the folder where your input file is located by a "cd" command, and type the following commands in the terminal.

```
phits.sh input.inp
```

where "input.inp" shows the input file name (e.g. "lec01.inp"). As histories of commands appears when you press the ↑ key in your Terminal, it helps to execute PHITS many times with same input file.

You can also use ANGEL, DCHAIN-SP, PHIG-3D, and PHITS-Pad via Terminal. To execute ANGLE, type the following command in the Terminal.

```
angel.sh angel.out
```

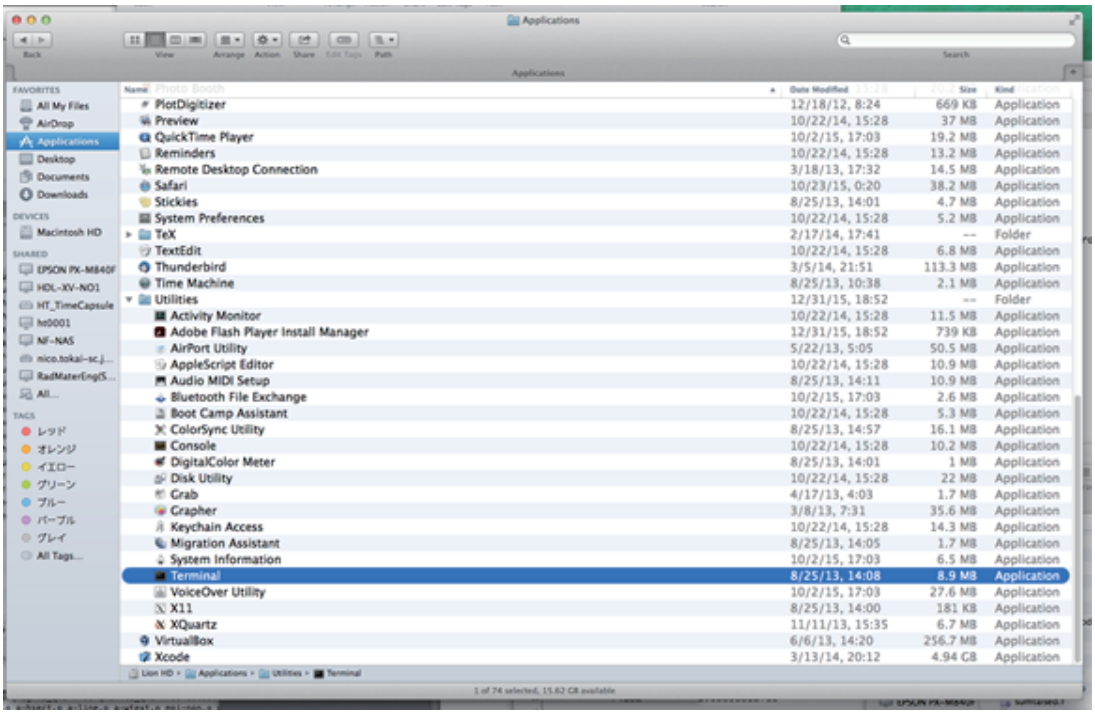


Figure 2.5: How to launch Terminal: Finder → Applications → Utilities → Terminal.

where “angel.out” is the input file name (tally output of PHITS, e.g. “track_xz.out”). To execute DCHAIN-SP, type the following command in the Terminal.

```
dchain.sh dchain.out
```

where “dchain.out” is the name of DCHAIN-SP input file (file name is designated in the `[t-dchain]` section of the PHITS input.) To execute PHIG-3D, type the following command in the Terminal

```
phid3d.sh input.inp
```

where “input.inp” shows the input file name (e.g. “lec01.inp”). To execute PHITS-Pad, type the following command in the Terminal

```
phitspad.sh input.inp
```

where “input.inp” shows the input file name (e.g. “lec01.inp”).

2.2.3 Installation procedure using `Manual` mode in “PHITS_Installer”

If you have trouble with `Automatic` mode, please try the followings.

(1) Create a directory named “phits” in your home directory (ex: “/Users/iwamoto”).

(2) Copy “phits.7z” in the “PHITSXXXA” folder created by extracting the downloaded zip file to the “phits” directory (ex: “/Users/iwamoto/phits”).

(3) Extract “phits.7z” with double-click.

(4) Copy “ANGEL.app.zip,” “DCHAIN.app.zip,” “EPSPDF.app.zip,” “PHITS.app.zip,” “PHITSPAD.app.zip” in the “PHITSXXXA/mac” folder to the “phits/bin” directory and extract them by double-click.

(5) Double-click “PHITS_Installer.command” and select `Manual` in the shortcut menu.

(6) Specify the location of PHITS2.52 binary, “phits252_mac.exe.”

(7) Execute PHITS by dragging and dropping “lec01.inp” from the folder “/phits/lecture/basic/lec01” to the blue PHITS icon on the Dock.

(Note 1) If you cannot install PHITS even using the manual mode, see Sec. 2.2.4.

2.2.4 Installation procedure without using “PHITS_Installer.command”

In this section, you will execute PHITS via Terminal. When you execute PHITS via Terminal at the first time, it is necessary to set a PATH to the folder with the PHITS execution file. To set a PATH to a folder, type the following commands in your Terminal.

```
echo 'export PHITSPATH=/PATH-TO-PHITS/phits/bin' >> .zshenv
echo 'export PATH=$PHITSPATH/bin:$PHITSPATH/dchain-sp/bin:$PATH' >> .zshenv
source .zshenv
```

where “/PATH-TO-PHITS” should be changed to the name of the folder where you installed PHITS (e.g. “/Users/iwamoto”).

If you do not know the name of your installed folder, you can check it by typing the following in your Terminal.

```
find $HOME -name phitsXXX_mac.exe
```

where XXX is the version of PHITS (e.g. 334. XXX is in the “PHITSXXXA” folder created by extracting the downloaded zip file). “PATH-TO-PHITS” is the result displayed on the terminal without “/phits/bin.”

Note that it is unnecessary to set a PATH after executing PHITS at the first time.

Next, in “/phits/bin/phits.sh,” “phitsYYY_mac.exe” at line 11, and then “phitsYYY_mac_openmp.exe” at line 14, match YYY with the version you are trying to install. Refer to the previous section for the execution method. The flow of execution is shown in Fig. 2.6.

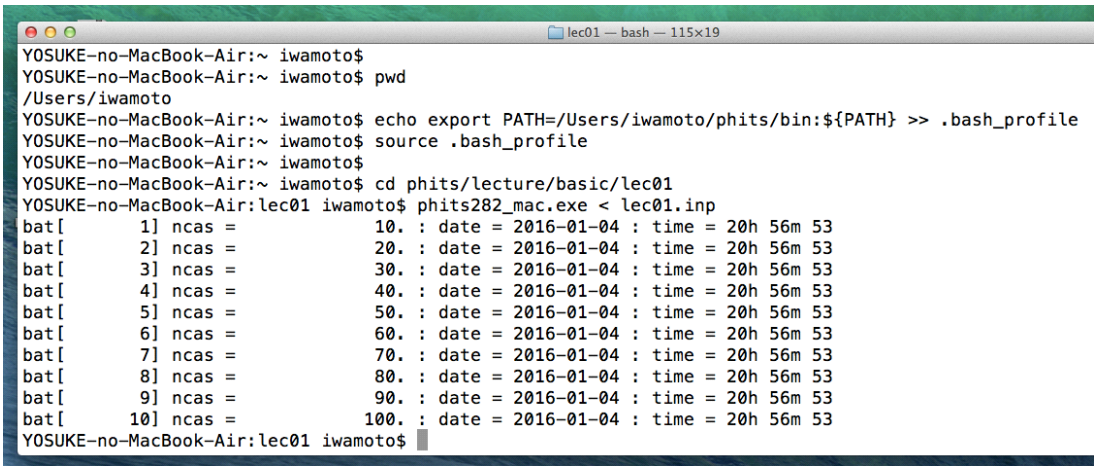


Figure 2.6: Example of PHITS execution via Terminal with the initial setting.

2.3 Installation and Execution on Linux OS

2.3.1 Installation on Linux OS

Copy “phits.zip” from the PHITS installation package to a directory where you would like to install. Copy also the installer for Linux “PHITS-Installer_lin.sh” from the “linux” folder in the PHITS installation package to the same folder. Open a terminal and move to the installing folder. Then type the following command to add executable property to the installer:

```
chmod +x PHITS-Installer_lin.sh
```

Then execute the installer by the following command:

```
./PHITS-Installer_lin.sh
```

Installation will proceed in unzipping the ZIP file, modifying the installation PATH in files, and setting the execution PATH. Input of password will be required during the unzip. PATH setting will be done in ".profile" with the assumption of the use of Bash. If you are using other shells, please change the setting accordingly. Installation is successfully finished if you do not see any error message.

Type the following command to activate the PATH setting:

```
source ~/.profile
```

This command is not necessary anymore from the next login, since ".profile" will be automatically read when logging in.

2.3.2 Execution on Linux OS

In Linux environment, PHITS is executed via terminal. Open a terminal and move to the directory which contains the input file by the "cd" command. Then PHITS can be executed in the following fashion.

```
phits.sh your_input.inp
```

Here "your_input.inp" is the PHITS input file name (e.g., lec01.inp). As command histories can be retrieved by pressing the ↑ key in the terminal, it is convenient to use the same input file name when you want to repeatedly execute PHITS.

The execution of ANGEL and DCHAN-SP can be done in a similar fashion. ANGEL can be executed by

```
angel.sh angel.inp
```

where "angel.inp" is the input file name (tally output of PHITS. e.g., "track_xz.out").

To execute DCHAIN, type the following command in the terminal:

```
dchain.sh dchain.out
```

where "dchain.out" is the name of the DCHAIN input file (the file name is designated in the `[t-dchain]` section of the PHITS input).

Note that if errors, such as no permission, occur when executing PHITS, it might be because the execution permission has not been added to the executable file. For such a situation, go to "phits/bin/" or where the executable file is located in the terminal, and type the following command for authorization:

```
chmod +x XXX
```

Here "XXX" is the name of the executable file (e.g., phits309.exe).

2.3.2.1 Execute by specifying an executable file without using a shell

The PHITS code can be executed on Linux without using a shell by the following command:

```
phitsXXX.exe < your_input.inp
```

where "phitsXXX.exe" is the PHITS executable file and XXX is the version number, and "your_input.inp" is the input file for PHITS calculation.

(Note #1) If additional files are used with the `infl:` parameter when PHITS is executed with this method. In this case, the following text should be written in the first line of the main input file:

```
file = phits.inp
```

See Sec. 4.3 for further discussion of `infl:`.

(Note #2) To perform a PHITS calculation using distributed-memory parallel computing, the name of the input file must be specified in the first line of the "phits.in" file, which locates in the same folder containing the executable file, by the following manner:

```
file = input.inp
```

The name of "phits.in" is fixed and cannot be changed arbitrarily. See the example in Sec. 11.1.3 for more detail.

2.4 How to change the executable file

The PHITS executables are all stored in the "phits/bin" folder for Windows, Mac, and Linux, and include not only a single mode, but also parallel computing and debugging. These executables can be used differently by putting the command before the first section of the input file.

To execute PHITS via parallel computing using MPI or OpenMP. The shared-memory parallel computing using OpenMP can be executed by adding `$OMP=`*N* (*N* is the number of CPU cores to be used) before the first section in the PHITS input file. If $N = 0$, all cores in the computer are used. If $N = 1$, the parallel computing is not used. From version 2.73, the executable file of the OpenMP version coming along with the PHITS package is available only for 64-bit Windows systems. The MPI version of PHITS can be also executed by added `$MPI=`*M* in the same manner as OpenMP, but the installation of a MPI protocol (MPICH2 for Windows, OpenMPI for Linux) is required. Please see Sec. 11.1.1 in more detail.

The executable files of PHITS in the debug mode, which checks for memory violations and uninitialized variables, can be set when `$DBG=1` is written before the 1st section of the input file. When "access violation" or "Segmentation fault" are occurred during the execution of PHITS, or when you find infinite loops and abnormal termination of calculations, please append `$DBG=1` to the first line of the input file that If you get any error messages, please send the messages to PHITS office, together with the information on your machine environment.

3 Restart and terminate PHITS calculations

3.1 Restart PHITS calculations

If tally results of PHITS calculations are not statistically significant enough, PHITS is able to read the previous tally results and resume the calculations (restart calculation mode). In the [parameters] section, restart calculation mode can be activated by setting istdev < 0, and the statistics of the restart calculation will be counted together with the previous tally results. It is allowed to add new tally sections for the input file with the restart calculation, even if there is no past tally result. Please refer to Table 5.2 for the setting of istdev.

The procedure of restart calculation is described as follows:

(1) Check whether or not the file specified by resfile in each tally section exists. The default file name of the resfile is that given by file parameter.

(2) If there is no resfile for a tally, it is regarded to be new. If no resfile can be found for any tally, a new calculation is begun with istdev=|istdev|.

(3) If resfile exists, PHITS reads information from the file on the standard deviation mode istdev, total weight resc2, total history number resc3, history number per batch maxcas, the next random seed rijklst, and the results and relative errors of the past calculation.

(4) The tally parameters given in the current and past PHITS input files are then checked for consistency. If they are not consistent with each other, PHITS stops the calculation and outputs an error message. It should be noted that not all tally parameters are checked for consistency in this process.

(5) (In batch standard deviation mode only) The consistency of istdev and maxcas among the resfiles is checked. If they are consistent, the restart calculation is performed using these values. If an inconsistency is found, the calculation is stopped.

(6) The initial random seed is changed to the value of rijklst obtained from the first resfiles. If rijklst differs by resfile, a warning message is output.

(7) Upon finishing the restart calculation, the tally results are output to the file specified by file=. If resfile is not specified, the past tally results are overwritten.

Important notice:

I. All past tally results should be calculated in the same standard deviation mode: i.e., istdev should be the same in all resfile.

II. The maxcas written in the input file is not used in the restart calculation in batch standard deviation mode.

III. Except for those given in the tally sections, the consistencies of input parameters are not checked by the PHITS program. If they are inconsistent with the previous setting, the tally results might be incorrect even if no error messages show up.

3.2 Terminate the PHITS code

Once PHITS is executed, it creates a "batch.out" [30] file following every batch that contains elapse information. In parallel calculation, batch.out also contains each processor element (PE) status. The occurrence of a PE abort can be determined by examining batch.out.

The first line of batch.out is written as follows:

```
10 <--- number of remaining batches
```

Here, the value on the left is the number of remaining batches at the moment. The remaining batches can be reduced by editing the value and saving batch.out. For example, when the value is changed to "3" the PHITS execution will be terminated after the calculation of three batches is finished. Changing the value to "0" terminates the code

[30] Until ver. 2.85, the name of this file had been named "batch.now."

execution right after the calculation of that batch is completed. This function can be useful in the way explained below.

The parameter `itall` can be used together with "batch.out". If `itall=0(default)` or `itall=1` is set in the `[parameters]` section, PHITS will update the calculation results (tally output) at the end of each batch. With `itall=1`, the image (*.eps) files will also be updated (See Sec. 6.7.18). In parallel computing mode, the results are updated every batch $\times$ (PE -1). Using these functions enables terminating PHITS calculations at any time with the latest results updated.

From ver. 2.86, the file name of batch.out can be modified by setting `file(22)` in the `[parameters]` section. Changing the name of each input file enables multiple executions of PHITS in a common directory. Note that this function is not available if `[t-dchain]` is set in the input files, because the same file name, "n.flux," is used.

The value of `rijk` written in the batch.out file is the initial random number of the current batch. Thus, for example, in cases of unsuccessful termination of PHITS, it is possible to reproduce the calculation of the specified batch using the value of `rijk`.

4 General format of input and output files

The PHITS input file is made in a text format that can be edited with a text editor. The input file consists of several sections beginning with [******], and the required information is read in sections according to the purpose and option. The sections can be in any order. Note that up to four spaces at the beginning of [******] will be ignored, but any more than that will not be recognized as the beginning of a section. The input format depends on each section, but basically takes the two formats shown below. One is a method where keywords (parameters) unique to each section are set as numeric values or strings, as follows:

```
keyword（parameter） = value or character
```

In this case, spaces are ignored. The other is to set multiple numbers with a space as a delimiter, the format is as follows:

```
parameter1    parameter2    parameter3  ...
```

The output file is basically in text format (".out"), but by setting an option, an eps file (".eps") as an image of calculation results can also be output.

The units of each physical quantity in PHITS are basically: length in cm, time in ns, and energy is MeV (or MeV/n for quantity per nucleon). The coordinate system is right-handed.

4.1 Sections

Table 4.1 and 4.2 shows the sections used in PHITS.

Table 4.1: Sections (1)

name	description
[title]	Title
[parameters]	Various types of parameters
[source]	Source definition
[material]	Material definition
[surface]	Surface definition
[cell]	Cell definition
[transform]	Definition of the coordinate transform
[temperature]	Cell temperature definition
[mat time change]	Definition of time-dependent materials
[magnetic field]	Magnetic field definition
[electro magnetic field]	Electro-magnetic field definition
[delta ray]	Definition of δ-ray production
[track structure]	Definition for using the track-structure simulation mode.
[super mirror]	Definition of super mirror for low-energy neutrons
[elastic option]	Definition of elastic scattering options for low-energy neutrons
[data max]	Definition of maximum energies (dmax) of each nucleus for using data libraries
[frag data]	Definition of user defined cross sections
[importance]	Definition of region's importances
[weight window]	Definition of region's weight windows
[ww bias]	Definition of values to bias in [weight window]
[forced collisions]	Forced collision definition
[repeated collisions]	Repeated collision definition
[volume]	Region volume definition
[multiplier]	Multiplier definition
[mat name color]	Material name and color definition for graphical plot
[reg name]	Region name definition for graphical plot
[counter]	Counter definition
[timer]	Timer and clock definition
[user defined interaction]	Control of user-defined interactions
[user defined particle]	Control of user-defined particles

Table 4.2: Sections (2)

name	description
[t-track]	Particle fluence in a given region.
[t-cross]	Particle fluence/current across a given surface.
[t-point]	Particle fluence at a given point.
[t-deposit]	Deposit energy in a given region.
[t-deposit2]	Deposit energies in two given regions.
[t-heat]	Heat generation in a given region. (Not recommended[31])
[t-yield]	Residual nuclei yields in a given region.
[t-product]	Reaction products in a given region.
[t-dpa]	Displacement Per Atom (DPA) in a given region.
[t-let]	LET distribution in a given region.
[t-sed]	Microdosimetric quantity distribution in a given region.
[t-time]	Time information of particle in a given region.
[t-interact] (former name [t-star])	Number of interactions occurred in a given region.
[t-dchain]	Residual nuclide yields (for the use of DCHAIN).
[t-wwg]	Output parameters for [weight window].
[t-wwbg]	Output parameters for [ww bias].
[t-volume]	Automatic calculation of the volume for a given region.
[t-userdefined]	Any quantities defined by users.
[t-gshow]	2D geometry visualization.
[t-rshow]	2D geometry visualization with physical quantities.
[t-3dshow]	3D geometry visualization.
[T-4Dtrack]	Particle track files for displaying particle tracks in PHIG-3D.
[end]	End of input file.

Note that PHITS does not read any input information written below the [end] section.

4.2 Reading control

Uppercase, lowercase, blank(␣)

Discrimination between lowercase and uppercase characters is not performed in PHITS input except in file names. Blanks at the beginning and the end of a line are ignored except for the declaration of the [Section Name] as described above.

Tab

A tab is replaced by eight blanks.

Line connecting

The maximum number of characters that can be written in a line is 200. If '\' is added at the end of a line, the next line is considered to be a continued line. Multiple lines can be used to write input data using this method, but '\' is not necessary in the [surface] and [cell] sections, in which lines are automatically connected without the use of an additional symbol. Note that more than four blanks are required at the beginning of a continued line.

Line dividing

Several short lines can be displayed in one line, and divided by ';' as follows:

```
idbg = 0;  ibod = 1;  naz = 0
```

However, this function is not available when the format is defined in the mesh description.

[31] Until ver. 3.04, the [t-heat] tally had been used to calculate deposit energy using the kerma approximation, because the [t-deposit] tally did not have the option.

Comment marks

In PHITS input file, the following characters can be used as comment remarks, and characters written after them are ignored.

- `$` is regarded as a comment remark anywhere in PHITS input file.
- `#` and `!` are regarded as comment remarks except when they are used in `[surface]` and `[cell]`.
- c␣ (put a space at ␣) is regarded as a comment remark only when it is used within the first 5th column in each line. Note that it cannot be used as a comment remark in the `[material]` section by the default setting. To use $ as a comment remark in the `[material]` section, it is necessary to set `icommat=1` in `[parameters]` section.

Note that % could be also used as a comment remark before version 3.30, but since % is now used for defining the character variables, it cannot be used as a comment remark anymore. If you need to use `%` as a comment remark, write `$RWT=0` before the first section begins.

Blank lines

Blank lines and lines beginning with a comment mark are skipped.

Section reading skip

Adding `off` after a section name (e.g., `[Section Name] off`) disables all the definitions in that given section (i.e., the section is being skipped). Note that `infl:` or `set:` written in the skipped section is also skipped.

Skipping lines within a section

To skip from any place in a section, put `qp:` at the line head. Lines from `qp:` to the end of the section are skipped.

Skip all

`q:` can be used as a terminator of an input file; this works the same way as using `[end]`.

4.3 Inserting files

Files can be placed in any location using:

`infl:` { *file.name* } [$n_1 - n_2$]

The name of the file to be inserted should be enclosed in curly brackets '{ },' and the number of lines from n_1 to n_2 of the file should be enclosed in square brackets '[ ].' If there is no '[ ],' PHITS includes all lines of the specified file.

The following style can be used to specify line numbers: [n_1-] and [-n_2]. These expressions specify the range from the n_1th line to the end, and from the initial line to the n_2th line, respectively. Inserted files can be nested more than once (up to eight times). After reading the end of the including file, the reading process returns to its parent file.

Care should be taken in using the command-line interpreter (Command prompt) to execute PHITS. If `infl:` is used, the following text should be written in the first line of the input file:

```
file = phits.inp
```

Here, "phits.inp" is the input file name.

4.4 User-defined variables

In the input file, where numbers should be entered, User-defined variables from `c1` to `c999` can be written instead of numbers. User-defined variables can be set as follows:

```
set:  c1[ 52.3 ] c2[ 2 * pi ] c3[ c1 * 1.e-8 ]
```

The set: definition can be written anywhere. Note that there should be no space between c*i* and [in the format c*i*[]. User-defined variables can be used as numerical values in input file, and the variables can be re-defined at any time, with the values retained until they are re-defined. In the third case of the above example (c3), another variable c1 is called in the definition; in this case, the value held by the variable c1 at that time is used. Therefore, even if c1 is re-defined following the definition of c3, the value of c3 is not changed. Note that, by default, pi is set to the value of π.

The number of c*i* can be output in file(6)(D=phits.out) by setting ncvalout in [paraemters]. The number is output as many times as specified by ncvalout. If you want to check the actual numbers used as user-defined constants, please set it.

The set: definition is ignored in sumtally subsection.

User-defined variables can be used in the user-defined programs such as "usrsors.f, usrtally.f, and usrdfn1.f" by setting as below:

```
parameter ( mfc = 100 )
common /rval1/ cval(mfc), aval(mfc)
```

For example, cval(1) corresponds to c1 defined in the PHITS input file. If a certain user-defined variable is specified by multiple times in the input file, the lastly defined value is effective in the user defined programs.

4.5 User-defined characters

After Version 3.31, it is possible to use any characters as a variable. You can define the character variable as follows:

```
set:  %part_name%[neutron proton alpha]
```

Parameter names must be defined with % between them. **Note that only character-variable names are case-sensitive in PHITS input file.** As with user-defined variables, they are effective from the next line after they are defined, and the same character variable can be re-set as many times as necessary. On the other hand, only one character variable can be defined in one line, in different from the user-defined variable. Note that variables and their values (strings) that can be defined are limited to 100 characters.

To use this feature, PHITS from version 3.31 onward outputs the input information after replacing character variables to a separate file (hereafter "rewrite file") and then reads the file. The rewrite file is named by adding "_rwt" before the extension of the original input file name. For example, if the original input file name is "phits.inp", the rewrite file name is "phits_rwt.inp" [32]. The rewrite file also contains the input information read from the included files using the infl command.

Whether the creation and deletion of the rewrite file or not can be changed by the special parameter $RWT. The $RWT must be defined before the first section begins, and the meaning of its value is as follows:

- $RWT=0: No rewrite file is created. In this case, character variables cannot be used. This option is effective when the input file is huge because computational time for initial setup can be reduced.
- $RWT=1(default): Rewrite file is temporarily created and deleted at the end of calculation.
- $RWT=2: Rewrite file is created and kept as it is after finishing the PHITS simulation.
- $RWT=3: Rewrite file is created and PHITS execution stops immediately afterwards. This option is available only for icntl=0, and will be useful when sharing input files using multiple infl commands with others.

[32] If you use the PHITS executable file directly via terminal or command prompt, without using a batch file or script file, the name of the rewrite file will be "phits_rwt.inp" without reference to that of the input file.

4.6 Using mathematical expressions

Mathematical expressions can be used in an input file following Fortran style guidelines. Available functions are shown in Table 4.3.

Table 4.3: Intrinsic Functions.

Intrinsic Function							
FLOAT	INT	ABS	EXP	LOG	LOG10	MAX	MIN
MOD	NINT	SIGN	SQRT	ACOS	ASIN	ATAN	ATAN2
COS	COSH	SIN	SINH	TAN	TANH		

As an example of the use of standard mathematical expressions:

```
parameter= c1 * 3.5 * sin( 55 * pi / 180 )
```

In the above example, because a single numerical value is expected after `parameter=`, blanks can be used in the expression. However, in some sections multiple numerical values cannot be accommodated; in such cases, an expression can be closed using '{ }', for example, { `c1 * 2 / pi` }.

4.7 Particle identification

The particle species that can be handled by PHITS are listed in Table 4.4. These particle species can be specified by their respective symbols or kf-codes. Particle species that are not assigned a symbol in Table 4.4 (itype=11) are specified only by their kf-codes. Table 4.5 lists the particles' decay-channels and their life-times.

Simulation of nuclei can be handled in PHITS with the use of physical models, such as JQMD. The isotope notation is in the format of the mass number followed by the chemical symbol of the element, e.g., 208Pb, 56Fe, etc.. When the mass number of an element is not specified (e.g., Pb and Fe), it refers to all naturally occurring isotopes of the respective element, and thus cannot be assigned as projectiles. Another way of describing nuclei is using the kf-code: `kf`= $Z * 1000000 + A$.

Table 4.4: List of transport particles.

ityp	symbol	kf-code	particle name
1	proton	2212	proton
2	neutron	2112	neutron
3	pion+	211	π^+
4	pion0	111	π^0
5	pion−	−211	π^-
6	muon+	−13	μ^+
7	muon−	13	μ^-
8	kaon+	321	K^+
9	kaon0	311	K^0
10	kaon−	−321	K^-
11	other	below	other particle
12	electron	11	e^-
13	positron	−11	e^+
14	photon[a]	22	γ
15	deuteron	1000002	deuteron
16	triton	1000003	triton
17	3he	2000003	^{3}He
18	alpha	2000004	α
19	nucleus	Z*1000000+A	nucleus

[a] In the previous version of PHITS, `photon` was coded as `gamma`; in the new version, this is changed to `photon`.

ityp	symbol	kf-code	particle name
11	–	+ − 12	$\nu_e\ \overline{\nu_e}$
11	–	+ − 14	$\nu_\mu\ \overline{\nu_\mu}$
11	–	−2212	$\overline{\mathrm{p}}$
11	–	−2112	$\overline{\mathrm{n}}$
11	–	−311	$\overline{\mathrm{K}^0}$
11	–	130	K^0_L
11	–	310	K^0_S
11	–	+ 221	η
11	–	331	η'
11	–	+ − 3122	$\Lambda^0\ \overline{\Lambda^0}$
11	–	+ − 3222	$\Sigma^+\ \overline{\Sigma^+}$
11	–	+ − 3212	$\Sigma^0\ \overline{\Sigma^0}$
11	–	+ − 3112	$\Sigma^-\ \overline{\Sigma^-}$
11	–	+ − 3322	$\Xi^0\ \overline{\Xi^0}$
11	–	+ − 3312	$\Xi^-\ \overline{\Xi^-}$
11	–	+ − 3334	$\Omega^-\ \overline{\Omega^-}$
11	–	+ - 16	$\nu_\tau\ \overline{\nu_\tau}$
11	–	+ - 15	$\tau^-\ \tau^+$
11	–	113	ρ^0
11	–	+ - 213	$\rho^+\ \rho^-$
11	–	223	ω
11	–	+ - 20213	$a_1^+\ a_1^-$
11	–	+ - 313	$K^{*0}\ \overline{K^{*0}}$
11	–	+ - 323	$K^{*+}\ K^{*-}$
11	–	333	ϕ
11	–	+ - 411	$D^+\ D^-$
11	–	+ - 421	$D^0\ \overline{D^0}$
11	–	+ - 431	$D_s^+\ D_s^-$
11	–	+ - 511	$B^0\ \overline{B^0}$
11	–	+ - 521	$B^+\ B^-$
11	–	+ - 531	$B_s^0\ \overline{B_s^0}$
11	–	+ - 541	$B_c^+\ B_c^-$

Table 4.5: Decay channels and life times. The branching ratio and decay mode can be changed by modifying and recompiling the source code "src/dklos.f". For other particles, please refer to "data/decay/decay_data_1.dat."

			branching ratio	life time (sec)
n	$\rightarrow$	$p + e^- + \overline{\nu}_e$	100%	8.867×10^2
π^0	$\rightarrow$	$\gamma + \gamma$	100%	0
π^+	$\rightarrow$	$\mu^+ + \nu_\mu$	100%	2.6029×10^{-8}
π^-	$\rightarrow$	$\mu^- + \overline{\nu}_\mu$	100%	2.6029×10^{-8}
μ^+	$\rightarrow$	$e^+ + \nu_e + \overline{\nu}_\mu$	100%	2.19703×10^{-6}
μ^-	$\rightarrow$	$e^- + \overline{\nu}_e + \nu_\mu$	100%	2.19703×10^{-6}
K_L^0	$\rightarrow$	$\pi^+ + e^- + \overline{\nu}_e$	20.34%	5.116×10^{-8}
	$\rightarrow$	$\pi^- + e^+ + \nu_e$	20.34%	
	$\rightarrow$	$\pi^+ + \mu^- + \overline{\nu}_\mu$	13.57%	
	$\rightarrow$	$\pi^- + \mu^+ + \nu_\mu$	13.57%	
	$\rightarrow$	$\pi^0 + \pi^0 + \pi^0$	19.59%	
	$\rightarrow$	$\pi^+ + \pi^- + \pi^0$	12.59%	
K_S^0	$\rightarrow$	$\pi^0 + \pi^0$	30.72%	8.954×10^{-11}
	$\rightarrow$	$\pi^+ + \pi^-$	69.28%	
K^+	$\rightarrow$	$\mu^+ + \nu_\mu$	63.56%	1.2371×10^{-8}
	$\rightarrow$	$\pi^0 + e^+ + \nu_e$	5.07%	
	$\rightarrow$	$\pi^0 + \mu^+ + \nu_\mu$	3.352%	
	$\rightarrow$	$\pi^+ + \pi^0$	20.67%	
	$\rightarrow$	$\pi^+ + \pi^0 + \pi^0$	1.760%	
	$\rightarrow$	$\pi^+ + \pi^+ + \pi^-$	5.583%	
K^-	$\rightarrow$	$\mu^- + \overline{\nu}_\mu$	63.56%	1.2371×10^{-8}
	$\rightarrow$	$\pi^0 + e^- + \overline{\nu}_e$	5.07%	
	$\rightarrow$	$\pi^0 + \mu^- + \overline{\nu}_\mu$	3.352%	
	$\rightarrow$	$\pi^- + \pi^0$	20.67%	
	$\rightarrow$	$\pi^- + \pi^0 + \pi^0$	1.760%	
	$\rightarrow$	$\pi^- + \pi^- + \pi^+$	5.583%	
η	$\rightarrow$	$\gamma + \gamma$	38.9%	0
	$\rightarrow$	$\pi^0 + \pi^0 + \pi^0$	31.9%	
	$\rightarrow$	$\pi^+ + \pi^- + \pi^0$	23.7%	
	$\rightarrow$	$\pi^+ + \pi^- + \gamma$	other	
η'	$\rightarrow$	$\pi^+ + \pi^- + \eta$	44.1%	0
	$\rightarrow$	$\pi^0 + \pi^0 + \eta$	20.5%	
	$\rightarrow$	$\pi^+ + \pi^- + \gamma$	30.1%	
	$\rightarrow$	$\gamma + \gamma$	other	
Λ	$\rightarrow$	$p + \pi^-$	64.1%	2.631×10^{-10}
	$\rightarrow$	$n + \pi^0$	other	
Σ^+	$\rightarrow$	$p + \pi^0$	51.57%	7.99×10^{-11}
	$\rightarrow$	$n + \pi^+$	other	
Σ^0	$\rightarrow$	$\Lambda + \gamma$	100%	0
Σ^-	$\rightarrow$	$n + \pi^-$	100%	1.479×10^{-10}
Ξ^0	$\rightarrow$	$\Lambda + \pi^0$	100%	2.90×10^{-10}
Ξ^-	$\rightarrow$	$\Lambda + \pi^-$	100%	1.639×10^{-10}
Ω^-	$\rightarrow$	$\Lambda + K^-$	67.8%	8.22×10^{-11}
	$\rightarrow$	$\Xi^0 + \pi^-$	23.6%	
	$\rightarrow$	$\Xi^- + \pi^0$	other	

4.8 Summary output file (phits.out)

Summary of PHITS calculation such as computational time and number of produced particles are given in a file with name specified by `file(6)` parameter (`D=phits.out`). The output information is distinguished by following 6 categories, and they can be selected by `infout` parameter in `[parameters]`.

Categories of output information

I Basic information (all information excluding the following items)

II Input echo

III Information on memory usage and batches

IV Information on transport particles

V Detailed information - variance reduction, number of scattered particles for each region, information for each material

VI Information on PHITS developers

The order of the output information is described as below.

Logo, version, developers (Category IV), title, starting time (Category I except for developers)
The version information is written in line 13. Please write this number when you send questions to PHITS forum or PHITS office.

Input echo (Category II)
Input echo, which repeats all parameters defined in the input file, is given between the lines >>> `Input Echo` >>> and `[END] of Input Echo` <<<. The meaning of each parameter is also output after comment remark `#`. When the parameters are specified using equations and/or user defined variables, `ci`, the actual values used in the PHITS simulation are output in the input echo. In the cases of the RI source (see Section 5.68) and cosmic-ray source (see Section 5.3.19.9) modes, the actual spectra of source particles are also provided here.

Memory usage (Category III)
Amounts of memory used for geometry (GG/CC memory), high and low-energy data libraries (high energy material / low energy material), tally (real tally memory), storing particle information (bank memory), and others (other memory) are written together with their total amount (used total memory). Note that these values are only for data stored in static memory, and thus, the values for data libraries and tally are always close to 0 because they are stored in the dynamic memory in the recent versions of PHITS.

Information on each source particle (only when `nwsors>0`)
The detailed information of each source such as type (`ityp`), cell ID (`reg`), material ID (`mat`), energy (`energy`), position (`x, y, z`) and unit vector of direction (`u, v, w`), particle weight (`wt`) is output up to the first `nwsors` sources.

Information on batch (Category III)
Total history number (`ncas`), the initial random seed (`bitseed`), and CPU time for each batch are output.

Information on calculation processes (number of analyz call vs ncol, Category I)
PHITS has a subroutine named "analyz", which is called every time when a process occurs, such as nuclear reaction, boundary crossing, source generation, and start and end of a batch. The process is tagged with `ncol` parameter (see Section 5.2.26 Dumpall in more detail). Here, the numbers of each process occurred in the PHITS simulation (i.e., number of analyz called with ncol) are output.

Information on variance reduction (Category V)
Detailed information on the variance reduction using `[importance], [weight window]` and `[forced collisions]` is output, such as the numbers of particle splitting and Russian roulette occurred in each cell.

Information on region-wise particle scattering (Category V)
Detailed information on the region-wise particle scattering is output, such as the numbers of high & low energy nuclear reactions and the gain and loss of particle fluences in each cell.

Information on medium-wise particle scattering (Category V)
The similar to above but medium-wise information is output.

Information on transport particle (Category IV)
List of transport particles and their produced, decayed, stopped, and leaked numbers are output.

Information on source (Category I)
Information on the summary of generated sources such as total history & batch numbers and their average and total weights is output.

Information on stopping power database (Category I)
When `ndedx` parameter is set to 1 or 3, ATIMA is used for calculating the stopping power of charged particles (see Table 5.10). In these cases, the stopping powers calculated by ATIMA are stored in memory with the maximum size specified by `mdbatima` parameter. Once the data size of the stopping power table reaches its maximum, the computational time may become slow. Here, the actual size of the stopping power table (`dedx`) and its maximum (`mdbatima`) are output. If these numbers are equal to each other, it is better to increase `mdbatima` in `[parameters]`.

Information on geometry error (Category I)
Information on geometry errors detected in the PHITS simulation is output. `Lost particle` indicates a particle entering to undefined region. When the number of lost particles exceed `nlost` parameter defined in `[parameters]`, PHITS calculation is terminated. In that case, please check your geometry and source definitions. If you cannot find any error in your geometry and source definitions, and the chance to cause the lost particles is quite small (approximately less than 1 per 10,000 history), you can avoid the termination of PHITS simulation by increasing `nlost` parameter. `Geometry recovering` indicates the situation that PHITS once fails to identify the cell ID where transport particle currently belonging to, but recovers the problem by slightly changing its coordinate. The most probable reason for causing geometry recovering is the loss of significant digit so that it is not necessary to seriously care about this problem even when the number of geometry recovering is large. On the other hand, `unrecovered error` indicates the similar situation but PHTIS fails to recover the problem and gives up to trace the particle trajectory anymore. Therefore, it is better to check your input file when the number of unrecovered errors is large (approximately more than 1 per 10,000 history).

Information on random seed (Category I)
Initial random seed used in this calculation and next random seed to be used in the restart calculation are output.

Information on CPU time and number of events (Category I)
Information on CPU time and number of each process occurred in the PHITS simulation is output. Note that output information does not include computational time except for the total CPU time when `icput=0` (default). When `infout=8`, the numbers of nuclear reaction models called (count) and succeeded (real), and their ratio in percent are additionally output. The meaning of each items are explained as follows:

- total cpu time: Total computational time
- transport (only when `icput>0`): Computational time for transporting particles
- set data (only when `icput>0`): Computational time for reading input file
- analysis: Number of all processes occurred and their computational time
- nevap: Number of evaporation models used and their computational time
- dexgam: Number of nuclear de-excitation models used and their computational time
- other: Computational time for others
- nreac: Number of nuclear reaction models used and their computational time

- dklos: Number of particle decay models used and their computational time
- hydro: Number of nucleon-nucleon scattering models used and their computational time
- n-data: Number of neutron nuclear data libraries used and their computational time
- h-data: Number of proton, deuteron, and alpha nuclear data libraries used and their computational time
- p-data: Number of photon atomic data libraries used and their computational time (only for `negs=-1`)
- e-data: Number of electron and positron atomic data libraries used and their computational time (only for `negs=-1`)
- p-egs5: Number of EGS5 photon interaction models used and their computational time
- e-egs5: Number of EGS5 electron and positron interaction models used and their computational time
- e-tsmode: Number of electron track-structure mode used and their computational time
- ion-tsmode: Number of proton and ion track-structure mode used and their computational time
- photonucl: Number of photonuclear reaction models used and their computational time
- photonucl lib: Number of photonuclear data libraries used and their computational time
- mu-reac: Number of muon nuclear reaction model used and their computational time
- mu-brem: Number of muon bremsstrahlung model used and their computational time
- mu-pprd: Number of muon pair-production model used and their computational time
- mu-cap: Number of muon capture model used and their computational time
- elast: Number of elastic scattering model used and their computational time
- ncasc: Number of cascade and pre-equilibrium models used and their computational time
- bertini: Number of the Bertini model used and their computational time
- isobar: Number of the isobar model used and their computational time
- JAM: Number of the JAM model used and their computational time
- QMD: Number of the JQMD and JQMD-2.0 models used and their computational time
- JAMQMD: Number of the JAMQMD model used and their computational time
- INCL: Number of the INCL model used and their computational time
- INCELF: Number of the INC-ELF model used and their computational time
- frag data: Number of frag data used and their computational time
- SCINFUL: Number of the SCINFUL mode used and their computational time

Citation lists (Category I)

References that must be explicitly cited in the publication using the PHITS result are provided at the end of summary output file.

4.9 Tally output file

Tally results are output in the file with name specified by file parameter in each tally. In the cases of 2-dimensional `axis` (`xy, yz, xz, rz, chart`), the statistical errors of the tally results are separately output in the file with name having `_err` before the extension of the original tally output file. In addition, when `epsout, vtkout`, and `bmpout` are specified in the tally, the image files are generated by changing the extension of the tally output file to ".eps", ".vtk", and ".bmt", respectively. For example, four output files are created from a tally with `file=track.out, axis = xy, epsout = 1`, which are "track.out" (numerical data of the tally results), "track_err.out" (numerical data of the statistical errors of the tally results), "track.eps" (image data of the tally results), and "track_err.eps" (image data of the statistical errors of the tally results). Our recommended extension of the tally output file is ".out", but file names with other extensions (or even file name without extension) are also acceptable.

At the beginning of tally output file, input echo, which repeats all parameters defined in the tally section, is written. Then, the tally results are written in the format that can be read by ANGEL. When parameters other than those specified by `axis` are changed in a tally, multiple pages of graphs are generated, and they are separated by an ANGEL command `newpage:`. Followings are the concrete explanations of tally output files in the cases of 1-dimensional and 2-dimensional axes, exampling "track_eng.out" and "track.out", respectively, in the folder of "recommendation/CosmicRay/track_eng.out". For 2-dimensional axes, the xy-plane is viewed from the + side of the side of the z-axis, the xz-plane from the + side of the y-axis, and the yz-plane from the from the + side of the z-axis.

track_eng.out(Example of 1-dimensional axis plot)

Lines 1–51 (from *[t-track]* to *#– – – – – – – – – – – – – – – – – – –*)

Input echo

Line 53(*#newpage:*)

This line indicates the data for 1st page graph start here.

Line 54(*# no. = 1 reg =1000001*)

Page number (no.) and fixed parameters in this page (reg in this case) are output after the comment mark #. Note that a large cell ID (1000001, 1000002 $\cdots$) is automatically assigned to a combined cell defined in reg parameter.

Line 56(*x: Energy [MeV/n]*)

Label of X-axis

Line 57(*y: Flux [1/cm^2/(MeV/n)/source]*)

Label of Y-axis

Line 58(*p: xlog ylog afac(0.8) form(0.9)*)

ANGEL parameters for specifying the fundamental setting of the graph. For example, `xlog` and `ylog` indicate that X and Y axes should be expressed in the logarithmic scale, respectively. Note that the default setting for x-axis is `xlin` (the linear scale) for mesh-type = 1, 2, 4 and `xlog` for mesh-type = 3,5, while that for y-axis is always `ylog`. ANGEL parameters specified in the tally setting are also written here. Please see Section 6.7.11 ANGEL parameter or ANGEL manual in more detail about the ANGEL parameters.

Line 59(*h: n x y(proton),hh0l n y(neutron),hh0dr n y(Fe),hh0ub n*)

This line specifies how to plot the data written in the following lines in the graph. `n` indicates skip the column, `x` and `y` represents the data for x and y axes, respectively. The name in the brackets (proton, neutron, Fe) is the legend of the data, and `hh0l` indicates that the data should be drawn by histogram (`hh`), no mark (`0`), black line (`l`).

Line 60(*# e-lower e-upper proton r.err neutron r.err Fe r.err*)

This line shows the meaning of data written in the following lines. In general, the data are given in the order of the lower and upper boundaries of the mesh, the tally result, and its relative statistical error. When more than 2 particles are specified in `part`, the tally results and their relative statistical errors are output by adding columns, and they are shown in the same graph.

Lines 61–180

Numerical data of the tally results. Number of lines is equal to the number of meshes, while the number of columns is equal to 2 + (number of particles specified in `part`)×2.

Line 182(*# sum over*)

The total or mean values of the tally results are output in this line, though their statistical errors are not provided.

Line 184(*'no. = 1, reg =1000001'*)

Page title

Lines 185–191(from *msuc:* {*\huge Track Detection in reg mesh*} to *e:*)

Title of the graph and versions of PHITS and ANGEL are output in the format of ANGEL parameter.

Lines 193–199(after *# Information for Restart Calculation*)

Information for restart calculations such as next random seed and total history number is written at the end of each tally output file.

track.out(Example of 2-dimensional axis plot)

Lines 1–97(from *[t-track]* to *#– – – – – – – – – – – – – – – – – – –*)

Input echo

Line 99(*#newpage:*)

This line indicates that the data for 1st page graph start after this line.

Lines 100–103(*# no. = 1 ie = 1 iz = 1 part. = all* から*# mset = 1* まで)

Page number (no.) and fixed parameters in this page (reg, ie, iz, part and #mset in this case) are output after the comment mark #.

Line 105(*'no. = 1, ie = 1, iz = 1, mset = 1'*)

Page title

Lines 106–108(from *msuc:* {*\huge Astronaut dose in pSv/source*} to *msdl:* {*\it calculated by \PHITS 3.28*})

Title of the graph and versions of PHITS and ANGEL are output in the format of ANGEL parameter.

Lines 110–119(from *x: x [cm]* to *p: ymin(-60.00000) ymax(60.00000)*)

ANGEL parameters for specifying the fundamental setting of the graph. For example, `cmin` and `cmax` defined by `c3` and `c4` parameters, respectively, indicate the minimum and maximum value of the tally data plotted in the graph, and `zlog` indicates that the data should be plotted in the logarithmic scale.

Lines 120–121(from *# ny = 120 nx = 120* to *# ((data(x,y), x = 1, nx), y = ny, 1, -1)*)

These lines specify the order of the tally data. In this case, the data are provided in the ascending order of x (from 1 to 120) and the descending order of y (from 120 to 1).

Line 123(*hc: y = 59.50000 to -59.50000 by 1.000000 ; x = -59.50000 to 59.50000 by 1.000000 ;*)

Information on the y and x coordinates of the plotting data. In this case, y(ny) = 59.5 and y(1) = −59.5 with the interval of 1.0, while x(1) = −59.5 and x(nx) = 59.5 with the interval of 1.0.

Lines 124–1563

Numerical data of the tally results. As previously defined, data are given in the order of data(1,ny) data(2,ny) $\cdots$ data(nx,ny) data(1,ny-1) data(2,ny-1) data(3,ny-1) $\cdots$ data(nx,ny-1) $\cdots$ data(1,1) data(2,1) $\cdots$ data(nx,1). Note that new line codes are inserted in every 10 data. Thus, the data are written in 1440 successive lines ($120 \times 120/10$) in this case. If you want to output the tally results in the format that can be directly copy & paste to the spreadsheet software such as EXCEL, please select `2d-type=5` in the tally setting. Please see Section 6.7.13 2d-type in more detail.

Line 1566(*# gshow*)

This line indicates that the data for drawing geometry start here.

Lines 1568–2681

Data for drawing geometry

Lines 2684–2713(from *z: xorg(1.03)* to *p: ymin(-60.00000) ymax(60.00000)*)

ANGEL parameters for specifying the setting of the graph.

Line 2716(*newpage:*)

This line indicates the data for next page graph start here. In different from 1-dimensional axis plot, one graph is created for one `part` data. Thus, the image file generated from this tally output has 4 pages: the 1st page is for `part=all` starting at line 99, the 2nd page is for `part=proton` starting at line 2716, the 3rd page is for `part=neutron` starting at line 5333, and the 4th page data is for `part=Fe` starting at line 7959.

Lines 10566–10572(after *# Information for Restart Calculation*)

Information for restart calculations such as next random seed and total history number is written at the end of each tally output file.

5 Section format

5.1 [Title] section

In the [title] section, the titles of calculations can be defined. Any number of title lines is allowed. Blank lines are skipped in this section.

```
[ Title ]
This is a test calculation of PHITS.
Any number of title lines is allowed.
..........
```

5.2 [Parameters] section

The various parameters of PHITS can be defined in [Parameters] using the following format:

```
[ Parameters ]
para1 = number | file.name
para2 = number | file.name
..........
```

The order of parameters can be changed. As each parameter has a default value, undefined parameters use the default values.

Parameters and default values are listed in the tables shown below. (D=) indicates the default value.

5.2.1 Calculation mode

Table 5.1: icntl parameter

Parameter	Value	Explanation
icntl	(D = 0)	Basic control option.
	= 0	Normal PHITS calculation.
	= 1	Output cross section and kerma factor (please see /phits/sample/icntl1 in detail).
	= 3	Output only input echo for checking memory usage, libraries, and file links.
	= 5	Calculation assuming all regions are void (no reaction, no ionization) for geometry check, and volume and area calculations.
	= 6	Source check. Source positions can be tallied by [t-product].
	= 7	Execute [t-gshow] for graphical output.
	= 8	Geometry output of xyz mesh tally with gshow option for graphical output.
	= 9	Execute [t-rshow] for graphical output.
	= 10	Geometry output of reg mesh tally with rshow option for graphical output.
	= 11	Execute [t-3dshow] for graphical output.
	= 12	Re-calculate using dumpall file. Dumpall file is specified by file(15).
	= 13	Use of function to sum up tally results (Sumtally function).
	= 14	Use of automatic calculation of region volume.
	= 15	Execute [t-wwbg] to obtain [ww bias] parameters.
	= 16	Use of script of analysis function.
	= 17	Execute [anatally] to obtain results of analysis function.

Cross sections and kerma factors used in the PHITS simulation can be outputted by icntl=1. Please see "/phits/sample/icntl1" in more details.

By setting icntl=12, PHITS reads the previous event-by-event calculation result from a dumpall file created when the dumpall=1 option was activated. All of the transport events are reproduced by the dumpall file without being re-calculated again. Thus, maxcas and maxbch cannot be changed. This functionality is suitable for tallying different physics quantanties with the same simulation setup, especially when such a simulation setup is computational time consuming. However, it should be noted that calculation with dumpall=1 may create a huge dumpall file. Please refer to 5.2.26 for more details.

By setting icntl=13, two (or more) tally results can be summed: see Sec. 6.8 for further detail.

By setting icntl=14 with [t-volume], automatic calculations to obtain volumes of specified cells can be performed. See Sec. 7.18 in more detail.

By setting icntl=15 with [t-wwbg], automatic calculations to obtain parameters of [ww bias]. See Sec. 7.17 for further detail.

When use of script of analysis function, icntl=16 should be set in a PHITS input file.

5.2.2 Number of history and Bank

In the distributed-memory parallel calculation, the number of batches should be an integer multiple of $N_{\mathrm{PE}} - 1$, where N_{PE} is the total number of Processing Element (PE). Otherwise, PHITS will automatically convert `maxbch` to an integer multiple of $N_{\mathrm{PE}} - 1$, and adjust `maxcas` such that the total number of histories becomes close to or the same as the given total number of histories. In this case, some comments will be output at the end of an input echo.

Time limit of a PHITS calculation can be activated by assigning `timeout` a positive value. If the CPU time reaches `timeout` in the middle of a batch, PHITS will finish the calculation of all histories in that batch, and then terminates the code. In the distributed-memory parallel computing, `timeout` is ruled by the sum of all CPU time. Note that the function of `timeout` can be used only when the computer's CPU time is correctly retrieved.

To calculate statistical uncertainties, two modes can be selected –"batch variance mode" and "history variance mode"– which calculate standard deviations of tally results by batch and by history, respectively. In both modes, standard deviation σ is calculated as

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N} (x_i w_i / \bar{w})^2 - N\bar{X}^2}{(N-1)}} \tag{1}$$

where N is the total batch number (`istdev=1`) or the total history number (`istdev=2`), x_i is the tally results, w_i is the source weight of each sample, $\bar{X}$ is the mean value of the tally results, and $\bar{w}$ is the source weight of N samples. PHITS calculates the absolute value of the error, and then outputs the relative error obtained by dividing it by the mean of the tally values, $\bar{X}$.

The standard error of the mean is given as $\sigma/\sqrt{N}$. Then, PHITS outputs the ratio calculated by the following formula as relative error in the tally output file.

$$(\mathrm{r.err}) = \frac{\sigma}{\sqrt{N}\bar{X}} \tag{2}$$

The standard errors calculated by the batch variance mode vary by combination of `maxcas` and `maxbch`, even for the same total history number. In principle, a larger `maxbch` provides more reliable statistical uncertainties in batch variance mode but may require a longer computational time. Setting the value of `maxbch` greater than 10 is recommended to obtain reliable results. By contrast, in history variance mode, the standard error depends only on the total history number and is independent of the combination of `maxcas` and `maxbch`. Thus, it is recommended to select the history variance mode (`istdev=2`) in most cases. Note, however, that computational time occasionally becomes extremely long in history variance mode, especially in cases in which there are tallies using many memories, e.g., `xyz` mesh tally with its very fine structure.

In tallies for calculating the standard error of deposition energies by individual history, such as `[t-deposit]` with `output=deposit` and `[t-deposit2]`, the standard errors instead of the standard errors are output as relative errors. The relative standard errors can be estimated as $1/\sqrt{K}$ where K is the number of histories contributing to the tally result. This calculation procedure is independent of the `istdev` parameter.

Statistical errors are generally output in the `r.err` column, which is the rightmost column of tally results. In the case of 2D-plots, such as tallies with `axis=xy` or `rz`, errors are output into another file, '*_err' where '*' indicates the file name specified in the tally. For example, when `file=tally.out`, the name of the error file is "tally_err.out." This error file has the same format of the conventional tally output file; hence, a graph for the error in the 2D-plot can be obtained by ANGEL through a conversion process. See Section 4.9 in more detail.

Note that as what PHITS calculates is the standard error, it means the true value has a 68% of chance falling within the confidence interval. Furthermore, PHITS does not estimate systematic uncertainties contributed by the nuclear reaction models.

When `istdev < 0`, the restart calculation mode is activated; however, if there is no past tally result, a new calculation is started with `istdev=|istdev|`; in other words, the batch and history variance modes are selected for `istdev = -1` and `-2`, respectively. When restarting a calculation, the variance mode is automatically determined from the past PHITS calculation. The tally parameters in the restart calculation must be consistent with the original tally parameters (i.e., parameters that already exist in the previous tally results must remain unchanged); however, new tally sections can be added in the restart calculation.

[31] R. Picard and T. Booth, LA-UR-08-06204.
[32] G. Marsaglia, 2003. Xorshift RNGs. Journal of Statistical Software. 8, 14, 1-6.

Table 5.2: Parameters for number of history and bank

Parameter	Value	Explanation
maxcas	(D = 10)	Number of histories per batch. The upper limit is 2147483647.
maxbch	(D = 10)	Number of batches. The upper limit is 2147483647.
maxbnk	(D = 10000)	Maximum number of secondary particles that can be temporarily stored in computer memory. (D=1000000 when [weight window] is defined.) When a large number of secondary particles are generated by high-energy spallation reactions or when a large number of particle splitting occurs using [importance] or [weight window], a warning like jbnk = 0, ibnk = 1 may appear. In such cases, increasing this value may speed up the calculation. Storing the data of one particle requires about 300 bytes of memory. So, please note that if the value is increased too much, the computer may run out of memory and an error may occur.
iwwbnk	(D=1)	Option to suppress the particle splits as the number of secondary particles stored in memory approaches maxbnk when [weight window] is set. 1: enabled, otherwise disabled.
igeomem	(D=10000)	Maximum reserved memory size for a single cell definition. When you define a complex cell using too much #, you may get an error message that suggests to increase this parameter. When you get the message, you must reduce the number of # used in a cell definition, or increase this parameter to the suggested value or more.
xsmemory	(D=1.0)	Relative size of memory space allocated for nuclear data. In general, this value is not necessary to be changed. If you get an error to request for increasing this value, please follow the request and send your input file to PHITS office.
timeout	(D = -1.0)	Cut-off CPU time for PHITS calculation (in units of second).
istdev	(D = 0)	Control parameter for statistical uncertainty calculation type and restart mode.
	= -2	Restart calculation mode is activated; if there is no past tally result, a new calculation is started with istdev=2.
	= -1	Restart calculation mode is activated; if there is no past tally result, a new calculation is started with istdev=1.
	= 0	istdev is automatically set to 1 for shared-memory parallel computing with italsh=1, and 2 for the other cases.
	= 1	A new calculation is begun. Statistical uncertainties are estimated from the standard error of each batch result.
	= 2	A new calculation is begun. Statistical uncertainties are estimated from the standard error of each history result.
italsh	(D = 0)	Control parameter to share tally variable among multi threads for the OpenMP shared-memory parallel computing. This is useful for parallel computing requiring a large tally memory space because this option enables such a simulation with a smaller physical memory by sharing the tally variables. On the other hand, this option induces a process required to prevent memory corruption which may slow down the computational speed. Test with a small number of histories is recommended before full simulation.
	= 0	Tallies are not shared.
	= 1	Tallies are shared.
ireschk	(D = 1)	Control parameter for tally consistency check.
	= 0	Check consistency between new and old tally settings.
	= 1	No check. This option is useful when setting a very complex tally whose settings are not fully written in the tally output file.

Table 5.3: Parameters for random numbers

Parameter	Value	Explanation
nrandgen	(D=1)	pseudo-random number generator option.
	=0	Linear Congruential Generator is used. Period length is 2^{46}. See the reference[33] for more details.
	=1	xorshift64 is used. Period length is $2^{64} - 1$. See the reference[34] for more details.
irskip	(D = 0)	Random number control.
	irskip > 0	Skip the first irskip number of histories and the random number seeds for debugging. (The total number of history is reduced by irskip.)
	irskip < 0	Begin calculation with the No. irskip+1 random number seeds for executing parallel computing manually. (The total number of history is unchanged.)
rseed	(D=0.0)	Initial random number option.
	rseed=0	As the initial random seed, 6.647299061401E+12 is used. When nrandgen=0, this value is used. When nrandgen=1, 64-bit data corresponding to this value is used.
	rseed≠0	rseed is used as the initial random seed. When nrandgen=0, absolute integer value of rssed is used. When nrandgen=1, 64-bit data corresponding to rssed is used, which can be set to a real and negative value.
itimrand	(D=0)	Option for time dependent initial random number.
	=0	Use a value determined by rseed.
	=1	Get an initial random seed from starting time.
bitrseed		64-bit data of the initial random seed. Default is bit data corresponding to 6.647299061401E+12.

5.2.3 Cut-off and switching energies

Table 5.4: Parameters for cut-off energies (1)

Parameter	Value	Explanation
emin(i)	(D=1.0e-3)	(i=1)Proton cut-off energy [MeV].
	(D=1.0e-11)	(i=2)Neutron cut-off energy [MeV].
	(D=1.0e-3)	(i=3,4,6-10)Cut-off energy for pion+, pion0, muons, and kaons [MeV].
	(D=1.0)	(i=5)Pion- cut-off energy [MeV]. If you set emin(5)<1 MeV, absorption of pion- does not occur at stopping by energy cut-off.
	(D=1.0)	(i=11)Cut-off energy for particle species with itype=11 [MeV].
	(D=1.0e+9)	(i=12,13)Cut-off energy for electrons (i=12), positrons (i=13) [MeV].
	(D=1.0e-3)	(i=14)Cutt-off energy for photons [MeV].
	(D=1.0e-3)	(i=15-19)Cut-off energy for deuterons (i=15), tritons (i=16), ^{3}He (i=17), ^{4}He (i=18), and nuclei (i=19)[MeV/n]. Here, MeV/n means MeV per nucleon. i=particle id; see Table 4.4.
dmax(i)	(D=emin(i))	(i=1,12,13,15,18)Upper energy limit of atomic (i=12,13) or nuclear data (i=1,15,18) library use for i-th particle in MeV (or MeV/n for deuteron and α particle). Note that cross section data library for i=3-11,16,17,19 cannot be used in PHITS. Please see APPENDIX B in more detail.
	(D=20.0)	(i=2)Upper energy limit of nuclear data library use for neutrons [MeV].
	(D=1.0e+3)	(i=14)Upper energy limit of atomic library use for photons [MeV]. Note that the user has to prepare the data library except for dmax(i=2,12,13,14), which are included in the PHITS package.
dpnmax	(D=0)	Upper energy limit of photo-nuclear data library in MeV. Note that dmax(14) is the upper energy limit of photo-atomic data library.
lib(i)		The suffix ID (file extension name) of the default nuclear data library file for particle type i. Note that lib(2) is effective only for dmax(2)>20. If the nuclear data libraries are not available for certain nuclides specified in [material], except for neutron, PHITS automatically employs the nuclear reaction models for the nuclides. For neutron, the conventional nuclear data libraries specified at the top of the address file (JENDL-4.0 in the case of "xsdir.jnd") are selected.
	(D=20h)	(i=1) Suffix ID of the default proton nuclear data library. 20h indicates JENDL-5[35]
	(D=20c)	(i=2) Suffix ID of the default high-energy neutron nuclear data library with dmax(2) above 20 MeV.
	(D=20u)	(i=14) Suffix ID of the default photo-nuclear data library. 20u indicates JENDL-5.
	(D=20o)	(i=15) Suffix ID of the default deuteron nuclear data library. 20o indicates JENDL-5.
	(D=20a)	(i=18) Suffix ID Of the default alpha-induced nuclear data library. 20a indicates JENDL-5.

If values given by these parameters are referred to the upper boundaries of a range, then the given values are not included in the range. If values given by the parameters are referred to the lower boundaries of a range, then the given values are included in the range. For example, a proton right at the emin(1) energy is not cut-off as the cut-off range is $\geqslant 0$ and $<$ emin.

When the kinetic energy of a transport particle is less than emin, the energy cut-off is performed and then its transport calculation is stopped. In this case, the residual kinetic energy is deposited locally at that place, and the particle decays in accordance with decay modes shown in Table 4.5 except for neutrons. If the particle species is positron, annihilation occurs instead of decay.

PHITS uses libraries in the energy region emin < energy < dmax. If emin > dmax is set, no libraries are

[35] Iwamoto O, Iwamoto N, Kunieda S, et al. Japanese evaluated nuclear data library version 5: JENDL-5. J Nucl Sci Technol. 2023;60(1):1-60.

Table 5.5: Parameters for cut-off energies (2)

Parameter	Value	Explanation
esmin	(D=0.001)	Lower energy limit for range calculation of charged particles [MeV/n].
esmax	(D=300000)	Upper energy limit for range calculation of charged particles [MeV/n].
cmin(i)	(D=1.0)	Reaction cut-off energy for i-th particle [MeV]. (The energy unit is MeV/n for i=15-19.) Any reactions (δ-ray productions are included) at energies below cmin(i) are not treated. Note that the default values for neutron, electron, positron, and photon are emin(i). The default value for deuteron (i=15) is 0.001 MeV/n.
etsmin	(D=1e-6)	Lower energy limit of electrons and positrons simulated by track structure mode [MeV].
etsmax	(D=1e-2)	Upper energy limit of electrons and positrons simulated by track structure mode [MeV].
tsmax	(D=1e-3)	Upper energy limit of protons and ions simulated by track structure mode [MeV/n]. If track-structure mode is not used, continuous-slowing approximation is applied. To perform proton and ion track structure simulation using your own PHITS executable file, please refer Section 10.4.

used. The upper energy limits for the use of data libraries are 20 MeV and 100 GeV, respectively for neutron and photon. When EGS5 is not used, the upper energy limits for electrons and positrons are 10 GeV. The default setting of emin(12-14) is automatically adjusted by the negs option; e.g., emin(14)=0.001 if negs=-1. The default settings of emin(2) and dmax(2) are automatically adjusted by the nucdata option; e.g., emin(2)=1.0e-11 if nucdata=1. See Table 5.6 for both settings in more details.

The range table of charged particles is set within esmin < energy < esmax. To use a significantly higher energy, the user should set esmax.

The minimum cut-off energy for charged particles, emin, cannot be set lower than esmin. In such cases, emin is automatically adjusted to esmin.

When [Frag Data] is used, cmin(i) should be set to the minimum incident energy of the defined cross sections.

etsmin>1e-9 (1 meV) can be set. But the setting below 1e-6 (1 eV) is not recommend because computational time becomes extremely long.

etsmax>1e-3 (1 keV) must be set. The setting this parameter below 0.1 (100 keV) is recommend, otherwise the computational time becomes extremely long.

For the track structure mode, emin(12) and emin(13) should be set to 1.0e-3, and EGS5 should be activated (negs=1,2).

When negs≠1,2 is set and the energies of electrons and positrons are between emin(12,13) and dmax(12,13), the original PHITS electron transport algorithm is used. However, please note that this algorithm is not recommended and its accuracy is not checked.

Below eqmdmin, the nuclear reactions of d, t, ^{3}He, ^{4}He, and nuclei are not treated by JQMD. As the applicability of JQMD is restricted in the low energy region and the range of nuclei is very low in the normal material, it is not necessary to consider the low energy reactions of nuclei for the usual case. As a default, high energy heavy ion collisions are treated by JAMQMD above 3.0 GeV/u. This switching energy can be changed by changing e jamqmd. It is possible to calculate even nucleon-induced collisions in JAMQMD by changing eqmdnu, ejamnu, and ejamqmd.

INCL (Intra-Nuclear Cascade of Liège) is a nuclear reaction model for nucleon (proton and neutron), pion, and light-ion (d, t, ^{3}He, or ^{4}He) induced reactions. From version 2.50, INCL is used by default for these reactions if the nuclear reaction model is not explicitly specified. Before using INCL results in a publication, please refer to a document [36] in the footnotes.

Intra-Nuclear Cascade with Emission of Light Fragment (INC-ELF) is a nuclear reaction model for nucleon-induced reactions. Before using results obtained by INC-ELF in a publication, please refer a document[37] in the footnotes.

[36] A. Boudard, J. Cugnon, J.-C. David, S. Leray, and D. Mancusi, Phys. Rev C87, 014606 (2013).

[37] Y. Sawada, Y. Uozumi, S. Nogamine, T. Yamada, Y. Iwamoto, T. Sato, and K. Niita, Nucl. Instr. & Meth. B 291, 38-44 (2012).

Table 5.6: Parameters for transport of photons, electrons, positrons, and neutrons

Parameter	Value	Explanation
negs	(D=-1)	Option for electron, positron, and photon transport.
	=-1	Transport only photons based on the PHITS original algorithm. By selecting this option, emin(14) and dmax(14) are automatically set to 0.001 and 1000.0, respectively. If these parameters are directly specified in the input file, the specified parameters overwrite the setting.
	= 0	Ignore electron, positron, and photon transport. With this setting, emin(12-14)=dmax(12-14)=1.0e+9.
	= 1	Transport electrons, positrons, and photons based on the EGS5 algorithm. By setting this option, emin(12,13) and emin(14) are automatically set to 0.1 and 0.001, respectively, while dmax(12-14) are set to 1000.0. If these parameters are directly specified in the input file, the specified parameters overwrite the setting. file(1) or file(20) must be specified in this option.
	= 2	Same as negs=1, but the maximum energies for electron, positron, and photon, dmax(12-14), are automatically set to 10 TeV (1e+7).
nucdata	(D=1)	Option for automatic setting of emin(2) and dmax(2) for the usage of nuclear data library.
	= 0	No adjustment (nuclear data are not used). With this setting, emin(2)=dmax(2)=1.0e-3.
	= 1	Change the parameters suitable for JENDL-4.0, i.e. emin(2)=1.0e-11 and dmax(2)=20.0. If these parameters are directly specified in the input file, the specified parameters overwrite the setting. file(1) or file(7) must be specified in this option.
ieleh	(D=0)	Options for electron and positron transport.
	= 0	No slowing down and no reaction in the energy region above dmax(12).
	= 1	When the energy, e, is above dmax(12), the weight value, wgt, is changed to wgt= e/dmax(12), and then the transport calculation is performed with the energy of dmax(12).

JQMD and JQMD-2.0 are nuclear reaction models; in particular, they can be used to describe heavy-ion induced reactions. In PHITS Ver. 2.7 and later, JQMD-2.0 can be used as an alternative to the conventional JQMD. JQMD-2.0[38] describes reactions - particularly peripheral collisions - more reasonably than JQMD. Users should note that JQMD-2.0 may take more than twice the CPU time required by JQMD.

[38] T. Ogawa, T. Sato, S. Hashimoto, D. Satoh, S. Tsuda, and K. Niita, Phys. Rev C92, 024614 (2015).

[39] Total particle production cross sections given in the HPD block of the ACE format.

Table 5.7: Parameters for switching energies 1

Parameter	Value	Explanation
ejamnu	(D=20.)	Switching energy of nucleon-nucleus reaction calculation from Bertini (or JQMD) to JAM model [MeV].
ejampi	(D=20.)	Switching energy of pion-nucleus reaction calculation from Bertini to JAM model [MeV].
eisobar	(D=0.0)	Maximum energy [MeV] of isobar calculation when isobar is defined (isobar=1).
isobar	(D=0)	Options for the isobar model.
	= 0	Without isobar.
	= 1	With isobar.
eqmdnu	(D=20.)	Switching energy of nucleon-nucleus reaction calculation from Bertini to JQMD model [MeV/n].
eqmdmin	(D=10.0)	Minimum energy of JQMD calculation [MeV/n].
ejamqmd	(D=3000.0)	Switching energy from JQMD to JAMQMD [MeV/n].
inclg	(D=1)	Control parameter for use of INCL.
	=0	INCL is not used. For nucleon-induced reactions, JAM, JQMD, or Bertini model is used, depending on the values of ejamnu and eqmdnu. For pion-induced reactions, JAM or Bertini model is used, depending on the value of ejampi. For light-ion-induced reactions, JQMD is used.
	=1	Use of INCL in a proton, neutron, pion, d, t, ^{3}He, or ^{4}He induced reaction.
	=2	Use of INCL in a proton, neutron, or pion induced reaction.
einclmin	(D=1.0)	Minimum energy of INCL calculation [MeV/n].
einclmax	(D=3000.0)	Maximum energy of INCL calculation [MeV/n].
incelf	(D=0)	Control parameter for use of INC-ELF.
	=0	INC-ELF is not used.
	=1	Use of INC-ELF in proton, neutron, and α induced reactions.
	=2	Use of INC-ELF in proton and neutron induced reactions.
eielfmin	(D=1.0)	Minimum energy of INC-ELF calculation [MeV].
eielfmax	(D=3500.0)	Maximum energy of INC-ELF calculation [MeV].
irqmd	(D=0)	Control parameter for use of JQMD or JQMD-2.0.
	=0	Use of JQMD in nuclear reactions.
	=1	Use of JQMD-2.0 in nuclear reactions.

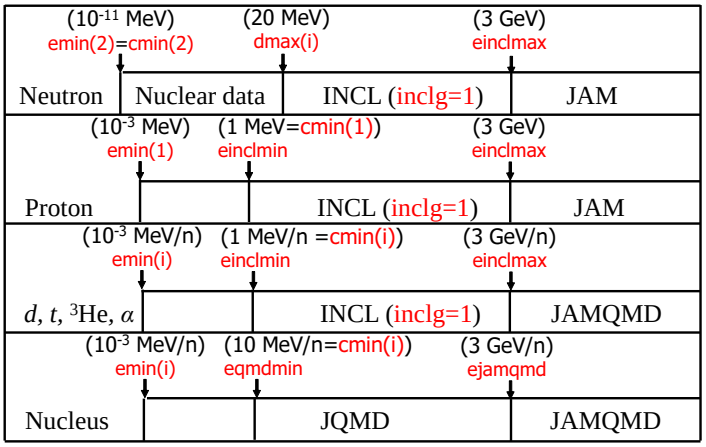


Figure 5.1: Map of Nuclear Reaction Models.

Table 5.8: Parameters for switching energies 2

Parameter	Value	Explanation
iscinful	(D=0)	Option of SCINFUL mode which uses cross section database contained in SCINFUL-QMD for neutron-carbon reaction between 0.1 and 150 MeV.
	=0	SCINFUL-QMD cross section database is not used (use nuclear reaction model).
	=1	SCINFUL-QMD cross section database is used. In this case, [data max] section must be defined in order to decrease the maximum library energy for neutron-carbon reaction down to 0.1 MeV. Please see utility\usrtally\scinful-qmd in more detail.
lionprd	(D=-1)	Neutron data library option for producing light ions (p,d,t,3He,α).
	=-1	Produce the light ions in neutron-induced reactions only at energies between 20 MeV and dmax(2).
	=0	Do not produce the light ions in neutron-induced reactions at energies below dmax(2).
	=1	Produce the light ions in neutron-induced reactions at energies below dmax(2). This option is valid for neutron induced reactions without e-mode. This option does not work if the used nuclear data do not include production reaction cross sections[39]for light ions. Neutron kerma factors (heating numbers) depend on this option and decrease in magnitude with the contributions of the productions. Note that the factors may not be accurate for neutron energies above 20 MeV.
epseudo	(D=10.0)	Energy for calculating the pseudo reaction cross section to determine the flight path of charged particles when their nuclear data library is used. When a smaller value is specified for this parameter, the accuracy in calculating low-energy nuclear reactions would slightly improved, but the computational time would dramatically increase.
tsxcl	(D=1)	Option for nuclear reaction calculations when track structure mode.
	=0	Nullify nuclear reaction cross sections of particles transported by track structure mode. Nuclear reactions are disregarded when KURBUC is used regardless of this parameter.
	=1	Do not nullify.

5.2.4 Cut-off time, cut-off weight, and weight window

Table 5.9: Parameters for cut-off time, cut-off weight, and weight window

Parameter	Value	Explanation
tmax(i)	(D=1.e+9) i = 1-20	Cut-off time for i-th particle [nsec]. (i; particle id, see Table 4.4).
wc1(i) wc2(i) swtm(i)	(D=-0.5) (D=wc1/2) (D=1.0)	Minimum weight for i-th particle. Cut-off weight for i-th particle. Minimum source weight for i-th particle.
wupn	(D=5)	Maximum value of weight window is given by the product of wupn × (minimum value in [Weight Window] section). Here, wupn ≥ 2 should be set.
wsurvn	(0.6*wupn)	Survival weight value. Here, 1 < wsurvn < wupn should be set.
mxspln	(D=5)	Maximum number of split and maximum multiple number of survival. Here, mxspln > 1 should be set.
mwhere	(D=0)	Where the weight window takes place. -1: at nuclear reaction, 0: both, 1: at region crossing.
iwwbias	(D=0) = 0 = 1	Option for [ww bias]. Values defined in [ww bias] are not used. Values defined in [ww bias] are used. The [weight window] parameters multiplied by inverse of the defined biases in [ww bias] are used.
nfcseg	(D=50)	The number of target area divisions when forced collision is applied to charged particles. Smaller values shorten the calculation time, but may result in less accurate calculations for thicker targets, since detailed energy changes within the target cannot be taken into account.
istdcut	(D=0) = 0 = 1	Option for controlling whether tally calculations should be stopped or not when their statistical uncertainties become less than stdcut. Note that the random numbers might be duplicated when you restart the PHITS simulation after terminating some tally calculations in the previous simulation. In the restart case, you have to write the tally lastly stopped in the previous simulation should be written at the top of the input file. Continue all tally calculations until the end of the PHITS simulation. Stop each tally calculation when its statistical uncertainty becomes less than stdcut.
istdbat	(D=0)	Inactivate stdcut until when batch number exceeds istdbat. This parameter is useful particularly for the OpenMP version because statistical uncertainties of tally results are fluctuated very much at the early stage of the PHITS simulation, due to the automatic selection of the batch variance (istdev = 1).

The cut-off time for each particle should be specified as tmax(i) (in units of [nsec]). After arrival at the cut-off time, the particle is killed; although this is not effective in the context of high energy particle transport, it is useful for low energy particle transport calculation.

A particle's weight is affected by the importance, forced collisions, implicit captures, and weight window functions. When the weight is lower than the user-defined weight cut off, a Russian roulette method is applied to determine whether or not the particle is killed. This function is not available for particles defined in the weight window.

Under the Russian roulette method, when the weight wgt is lower than the product, wc2× R, of wc2 and the ratio R of the importances at the source and current points (i.e., when wgt < wc2 × R), the particle survives with a probability wgt /(wc1×R), which is a function of the particle's own weight, WGT. If the particle survives, its weight is changed to wgt = wc1 × R. If wc1 and wc2 are negative, they are respectively set to |wc1| = swtm(i) and |wc2| = swtm(i).

Any particles or regions for which the user has not set importances are given default importances of 1.

In this case of iwwbias=1, the products of the multiplication are output in the input echo of [weight window], and [ww bias] with off is output. If an input file without [ww bias] is used, all values of [ww bias] in the input echo are set to 1.

5.2.5 Stopping power models

Table 5.10: Parameters for stopping power models

Parameter	Value	Explanation
ndedx	(D=3)[40]	Options for dE/dx of charged particle and nucleus. Note that this option is not applicable for electrons and positrons.
	= 0	SPAR[41] for nucleus, NMTC for the rest of particle species.
	= 1	ATIMA[42] for nucleus and proton, NMTC for the rest of particle species.
	= 2	SPAR for nucleus, proton, pion, and muon, NMTC for the rest of particle species.
	= 3	ATIMA for all charged particles.
mdbatima	(D=500)	Maximum database size of ATIMA.
dbcutoff	(D=0.0)	Cut-off energy of ATIMA database [MeV/n].
ih2o	(D=-1)	Setting for the ionization potential of water in ATIMA. This setting is applicable for H_2O only.
	< 0	Use of the default 75 eV.
	> 0	User-defined value [eV].
ifixchg	(D=0)	Option for the charge number of source particles in ATIMA calculations.
	= 0	Effective charge. The charge number is equal to the Z number at high energies, and gradually decreases with decreasing energies.
	= 1	Fixed charge. The charge number is equal to the Z number as a constant, unless it is specified by the izst parameter in the [source] section.
irlet	(D=1)	Options for the method of considering δ-ray energy in calculating dE/dx for charged particle and nuclei.
	= 0	Conventional method (subtract the energy of sampled δ-ray from the unrestricted stopping power).
	= 1	Restricted LET. The cut-off energy of the delta rays is set in the [Delta Ray] section.

From version 2.85, the algorithm for calculating stopping power with ATIMA in PHITS was improved and speeded up. In the old algorithm, calculation of range and stopping power was always performed every time when a new charged particle was produced or a transporting particle entered a new material. The new algorithm creates a database for each combination of transporting particle and material when a new particle-material combination appears, and look up the range and stopping power in the database the next time, instead of calling the high computational-cost routine ATIMA. As such, PHITS simulation using ATIMA is compatible with SPAR in terms of calculation time but with higher accuracy.

Starting from ver. 2.97, the setting of ndedx=3 can be used for nuclei with the atomic number of $93 \leq Z \leq 97$ (from Np to Bk) [43]. Ionization potentials and density correction parameters needed in calculating nuclear stopping powers for those nuclei are adopted from the literature [44] , and SPAR is used in the calculation of electronic stopping power.

The setting of ifixchg = 1 is not appropriate in most circumstances. The charge state of ions changes along their traveling paths in a material owing to attachment and disattachment of electrons. Thus, this setting only suits calculations of energy losses in a thin foil or gas.

[40] The default was ndedx=0 until PHITS ver. 2.00, and ndedx=2 from ver. 2.01 to ver. 2.85.
[41] T.M. Armstrong and K.C. Chandler, ORNL Report, ORNL-4869 (1973).
[42] About ATIMA, see the web site: https://web-docs.gsi.de/~weick/atima/
[43] Prior to ver. 2.96, ATIMA cannot be used for nuclei with $Z > 92$ (U). For those cases, use SPAR by setting ndedx=2.
[44] Table I of R.M.Sternheimer, Atomic data and nuclear data tables, 30, 261-271 (1984).

5.2.6 Elastic scatterings, total cross sections, and reaction cross sections

Table 5.11: Parameters for elastic scatterings, total cross sections, and reaction cross sections

Parameter	Value	Explanation
ielas	(D=2)	Elastic scattering option.
	= 0	Exclude elastic scattering.
	= 1	Include neutron elastic scattering.
	= 2	Include neutron and proton elastic scattering.
ielms	(D=720)	Number of angle groups for elastic scattering.
icxnp	(D=1)	Options for neutron-hydrogen scattering cross sections below 1 GeV.
	=0	Original database included in JAM.
	=1	Data included in JENDL/HE-2007.
icxsni	(D=0)	Options for inelastic scattering, elastic scattering, and total cross sections for nucleon-nucleus collisions.
	= 0	Pearlstein-Niita's formula.[45]
	= 1	KUROTAMA model.
	= 2	Sato's formula.[46]
icrhi	(D=2)	Options for total cross section of nucleus-nucleus collisions.
	= 0	Shen formula.
	= 1	NASA formula.
	= 2	KUROTAMA model.
icrdm	(D=0)	Options for total reaction cross sections for deuteron-induced reactions.
	= 0	Same as icrhi.
	= 1	MWO formula.
icxspi	(D=1)	Options for total reaction cross sections for pion-induced reactions.
	= 0	Geometrical formula.
	= 1	Hashimoto's formula.

The KUROTAMA model covers a wide energy range of projectiles in nucleon-nucleus and nucleus-nucleus reaction cross sections. See the paper [47] for more details, and please cite the reference if your publication is based on the KUROTAMA model.

In the original NASA formula [48], a low-energy multiplier, X_m, is introduced for calculating light-ion cross sections. However, we regard $X_m = 1$ in PHITS.

In the icrdm=1 setting, the MWO formula reproduces reaction cross sections of deuteron-induced reactions for incident energies below 1 GeV, and for target nuclei no lighter than C-12. See the reference [49] for more details; please cite this document in published results based the MWO model.

For the icxspi setting, the Hashimoto's formula has been used as the default model for calculating pion-induced reaction cross sections since version 2.86. This is an empirical formula that reproduces cross section experimental data better than the geometrical formula specified by icxspi=0.

[45] Improved model based on the formulas shown in the reference [S. Pearlstein, Astrophys. J. 346, 1049 (1989)].
[46] T. Sato, R. Kataoka, H. Yasuda, S. Yashiro, T. Kuwabara, D. Shiota, and Y. Kubo, Radiat. Prot. Dosim. 161 (2014) 274-278.
[47] K. Iida, A. Kohama, and K. Oyamatsu, J. Phys. Soc. Japan 76, 044201 (2007), and L. Sihver *et al.*, Nucl. Instr. & Meth. B 334, 34-39 (2014).
[48] R.K. Tripathi, F.A. Cucinotta, and J.W. Wilson, Nuclear Instruments and Methods in Physics Research B 155 (1999) 349-356
[49] K. Minomo, K. Washiyama, and K. Ogata, J. Nucl. Sci. Technol. 54, 127-130 (2017).

5.2.7 Decays and de-excited process models

Table 5.12: Parameters for decays and de-excited process models

Parameter	Value	Explanation
nevap	(D=3)	Options for evaporation model.
	= 0	Without evaporation model.
	= 3	Using GEM model.
ngem	(D=1)	Options for the version of the GEM model.
	= 0	Use the default version (currently ver.1).
	= 1	Use Ver.1.
	= 2	Use Ver.2 (including nucleon-gamma emission competition、KTUY data[50] for shell correction).
ifission	(D=1)	Options for fission reaction model.
	= 0	Use the default version (currently Ver.1).
	= 1	Use Ver.1.
	= 2	Apply Iwamoto's model[51] to nuclei with charge less than 89.
igamma	(D=2)	γ decay options for residual nuclei[52] .
	= 0	Without γ decay.
	= 1	With γ decay.
	= 2	With γ decay based on the EBITEM model.
	= 3	With γ decay and isomer production based on the EBITEM model.
	=-1,-2,-3	Same as 1, 2, 3, except that Doppler effect owing to the motion of excited nuclei is disregarded. Do not use unless necessary, as it distorts physical reaction processes.
ismm	(D=0)	Control parameter of Statistical Multi-fragmentation Model (SMM).
	= 0	SMM is not used.
	= 1	SMM is used. When a JQMD calculation is performed, switching time from JQMD to GEM changes from the default value of 150 fm/c to 75 fm/c.

Setting igamma=3 enables obtaining of information on isomer production using [t-yield] with axis=chart or dchain: see reference [53] in the footnotes for details.

With the Statistical Multi-fragmentation Model (SMM), PHITS improves the accuracy of calculating the production cross sections of light and medium-heavy fragments for collisions of heavy ions such as Pb or Hg or for incident energies over 100 MeV/n. It should be noted that the computational time becomes long when using this model: see the references [54] [55] for details.

[50] H. Koura, T. Tachibana, M. Uno, and M. Yamada, RIKEN Accel. Prog. Rep. 36 (2003) 9: H. Koura, TOURS Symposium on Nuclear Physics V, AIP Proceedings 704 (2004) 60.

[51] Hiroki Iwamoto, and Shin-ichiro Meigo, Journal of Nuclear Science and Technology, 56 (2019) 160-171.

[52] Prior to ver. 2.73, file(14)=trxcrd.dat is required for igamma=1-3.

[53] T. Ogawa, S. Hashimoto, T. Sato, and K. Niita, Nuclear Instruments and Methods in Physics Research B 325 (2014) 35-42.

[54] T. Ogawa, T. Sato, S. Hashimoto, and K. Niita, Nuclear Instruments and Methods in Physics Research A 723 (2013) 36-46.

[55] J.P. Bondorf, A.S. Botvina, A.S. Iljinov, I.N. Mishustin, and K. Sneppen, Physics Reports 257 (1995) 133-221.

5.2.8 Event generator mode and kerma approximation

Table 5.13: Parameters for event generator mode and kerma approximation

Parameter	Value	Explanation
e-mode[56]	(D=0)	
	= 0	Simple sampling mode (non event generator mode).
	= 1	Event generator mode Ver.1 (simple sampling + evaporation model).
	= 2	Event generator mode Ver.2 (Ver.1 + accurate charged particle yields + accurate neutron energy distribution). When e-mode≠0 and high-energy nuclear data libraries[57]are used, PHITS determines the reaction point using the total cross section contained in the high-energy nuclear data libraries while the information on secondary particles is evaluated using the nuclear reaction models. Even in this case, the event-generator mode ver. 1 or 2 is still used for neutrons below 20 MeV.
ikerman	(D=0)	Control parameters for the use of kerma approximation for neutrons in [t-deposit].
	= 0	Automatic selection. Kerma approximation is not used when event generator mode is activated (e-mode $\geq$ 1).
	= 1	Kerma approximation for neutrons is always not used.
ikermap	(D=0)	Control parameters for the use of kerma approximation for photons in [t-deposit].
	= 0	Automatic selection. Kerma approximation is not used when EGS5 mode is activated (negs=1,2), or when emin(12) and emin(13) are set below 10 MeV.
	= 1	Kerma approximation for photons is always not used.

The event generator mode (e-mode=1,2) is essential for calculation of event-by-event quantities, such as detector signals, residual recoiled nuclides, lineal energies, attributed to neutrons below 20 MeV. The simple sampling mode (e-mode=0) reproduces secondary neutron spectra given by the nuclear data library better than e-mode=1,2. Note that e-mode=0 does not consider the conservation of energy and momentum. See Sec. 5.2.27 for further discussion of the use.

[56] Until ver. 2.73, file(14)=trxcrd.dat is required for e-mode=1,2.
[57] Including charged-particle and photo-nuclear data libraries.

5.2.9 Adjoint (backward-in-time) mode

Table 5.14: Parameter for adjoint mode

Parameter	Value	Explanation
iadjoint	(D=0)	Option for adjoint mode.
	= 0	Turn off adjoint mode (conventional forward transport).
	= 1	Use adjoint mode for photon transport (see "/phits/utility/adjoint/adjoint").
	= 2	Use adjoint mode for charged-particle transport (see "/phits/utility/adjoint/charged")

Adjoint mode is a calculation mode that performs particle transport in backward time. Adjoint mode calculations can be more efficient than conventional forward mode calculations for transport problems where the source region is much larger than the tally region. The source and tally regions are switched in an adjoint mode calculation compared to the corresponding forward mode calculation. Adjoint photons start within the tally region of the forward calculation and are tallied in its source region. The PHITS adjoint mode support the transport of only photons below 3 MeV (iadjoint=1) and charged-particles excluding electrons and positrons (iadjoint=2).

The adjoint mode for photon (iadjoint=1) should be used in tandem with [t-adjoint]. Adjoint photons undergo reserve interactions with matter. The interactions are modeled for continuous energies up to 3.0 MeV. See reference [58] for more details. When using the adjoint mode, you must also set these parameters as follows:

```
dmax(14) = 3.0
negs = -1
file(7) = file(1)/data/xsdir.adj
```

The geometry of the [source] in an adjoint mode calculation should match the geometry of the tally region in the corresponding forward mode calculation. Within the adjoint mode [source] section, set proj=photon and define a uniform source energy distribution with minimum and maximum energies covering the range of energies you want to tally. In adjoint mode, the minimum and maximum energies of the e-type mesh in [t-adjoint] should match the minimum and maximum energies set in the adjoint [source] section. Use e-smin and e-smax in [t-adjoint] to set the minimum and maximum energies, respectively, of the real photon source.

The adjoint mode for charged particles (iadjoint=2) should be used in tandem with conventional tallies such as [t-track] and [t-cross]. In this mode, charged particles gain the energy equal to what they should lose in the forward mode, as if they had negative stopping power. There is no energy restriction in this mode, but nuclear reaction cannot be considered. The maximum flight mesh size should be small in this mode, and thus, deltm should be set to a small number (less than 0.01) while confirming that changing this value will not significantly change the results.

[58] A. Malins *et al.*, "Continuous energy adjoint transport for photons in PHITS " EPJ Web Conf., 153, 06001 (2017). DOI: 10.1051/epjconf/201715306001

5.2.10 Nuclear reaction model options

Table 5.15: Parameters for nuclear reaction model options

Parameter	Value	Explanation
inmed	(D=1)	Nucleon-nucleon cross section options for Bertini model.
	= 0	Free (nmtclb25.dat).
	= 1	Cugnon old (nmtclb95.dat); the in-medium effect for nucleon-nucleon cross sections is included.
	= 2	Cugnon new (nmtclb30.dat).
andit	(D=0)	Δ angular distribution for Bertini.
	= 0	50 % isotropic, 50 % forward.
	= 1	All isotropic.
	= 2	All forward.
iidfs	(D=0)	Options for neutron-induced fission.
	= 0	Use the nuclear data library.
	= 1	Neutron multiplicity ν and neutron energy spectrum are taken from the literature.
idwba	(D=0)	Options for the Distorted Wave Born Approximation (DWBA) spectra.
	= 0	Without discrete DWBA spectra.
	= 1	With discrete DWBA spectra.
npidk	(D=0)	Treatment of negative-charged particles listed in Table 4.5 when they are below cut-off energies.
	= 0	Force to be absorbed by nucleus.
	= 1	Sets to decay.

If iidfs=1 is specified, the multiplicity ν and energy spectra of neutrons from neutron-induced fission taken from reference [59] are used instead of the nuclear library data for the following 18 nuclei: U-238, Pu-238, Pu-240, Pu-242, Cm-242, Cm-244, Cf-252, Th-232, U-232, U-233, U-234, U-235, U-236, Np-237, Pu-239, Pu-241, Am-241, and Bk-249.

With the idwba=1 setting, the discrete spectra calculated by the DWBA are added to the neutron and proton spectra obtained using other nuclear reaction models for the following reactions:

^{7}Li$(p,n)^{7}$Be reactions at 30–400 MeV
^{9}Be$(p,n)^{9}$B reactions at 10–50 MeV
^{6,7}Li$(d,n)^{7,8}$Be and ^{6,7}Li$(d,p)^{7,8}$Li reactions at 10–50 MeV
^{9}Be$(d,n)^{10}$B and ^{9}Be$(d,p)^{10}$Be reactions at 5–25 MeV
^{12,13}C$(d,n)^{13,14}$ and ^{12,13}C$(d,p)^{13,14}$C reactions at 10–50 MeV

See the reference [60] for more details.

If a particle with a decay channel (See Table 4.5) has an energy lower than the cut-off, the particle decays completely. Negative-charged pions (π^-) with the setting of npidk=0 are forced to be absorbed by nucleus. If π^- is not able to be absorbed in the program, the particle decays (to prevent an infinite calculation loop).

[59] J. M. Verbeke, C. Hagmnn, and D. Wright, UCRL-AR-228518 (2014).
[60] S. Hashimoto *et al.*, Nucl. Instr. & Meth. B 333, 27-41 (2014).

5.2.11 Options for low energy neutrons

The following parameters are used for neutrons with energies less than 20 MeV.

Table 5.16: Parameters for low energy neutrons

Parameter	Value	Explanation
emcnf	(D=0.0)	Threshold energy for neutron capture [MeV]: implicit capture is considered above this energy; analog capture is considered below this energy.
dnb	(D=-1) =-1 = 0 > 0	Number of delayed neutron produced by fission. Natural sampling. No delayed neutrons. Specify number of delayed neutrons.
nonu	(D=1) =0 =1	Options for fission neutrons Turn off fission: fission is treated as simple capture. Normal fission producing neutrons.
isaba	(D=0) = 0 = 1	Option for $S(\alpha,\beta)$. No interpolation. Perform interpolation of the data of $S(\alpha,\beta)$, which should be specified for calculations with thin targets in which particle scattering rarely occurs.
isans	(D=0) = 0 = 1	Option for small angle neutron scattering (SANS) in thermal neutron scattering library. No SANS model. Activates the SANS model included in the thermal scattering file. Requires a special thermal scattering file with the SANS parameters.

5.2.12 Photon transport based on the PHITS original model

Table 5.17: Parameters for photon transport based on the PHITS original model

Parameter	Value	Explanation
emcpf	(D=100)	Maximum energy in the detailed photon model [MeV].
nocoh	(D=0) = 0 = 1	Photon coherent scattering options. With coherent scattering. Without coherent scattering.
xnum	(D=1) = 0 > 0	Options for X-ray photon. Not create X-ray photon. Number of analog X-ray photons.

5.2.13 Parameters for EGS5

Table 5.18: EGS5 parameters 1

Parameter	Value	Explanation
ipegs	(D=0)	Options for controlling PEGS5 (invalid when negs≠ 1, 2).
	= -1	Output PEGS5.inp: Yes, Execute PEGS5: Yes, Execute PHITS: No.
	= 0	Output PEGS5.inp: Yes, Execute PEGS5: Yes, Execute PHITS: Yes.
	= 1	Output PEGS5.inp: No, Execute PEGS5: Yes, Execute PHITS: Yes.
	= 2	Output PEGS5.inp: No, Execute PEGS5: No, Execute PHITS: Yes.
imsegs	(D=1)	Options for treating multiple scatterings in EGS5.
	= 0	Original EGS5 method. With this option, the starting scattering strength is used only when the electron is generated as a source particle. After several steps, the maximum scattering strength is used.
	= 1	Original PHITS-EGS5 method. Use starting scattering strength whenever an electron enters a new material. This option enables transporting electrons in thin foils without changing the chard parameter; the computational time becomes slightly longer.
iegsout	(D=0)	Options for controlling EGS5 output files.
	= 0	Delete all EGS5 output files when PHITS calculation is finished. Some files remain when the user self-terminates the PHITS calculation.
	= 1	Keep egs5.inp, pegs5.dat and pegs5.msfit and delete the others.
	= 2	Keep all EGS5 output files. See EGS5 manual[61] for details on each file.
iegsrand	(D=-1)	Options for random number used in EGS5. Setting parameter to 0 or to a positive value hides the MPI parallelization and the restart-calculation function.
	< 0	Use random number generated by PHITS.
	= 0	Use random number generated by EGS5 with the default initial random seed (314159265).
	> 0	Use random number generated by EGS5 with the initial random seed = iegsrand.

PEGS is the prepocessor for EGS. It contains a set of fortran subprograms, which negerate material data for the use of EGS5. It is needed to change the parameter ipegs in order to change the input file of PEGS5 (pegs5.inp). Note that, to retain PEGS5-related files after executing PHITS, iegsout automatically becomes 1 (or greater) when ipegs is set ≥ 1.

When using EGS5 (negs=1,2), the default values of the minimum energies for electron, positron, and photon transports are changed to emin(12,13)=0.1, emin(14)=0.001. The maximum energies are set to dmax(12-14)=1,000.0 (1 GeV) for negs=1,2 and dmax(12-14)=1e+7 (10 TeV) for negs=2 respectively. These values can also be changed by explicitly setting the related parameters. Note that the default values of several EGS5 parameters differ from those employed in the original EGS5 code.

From ver. 2.76, photo-nuclear reactions can be considered in EGS5. Thus, input files with negs=1 and ipnint=1 produce results that differ from those calculated by the previous version of PHITS.

[61] H. Hirayama *et al.*, SLAC-R-730 (2005) and KEK Report 2005-8 (2005).

Table 5.19: EGS5 parameters 2

Parameter	Value	Explanation
iedgfl	(D=1)	Option for K and L-edge fluorescent photons.
	= 0	Do not explicitly treat K and L-edge fluorescent photons.
	= 1	Explicitly treat K and L-edge fluorescent photons.
iauger	(D=1)	Option for K and L-edge Auger electrons.
	= 0	Do not explicitly treat K and L-edge Auger electrons.
	= 1	Explicitly treat K and L-edge Auger electrons.
iraylr	(D=1)	Option for coherent (Rayleigh) scattering.
	= 0	Do not consider coherent scattering.
	= 1	Consider coherent scattering.
lpolar	(D=0)	Option for linearly polarized photon scattering (not available in the PHITS code at this moment).
	= 0	Do not consider linearly polarized photon scattering.
	= 1	Consider linearly polarized photon scattering.
iunrst	(D=0)	Option for outputting stopping powers by PEGS5. When this parameter is changed, ipegs must be set to -1 to not to execute PHITS calculations, and to prevent obtaining incorrect results.
	= 1	Unrestricted collisional stopping power only.
	= 2	Unrestricted collisional and radiative stopping power.
	= 3	Unrestricted collisional and restricted radiative stopping power.
	= 4	Restricted collisional and unrestricted radiative stopping power.
	= 5	Unrestricted radiative stopping power only.
	= 6	Restricted radiative stopping power only.
	= 7	Restricted collisional stopping power only.
chard	(D=0.1)	Array of 'characteristic dimensions,' or representative size (in units of cm) of each material. This parameter controls the electron step-size in each material. The value must be decreased when simulating electron scattering in thin foils by setting imsegs=0. When a negative value is set for this parameter, the actual chard values used in EGS5 are adjusted to their absolute values divided by the material density in g/cm^3. This option is useful for simulating electrons in huge objects (of km order).
epstfl	(D=0)	Option for using density correction factors used in the evaluation of ICRU90.
	= 0	Do not use ICRU90 density correction factors.
	= 1	Use ICRU90 density correction factors. In this case, PHITS automatically find liquid water, graphite, and dry air from the [material] section based on the compositions and densities of each material. The criterions for the material judgement are as follows: Liquid water: A material composed of H and O with density between 0.9 and 1.1 g/cm^3 Graphite: A material composed of C with density between 1.55 and 2.40 g/cm^3 Dry air: A material composed of C, N, O, and Ar with density below 0.03 g/cm^3. This threshold density can be changed by gasegs parameter.
gasegs	(D=0.03)	A material with density below this value is automatically regarded as gas in PEGS.

Table 5.20: EGS5 parameters 3

Parameter	Value	Explanation
incohr	(D=1)	Option for incoherent scattering function.
	= 0	Do not use incoherent scattering function for Compton scattering.
	= 1	Use incoherent scattering function for Compton scattering (ibound is automatically set to 1).
iprofr	(D=1)	Option for Doppler broadening of Compton scattering energies.
	= 0	Do not consider Doppler broadening.
	= 1	Consider Doppler broadening (incohr and ibound are both automatically set to 1).
impacr	(D=1)	Option for electron-impact ionization (EII).
	= 0	Do not consider EII.
	= 1	Consider EII.
ieispl	(D=0)	Option for split of X-rays generated by electron-impact ionization.
	= 0	No splitting.
	= 1	Splitting.
neispl	(D=0)	Number of electron-impact ionization X-rays for splitting when ieispl=1.
ibrdst	(D=1)	Option for determination of bremsstrahlung polar angle.
	= 0	Angle given by m/E.
	= 1	Sampling.
iprdst	(D=1)	Option for determination of polar angles of pair electrons.
	= 0	Angle given by m/k.
	= 1	Sampling.
iphter	(D=1)	Option for determining angular distributions of photoelectrons.
	= 0	Emission in the direction of incident photon.
	= 1	Sampling.
ibound	(D=1)	Option for Compton cross section.
	= 0	Use free Compton cross section.
	= 1	Use bound total Compton cross section.
iaprim	(D=1)	Option for correcting bremsstrahlung cross section.
	= 0	Use Motz *et al.* empirical.
	= 1	Normalize integrated cross section to ICRU-37 radiative stopping power.
	= 2	No correction.

5.2.14 Photon- and muon-induced reaction models

Table 5.21: Parameters for photon- and muon-induced reaction models 1

Parameter	Value	Explanation
ipnint	(D=0)	Options for photo-nuclear reaction. [62]
	= 0	This reaction mechanism is not considered.
	= 1	This reaction mechanism is considered, except for nuclear resonance fluorescence.
	= 2	This reaction mechanism is considered, including nuclear resonance fluorescence. (See notes below this table for polarization of incident photons.)
pnimul	(D=1.0)	Multiplying factor to increase sampling probability of photo-nuclear reaction.
imucap	(D=1)	Options for capture reactions of negative-muons considering characteristic X-ray production from muonic atoms and nuclear absorption.
	= 0	This reaction mechanism is not considered.
	= 1	This reaction mechanism is considered with default parameters used for calculation of characteristic X-ray production.
	= 2	This reaction mechanism is considered with user defined parameters written in file(28) used for calculation of characteristic X-ray production.
iqmdscm	(D=0)	Option for Surface Coalescence Model[63](SCM) in the calculation of capture reactions of negative-muons.
	= 0	Turn off SCM.
	= 1	Activate the SCM.
scm_h0	(D=0.25)	Adjustable parameter of SCM, $h0$. [GeV fm/c]
scm_d	(D=1.75)	Adjustable parameter of SCM, D. [fm]
scm_rcls	(D=-1.124)	Adjustable parameter of SCM, $Rcls$[fm]. The adjustable parameter of SCM, $R0$ [fm], is calculated as follows:
	> 0	$R0 = Rcls$
	< 0	$R0 = \|Rcls\| * A^{-1/3}$

For the nuclear resonance fluorescence considered in ipnint=2, polarization of incident photons can be specified using sx,sy,sz in the [source] section. The angular distribution of the scattered photons is determined with regard to the polarization.

[62] Until ver. 2.30, this parameter was named ipngdr.
[63] Y. Watanabe and D. N. Kadrev, Proc. Int. Conf. Nucl. Data Sci. Technol. 2007 (ND2007) (2007) 1121-1124.

Table 5.22: Parameters for photon- and muon-induced reaction models 2

Parameter	Value	Explanation
imuint	(D=1) = 0 = 1	Options for muon-induced nuclear reaction based on the virtual photon theory. This reaction mechanism is not considered. This reaction mechanism is considered.
imubrm	(D=1) = 0 = 1	Options for muon-induced bremsstrahlung. This reaction mechanism is not considered. This reaction mechanism is considered.
imuppd	(D=1) = 0 = 1	Options for muon-induced electron-positron pair production. This reaction mechanism is not considered. This reaction mechanism is considered.
emumin	(D=200.0)	Minimum energy of muon-induced nuclear reaction [MeV]. The change of emumin also changes the reaction cross sections over the entire energy range.
emumax	(D=1.0e+6)	Maximum energy of muon-induced nuclear reaction [MeV].

5.2.15 Muon production model

Table 5.23: Parameters for muon production models

Parameter	Value	Explanation
igmuppd	(D=0) = 0 = 1	Options for photon-induced muon-pair production. This reaction mechanism is not considered. This reaction mechanism is considered. If you take into account for multiple Coulomb scattering of the generated muons, please set nspred=-2.
gmumul	(D=1.0)	Multiplying factor to increase sampling probability of photon-induced muon-pair production.

- For muon pair production caused by positrons ($e^+e^- \rightarrow \mu^+\mu^-$) , please refer to Section 5.29.1 of the [User Defined Interaction] section
- See the reference [64] for more details of muon-pair production by igmuppd.

[64] Y. Sakaki et al., Implementation of muon pair production in PHITS and verification by comparing with the muon shielding experiment at SLAC, Nucl. Instr. & Meth. A 977, 164323 (2020).

5.2.16 Neutrino-induced reaction model

Table 5.24: Parameters for neutrino-induced reaction model

Parameter	Value	Explanation
ntrnore	(D=0)	Options for neutrino-induced reactions.
	= 0	This reaction mechanism is not taken into account.
	= 1	This reaction mechanism is taken into account.

By setting ntrnore=1, following reactions are considered; ^{1}H($\overline{\nu}_e, e^+$)n, ^{2}H($\overline{\nu}_e, \overline{\nu}'_e$)$np$, ^{2}H($\nu_e, e^-$)$2p$, ^{2}H($\nu_e, \nu'_e$)$np$, and elastic scattering with electrons. Above 20 MeV, the cross sections of some reactions channels are extrapolated. Note that, the reaction channels above 150 MeV (e.g. charge-current reactions of μ-neutrinos) are not considered. Table 5.25 shows neutrino reaction systems considered in the current version of PHITS. In the case of that considered reaction channel, ∞ or values are shown as an energy upper-limit.

Table 5.25: List of neutrino reaction systems considered and energies above which cross sections are extrapolated. $\times$: unimplemented, -: reaction is not existing.

Charge current

	e^-	^{1}H	^{2}H
ν_e	∞	-	20
$\overline{\nu}_e$	-	150	20
ν_μ	$\times$	-	$\times$
$\overline{\nu}_\mu$	-	$\times$	$\times$
ν_τ	$\times$	-	$\times$
$\overline{\nu}_\tau$	-	$\times$	$\times$

Neutral current

	e^-	^{1}H	^{2}H
ν_e	∞	$\times$	20
$\overline{\nu}_e$	∞	$\times$	20
ν_μ	∞	$\times$	20
$\overline{\nu}_\mu$	∞	$\times$	20
ν_τ	∞	$\times$	20
$\overline{\nu}_\tau$	∞	$\times$	20

By using the [User Defined Interaction] section, it is possible to simulate higher energy neutrino reactions. For details, please refer to Section 5.29.1.

5.2.17 Options for Coulomb diffusion, gravitational field, and electromagnetic field

Table 5.26: Parameters for Coulomb diffusion

Parameter	Value	Explanation
nspred	(D=0)	Options for Coulomb diffusion (angle straggling)[65]. Note that this option is not applicable for electrons and positrons.
	= 0	Without Coulomb diffusion.
	= 1	With Coulomb diffusion by the original NMTC model.
	= 2	(Recommended) With Coulomb diffusion by the Lynch's formula based on the Moliere theory[66].
		You can also set nspred=-1, -2. When it is specified as a negative value, Coulomb scattering is always taken into account; when specified as a positive value, Coulomb scattering is taken into account until a source particle occurs the first reaction. To reduce calculation time, setting 0 or positive values is recommended for usual calculations. For transport calculations of charged particles in a large scale with very low density or highly transparent charged particles such as muons, please set the negative values.
ascat1	(D=13.6)	Parameter A in the Lynch's formula (see below) for nspred=2.
ascat2	(D=0.038)	Parameter B in the Lynch's formula (see below) for nspred=2.
nedisp	(D=0)	Energy straggling option for charged particle and nuclei. Note that this option is not applicable for electrons and positrons.
	= 0	Without energy straggling.
	= 1	(Recommended) With Landau Vavilov energy straggling[67].

[65] When describing a scattering process on thin films less than 1cm, delt0 should be set to 1/10 of its thickness.

[66] $\sigma = A\frac{\sqrt{X/X_0}}{p\beta}[1 + \frac{B}{\log_{10}(e)}\log_{10}(X/X_0)]$: Eq.(12) in G.R. Lynch and O.I. Dahl, Nucl. Instrum. Methods Phys. Res, B 58, 6-10 (1991).

[67] About these models, see the web site: http://www.dnp.fmph.uniba.sk/cernlib/asdoc/geantold/H2GEANTPHYS332.html

Table 5.27: Parameters for gravitational field and electromagnetic field

gravx	(D=0)	x-component of gravity direction.
gravy	(D=0)	y-component of gravity direction.
gravz	(D=0)	z-component of gravity direction.
usrmgt	(D=1)	Options for user subroutine of time dependent magnetic field.
	= 1	usrmgt1.f, which includes Wobbler magnet, is used.
	= 2	usrmgt2.f, which includes Pulse magnet, is used.
usrelst	(D=1)	Options for [elastic option].
	= 1	usrelst1.f, for Bragg scattering, is used.
	= 2	usrelst2.f, a sample program, is used.
imagnf	(D=0)	Options for magnetic field.
	= 0	Without Magnetic field.
	= 1	With Magnetic field. If you want to precisely calculate the trajectory of charged particles (except electrons and positrons) under a strong magnetic field that significantly bends their trajectories within a short distance (less than a few cm), set the maximum flight mesh size (deltg) in the magnetic field to a smaller value.
ielctf	(D=0)	Options for electromagnetic field.
	= 0	Without electromagnetic field.
	= 1	With electromagnetic field. If you want to precisely calculate the trajectory of charged particles (except electrons and positrons) under a strong electromagnetic field that significantly bends their trajectories within a short distance (less than a few cm), set the maximum flight mesh size (deltg) in the electromagnetic field to a smaller value.

gravx, gravy, gravz represent the directions of gravity; the gravitational force is effective for neutrons at energies below 1 eV. For example, if gravx=1, gravy=0, gravz=0, the direction of the gravitational force is negative along the x-axis.

5.2.18 Output options (1)

Table 5.28: Parameters for output options (1)

Parameter	Value	Explanation
infout	(D=7)	Option to specify output information in file(6). The notes followed by this table give the meanings of the category codes. See Section 4.8 for the detailed information on the categories.
	= 0	I
	= 1	I, II
	= 2	I, III
	= 3	I, IV
	= 4	I, II, III
	= 5	I, II, IV
	= 6	I, III, IV
	= 7	I, II, III, IV, VI
	= 8	I, II, III, IV, V, VI
ncvalout	(D=0)	Otion to output numbers defined as user-defined variables c*i* to file(6)(D=phits.out). The number is output as many times as specified by ncvalout.
nrecover	(D=0)	Number of warning messages output when the recovery of lost particles succeeds.
ierrout	(D=0)	Option to output the line number, the subroutine and file names where error or warning message is written in the source program.
	= 0	No output.
	= 1	Output the information. For example, ---L:22332/R:param/F:read02.f indicates that the error (or warning) message is written at the 22332th line of read02.f, in subroutine param.
idmprijk	(D=0)	Debug option to check the last random number (possible slow down).
	= 0	rijkdmp.inp will not be created.
	= 1	The history random number is dumped into rijkdmp.inp file. The file is overwritten at each history so the last random number can be obtained at abnormal stop. For MPI parallelization case, multiple files rijkdmp***.inp can be obtained, where *** is the PE number. This option is not compatible with OpenMP.

5.2.19 Output options (2)

Table 5.29: Parameters for output options (2)

Parameter	Value	Explanation
dumpall	(D=0)	dumpall option. Activate and use with icntl=12. See 5.2.26 for more details.
	= 0	No dump.
	= 1	Write down all information on file(15) as binary data.
	= -1	Write down all information on file(15) as ASCII data.
idpara	(D=3)	dumpall file name option in the parallel calculation.
	= 0	/wk/uname/file-name of file(15).
	= 1	/wk/uname/file-name of file(15)+(PE number).
	= 3	File-name of file(15)+(PE number).

In the setting of inpara, igpara, ippara, idpara, "/wk/" and "/uname/" are the default directory name and a user-name read in from the environment variable LOGNAME, respectively. In parallel computing, files corresponding to each processor element (PE) are created for writing the output or the dumped data. If idpara is set to 0 or 1, a file is made in the directory named by /wk/uname/ on each of the nodes. If idpara is set to 1 or 3, the PE number is added at the end of the filename. Each PE writes down its calculation result only on its corresponding file.

5.2.20 Output options (3)

Table 5.30: Parameters for output options (3)

Parameter	Value	Explanation
itall	(D=0)	Options for tally output after every batch.
	= -1	No output.
	= 0	Output only numerical data files.
	= 1	Output both numerical data and image (*.eps) files[68].
	= 2	Output to different files. (file name = specified file name + batch number))
	= 3	Output transition of tally results and these statistical errors as a function of the history number (or batch number). These values are output into a file '*_StD' where '*' indicates the file name specified in the tally section. This function is available only for [t-track] and [t-point].
	= 4	Output to different files by independently summarizing each batch results (file name = specified file name + batch number). This option cannot be selected in the batch variance mode (istdev=1). This function is available only for [t-track] and [t-deposit].
iMeVperu	(D=0)	Options for the unit of energy in tally output.
	=0	Output in MeV.
	=1	Output in MeV/n (MeV per nucleons) for nuclei. The unit automatically changes to MeV for the rest of the particle species.
itstep	(D=1)	Options for the tallying timing when the tracking particle is in a field, which constantly changes the particle's momentum, such as a magnetic, electromagnetic, or gravity field.
	= 0	Tally at reaction or surface cross (normal).
	= 1	Tally at each step of the transport.
imout	(D=0)	Options for the output format of the [material] section.
	= 0	e.g., mat[12], 208Pb.33c.
	= 1	e.g., mat[12], Pb-208.33c.
	= 2	e.g., m12, 82208.33c (MCNP style).
jmout	(D=0)	Options for the output format of material density in [material].
	= 0	No conversion.
	= 1	Conversion into particle density.
	= 2	Conversion into mass ratio (%).
kmout	(D=0)	Options of nuclear data information.
	= 0	No display.
	= 1	Writing in input echo.[69]

When iMeVperu=1, the unit of nucleus energy is changed from [MeV] to [MeV/n] in outputs of [t-track], [t-cross], [t-point], [t-product], [t-time], [t-interact]. For example, in the case of [t-track] with unit = 2, quantities in units of [1/cm^2/MeV/source] are converted to those in [1/cm^2/(MeV/n)/source].

[t-track] uses deltm as the step length to describe particle trajectories in fields that continuouly change particles' momenta, such as a magnetic field. To describe the trajectory curve more smoothly, adjust the setting of deltm.

[68] Note that vtk files are output with both itall=0,1.
[69] From Ver. 2.86, the dmax of each nucleus is also output.

5.2.21 Output options (4)

Table 5.31: Parameters for output options (4)

Parameter	Value	Explanation
matadd	(D=1)	Options when different densities are defined for a given material number in the [cell] section.
	= 0	Use the same material number.
	= 1	Use a new material number.
natural	(D=1)	Options for using the naturally occurring isotopes for nuclear cross section data, and for writing their ratios in the input echo, when a given element is defined without the mass number in [material].
	= 0	Use nuclear cross section data for natural elements, such as ^{nat}Fe and ^{nat}Pb, in the nuclear data library.[70] Isotopes are not output in file(6).
	= 1	Assign all the naturally occurring isotopes to the given element. Isotopes are not output in file(6).
	= 2	Assign all the naturally occurring isotopes to the given element. Isotopes are output in file(6).
iggcm	(D=0)	Option of geometry warnings.
	= 0	No echo.
	= 1	In input echo.
icput	(D=0)	Options for recording and outputting CPU time for each calculation process.
	= 0	Not recorded nor output.
	= 1	Recorded and output.
ipara	(D=0)	Parameter display options.
	= 0	Only described parameters.
	= 1	All parameters.
nwsors	(D=0)	Write down the information on nwsors source particles in file(6).
gsline	(D=2)	Control parameter to set gslat in all tally sections as the default setting; gslat=gsline is used in the sections. When gslat is specified in a tally section, its value is available in the section.
	= 0	Does not draw boundary lines of lattice structures.
	= 1	Draw boundary lines of lattice structures. Note that this setting is not recommended when voxel phantoms are included.
	= 2	Does not draw boundary lines of lattice structures in a cell.
	= 3	Does not draw boundary lines of cells filled by the same material. The lines are not also drawn in the case other than lattice or tetrahedron geometry. When gsline=2,3, cell or lattice numbers are not described even gshow≥ 3.

When two (or more) cells are defined with the same material number but differing densities, with the default setting of matadd=1, such an material will be given a new material number as it is redefined by the other density. If matadd=0 is set, the cells will have the same material number. Note that this setting is invalid in the [mat name color], in which other given material numbers are output as a warning message in the first part of file(6) (D=phits.out).

To save calculation time, CPU time for each calculation process, such as transport, set data, analysis, etc., is not recorded and output by default. To obtain such information, set icput=1.

Setting ipara=1 enables confirmation of all parameters in the PHITS code.

[70] An error message occurs if such a natural element does not exist in the data library.

5.2.22 Voxel and tetrahedron geometry options

Table 5.32: Parameters for voxel and tetrahedron geometry options

Parameter	Value	Explanation
icells	(D=3)	Read and write [cell] section in binary. The concept is similar to ivoxel parameter, but the entire [cell] section is turned into binary files. However, if your geometry contains a numerous number of cells with simple definitions, the computational time may increase by using this option. Furthermore, this function cannot be used when different densities are defined for a given material number.
	= 0	Not using file(19). No input echo for [cell] section.
	= 1	Read [cell] section from binary files named file(19) and file(19).cfg. No input echo for [cell] section. ivoxel is automatically set to 1.
	= 2	Write down [cell] section in binary files named file(19) and file(19).cfg. No input echo for [cell] section. ivoxel is automatically set to 2.
	= 3	Not using file(19) (conventional mode). Output input echo for [cell] section.
ivoxel	(D=0)	Read and write voxel data in binary.
	= 0	Not using file(18).
	= 1	Read from voxel data in binary file(18).
	= 2	Write down voxel data in binary on file(18).
	= 3	Same action as =0 but voxel data echo is included in input echo (no voxel data echo for other options).

The ivoxel parameter can be used to accelerate calcutions with voxel geometry. When performing PHITS calculation with ivoxel=2, voxel data are output in file(18) in binary format, and the calculation is then stopped (until ver. 2.30, calculation was continued). In the next calculation with ivoxel=1, the necessary information is read from file(18) created in the previous calculation with ivoxel=2, so the data output process is omitted and the calculation time is shortened. Note that this parameter is not available in the case that the description after fill includes universe numbers with the transformation of the coordinate system, e.g., 3(1 0 1).

Table 5.33: Parameters for voxel and tetrahedron geometry options

Parameter	Value	Explanation
itetvol	(D=0)	Options to compute volume of tetrahedron geometry for LAT=3.
	= 0	The volume will not be computed.
	= 1	The volume will be automatically computed and used in tallies, etc.
itetra	(D=0)	Read and write tetrahedron geometry data in binary. Binary file must be reconstructed by setting itetra=2 when ntetsurf and/or ntetelem are changed.
	= 0	Not using the data file.
	= 1	Read from tetrahedron geometry data in binary file(30)(D=Tetra.bin).
	= 2	Write down tetrahedron geometry data in binary file(30)(D=Tetra.bin).
ntetsurf	(D=100)	Number of outer surfaces of tetrahedron geometry allowed for each sub-space of the container box.
ntetelem	(D=200)	Number of elements of tetrahedron geometry allowed for each sub-space of the container box.
itetauto	(D=0)	Options to automatically create cells corresponding to universes of tetrahedron geometry.
	= 0	No cell is automatically created.
	= 1	The cells will be automatically created. The text file (.txt) is required for the TetGen format.
itgchk	(D=0)	Option to check irregular zero-volume tetrahedrons and intersections among tetrahedrons in tetrahedron geometries.
	=0	No check.
	=1	The irregular geometry check will be performed and the program ends abruptly without further computation[71]. Choose 0 for transport calculation etc. The first detected irregular geometry will be outputted into the "tet_geoerr.inp" file. The situation can be checked with this file by PHIG3D.

Using the option of itetvol=1, when the tetrahedron geometry is treated with LAT=3, the volume of the universes appearing in tetrahedron geometry can be automatically computed. However, the volume is computed by simply summing up the volumes of individual tetrahedrons, which might produce incorrect values in cases in which tetrahedron geometry is clipped out through the use of nest structure of "universe"and "fill". The computed value with the option of itetvol=1 is given priority over values specified in the [volume] or other sections.

When tetrahedron geometry (LAT=3) is used in PHITS calculation, necessary information for transport calculation is generated by reading the tetrahedron geometry file prior to the actual transport calculation. This process becomes computationally heavy as the number of elements in the tetrahedron geometry increases. In such case, the itetra parameter can be used to reduce the computational time. Performing PHITS calculation with itetra=2 will have the processed tetrahedron geometry data recorded in the binary file file(30)(D=Tetra.bin) followed by the calculation being stopped. In the next calculation with itetra=1, the necessary information is read from the Tetra.bin file, and the data generation process is omitted to reduce the calculation time.

The ntetsurf and ntetelem parameters are associated with the two algorithms sub-dividing the box containing the tetrahedron geometry for PHITS calculations with (LAT=3), which reduce the computational time by limiting the number of elements belonging to each sub-space. One algorithm finds a surface of the tetrahedron geometry when a particle enters the geometry. The container box is sub-divided so that the number of outer surfaces at each sub-space is less than ntetsurf. The other algorithm finds an element of tetrahedron geometry when a particle is initiated inside the geometry. The box containing all tetrahedron elements is subdivided so that the number of elements belonging to each sub-space is less than ntetelem.

Using the option of itetauto=1, cells required by each tetrahedron with specfied universe in tetrahedron geometry are created automatically. The cost manually creating the cells can be reduced with this option when the number of universes contained in tetrahedron geometry increase. This option is especially helpful when the tetrahedron geometry is imported with the NASTRAN bulk data format. See 5.6.5.4 for more details.

[71] Instead of 1, by specifying number $n < 11$, the geometry of the n-th detected error will be outputted into the "tet_geoerr.inp" file. The check will end after finding 10 irregular intersections so no file will be created when a number more than 10 is specified.

5.2.23 About geometrical errors

Table 5.34: Parameters for geometrical errors

Parameter	Value	Explanation
nlost	(D=10)	Acceptable value against lost particle (per 1 PE).
igerr	(D=0)	Number of recoveries from region error.
igchk	(D=0) = 0 = 1	Options for region check. No region check. Set the flight mesh to deltb after region-crossing, and check region.
ichkmat	(D=0) = 0 = 1	Option for checking material numbers and densities of each cell when different densities are defined for a given material number in [cell] section. No check. In the above conditions for the [cell] section, incorrect drawing of the geometry by gshow option may occur. Note that this does not influence the results of the particle transport simulation. Check. The geometry by gshow option is correctly drawn, but it increases the loading time of complex geometry with a lot of cells.
deltb	(D=1.e-9)	Distance of flight mesh [cm] after region-crossing if igchk=1. (deltb is also the distance from the region boundary to where the particle is created by importance and forced collision.)
idelt	(D=1) = 0 = 1	Options for minimum flight mesh. Use values of deltm and deltc as flight mesh maxima. Maximum flight mesh is deltm and deltc divided by the density in the region. Note that deltc is not linked with this parameter when the density is less than 1.0 g/cm^3.
deltm	(D=20.12345)	Maximum flight mesh [cm]. This parameter is linked with idelt.
deltc	(D=2.012345)	Maximum flight mesh [cm] for charged particles (except for electrons and positrons) if their energy or angular straggling is considered (nedisp$\neq$ 0 or nspred$\neq$ 0). This parameter is linked with idelt when the material density is greater than 1.0 g/cm^3.
delt0	(D=0.1)	The maximum step size of charged particles (except for electrons and positrons) before crossing a boundary when their angular straggling is considered (nspred$\neq$ 0). Within this distance, the angular straggling is not considered. When describing a scattering process on thin films less than 1 cm, delt0 should be set to approximately 1/10 of its thickness.
deltg	(D=1.012345)	Maximum flight mesh [cm] in electric and magnetic fields for charged particles (except electrons and positrons).
deltt	(D=1.0)	Maximum flight time [msec] in a time-dependent magnetic field.
deltxyz	(D=0.0001)	Automatic adjustment width for the minimum and maximum values of xyz-mesh in [weight window] and [t-wwg] when they are defined as integers. Note that the minimum value is shifted by $-1*$deltxyz and the maximum by $+2*$deltxyz.

If igchk is set to 1, after a particle crosses the boundary of a region, the particle's location is artificially moved a short distance away from the boundary surface along the forward direction in the PHITS code, and this short distance is defined as "flight mesh."

The purpose of the use of flight mesh is to more accurately determine which region the particle belongs to, for prevention of misjudments due to insufficient significant digits. This function is important in particular for large scale simulations that use curved surfaces.

From ver. 2.80, idelt=1 has become the default setting of PHITS, which sets the quotients of deltm and deltc divided by the region's density (in units of g/cm^3) as the flight mesh maxima. This setting is useful in for shortening the particle transport calculation time in air with a spatial scale over several hundred meters.

5.2.24 Input-output file name

Table 5.35: Parameters for Input-output file name

Parameter	Value	Explanation
file(1)	(D=PHITSPATH)	PHITS installation folder name. If environment variable PHITSPATH is not defined, c:/phits is used as default setting. If this parameter is set properly, there is no need to specify the names of other input files, i.e. file(7, 20, 21, 24, 25, and 26), unless the folder structure of PHITS have not been changed. If these parameters are directly specified in the input file, the specified parameters overwrite the setting.
file(6)	(D=phits.out)	Summary output file name.
file(7)	(D=file(1)/data/xsdir.jnd)	Cross section directory file name.
file(11)	(D=nuclcal.out)	Nuclear reaction output file name.
file(12)	(D=fort.12)	Cut-off neutron output file name.
file(13)	(D=fort.13)	Cut-off γ-ray output file name.
file(10)	(D=fort.10)	Cut-off proton output file name.
file(15)	(D=dumpall.dat)	dump file name for dumpall=1 option.
file(18)	(D=voxel.bin)	File name when using ivoxel=1, 2.
file(19)	(D=gcell.bin)	File name when using icells=1, 2. Two files named file(19) and file(19).cfg are created.
file(20)	(D=file(1)/XS/egs/)	Directory containing the library data for EGS5.
file(21)	(D=file(1)/dchain-sp/data/)	Directory containing the library data for DCHAIN.
file(22)	(D=batch.out)	Name of the file to output information on current batch.
file(23)	(D=pegs5)	Output file name for PEGS5.
file(24)	(D=file(1)/data)	Directory containing RIsource.dat (DECDC database).
file(25)	(D=file(1)/XS/tra)	Directory containing cross section database used for track-structure simulation.
file(26)	(D=file(1)/data/multiplier)	Directory containing the pre-defined [multiplier] sections.
file(27)	(D=file(1)/XS/yield/)	Directory containing the user-defined activation cross section data. The last "/" of the path is needed.
file(28)	(D=file(1)/data/aama.dat)	Input file name for the MUONIC ATOM CASCADE program to use the user-defined parameters.
file(29)	(D=file(1)/data/dedx)	Directory containing database files of the stopping power.
file(30)	(D=Tetra.bin)	File name when using itetra=1, 2.

From ver. 2.74, the file(14) parameter does not have to be specified in the PHITS input file even when setting |igamma|$\geq$1.

5.2.25 Others

Table 5.36: Parameters for others

Parameter	Value	Explanation
inucr		Option for selecting the output quantities in the cross-section and kerma-factor output mode (icntl=1). Please see "phits/sample/icntl1" in more detail.
	= 2	Output total, inelastic, and elastic reaction cross sections of a target nucleus.
	= 12	Output neutron kerma factors of a target material.
	= 14	Output photon kerma factors of a target material.
	= 16	Output stopping power of charged particle in a target material.
	= 100	Output various data contained in cross section libraries of a target material.
idam(i)	integer	User-defined integer variable.
rdam(i)	real*8	User-defined real variable.
	i = 1-100	These values can be used in the PHITS code by common /userp/ idam(100), rdam(100)
icommat	(D=0)	Option for comment marks in [material] section.
	= 0	c␣ (␣ is a space) cannot be used as comment marks.
	= 1	c␣ (␣ is a space) can be used as comment marks.

From version 2.89, in the default setting, c␣ (␣ is a space), cannot be used as comment marks in [material] section. To use c␣ as comment marks in the section, set icommat=1.

5.2.26 Dumpall option

With the setting of `icntl=12`, PHITS reads the detailed information of a completed transport calculation from a `dumpall` file created beforehand by using the `dumpall` option, and reproduces all of the transport result event by event. As such, the input file except the tally sections must be identical as used in the previous calculation when the `dumpall` option is activated. `maxcas` and `maxbch` cannot be changed either, because they are read from the dump file.
This can be a very useful function in order to calculate different tallies which were not used in the previous calculation. However, please be careful that the calculation with the `dumpall` option may create a huge dump file.

The dumped data written in ascii can be used on the computers, but the data written in binary cannot. The data sequence and meaning are given in the following.

(1) NCOL
NCOL is an intrinsic variable in the program and denotes identification of processes.

NCOL		
1	:	start of calculation
2	:	end of calculation
3	:	end of a batch
4	:	source
5	:	detection of geometry error
6	:	recovery of geometry error
7	:	termination by geometry error
8	:	termination by weight cut-off
9	:	termination by time cut-off
10	:	geometry boundary crossing
11	:	termination by energy cut-off
12	:	termination by escape or leakage
13	:	(n,x) reaction
14	:	(n,n'x) reaction
15	:	sequential transport only for tally
16	:	surface cross for WW of xyz mesh

When NCOL=1, 2, 3, the output is finished. The followings are for NCOL≥4.

(2) NOCAS, NOBCH, RCASC, RSOUIN
These four data are written only for NCOL=4 and their meaning are,

NOCAS	:	current history number in this batch
NOBCH	:	current batch number
RCASC	:	real number of NOCAS+maxcas*(NOBCH-1)
RSOUIN	:	sum of the weight of source particle

(3) NO, MAT, ITYP, KTYP, JTYP, MTYP, RTYP, OLDWT
These mean

NO	:	cascade id in this history
MAT	:	material id
ITYP	:	particle type
KTYP	:	particle kf-code
JTYP	:	charge number of the particle
MTYP	:	baryon number of the particle
RTYP	:	rest mass of the particle (MeV)
OLDWT	:	weight of the particle at (x,y,z)

(a) QS
This data is written only for ITYP=12, 13, namely electron and positron. QS is dE/dx for electrons.

QS : dE/dx of electron at (x,y,z)

(4) IBLZ1, IBLZ2, ILEV1, ILEV2
These mean

IBLZ1 : cell id at (x,y,z)
IBLZ2 : cell id after crossing
ILEV1 : level structure id of the cell at (x,y,z)
ILEV2 : level structure id of the cell after crossing

(a) ILAT1
This is a variable of level structure of cell. The next data are written only for ILEV1>0 as

```
write(io) ( ( ILAT1(i,j), i=1,5 ), j=1,ILEV1 )
```

(b) ILAT2
This is a variable of level structure of cell. The next data are written only for ILEV2>0 as

```
write(io) ( ( ILAT2(i,j), i=1,5 ), j=1,ILEV2 )
```

(5) COSTH, UANG(1), UANG(2), UANG(3), NSURF
These variables mean as follows. These had come to be output from ver. 2.30.

COSTH : cosine of an angle of incidence in a surface crossing
UANG(1,2,3) : x,y,z component of a normal vector of its surface, respectively
NSURF : internal number of the surface
Note that this is different from the surface number defined in the [surface] section

(6) NAME, NCNT(1), NCNT(2), NCNT(3)
These mean

NAME : collision number of the particle
NCNT(1,2,3) : values of counter 1, 2, and 3

(7) WT, U, V, W
These mean

WT : weight of the particle at (xc,yc,zc)
U, V, W : unit vector of momentum of the particle

(8) E, T, X, Y, Z
These mean

E : energy of the particle at (x,y,z) (MeV)
T : time of the particle at (x,y,z) (nsec)
X, Y, Z : position coordinate of the preceding event point (cm)

(9) EC, TC, XC, YC, ZC
These mean

EC : energy of the particle at (xc,yc,zc) (MeV)
TC : time of the particle at (xc,yc,zc) (nsec)
XC, YC, ZC : position coordinate of the particle (cm)

(10) SPX, SPY, SPZ
These mean

SPX, SPY, SPZ	:	unit vector of spin direction of the particle

(11) NZST
This is the charge state of particles.

(12) NCLSTS
This variable is written only for NCOL=13, 14, collision case, and means the number of produced particle and nucleus. The next data are written for NCLSTS>0 case.

(a) MATHZ, MATHN, JCOLL, KCOLL
These mean

MATHZ	:	Z number of the mother nucleus
MATHN	:	N number of the mother nucleus
JCOLL	:	reaction type id1
KCOLL	:	reaction type id2

JCOLL and KCOLL indicate the following meaning.

JCOLL

0	:	nothing happen
1	:	Hydrogen collisions
2	:	Particle Decays
3	:	Elastic collisions
4	:	High Energy Nuclear collisions
5	:	Heavy Ion reactions
6	:	Neutron reactions by data
7	:	Photon reactions by data
8	:	Electron reactions by data
9	:	Proton reactions by data
10	:	Neutron event mode
11	:	delta ray production
13	:	Photon reactions by EGS5
14	:	Electron reactions by EGS5

KCOLL

0	:	normal
1	:	high energy fission
2	:	high energy absorption
3	:	low energy n elastic
4	:	low energy n non-elastic
5	:	low energy n fission
6	:	low energy n absorption

(b) ICLUSTS, JCLUSTS, QCLUSTS, JCOUNT
These variables have an array and denote the information on the produced particle and nucleus.

```
do i = 1, NCLSTS
   write(io) ICLUSTS(i)
   write(io) ( JCLUSTS(j,i), j=0,7)
   write(io) ( QCLUSTS(j,i), j=0,12)
   write(io) ( JCOUNT(j,i),  j=1,3)
end do
```

These mean

ICLUSTS kind of particle

0	:	nucleus
1	:	proton
2	:	neutron
3	:	pion
4	:	photon
5	:	kaon
6	:	muon
7	:	others

JCLUSTS(i)

i = 0	:	angular momentum
= 1	:	proton number
= 2	:	neutron number
= 3	:	ityp
= 4	:	status of the particle 0: real, <0 : dead
= 5	:	charge number
= 6	:	baryon number
= 7	:	kf code

QCLUSTS(i)

i = 0	:	impact parameter
= 1	:	x-component of unit vector of momentum
= 2	:	y-component of unit vector of momentum
= 3	:	z-component of unit vector of momentum
= 4	:	$e_{tot} = \sqrt{p^2 + m^2}$ (GeV)
= 5	:	rest mass (GeV)
= 6	:	excitation energy (MeV)
= 7	:	kinetic energy (MeV)
= 8	:	weight
= 9	:	time (nsec)
= 10	:	x coordinate (cm)
= 11	:	y coordinate (cm)
= 12	:	z coordinate (cm)

5.2.27 Event Generator Mode

For Event Generator mode [72], `dmax(2)` should be defined appropriately using information from the nuclear data library. In the special statistical decay model, detailed information on the level structure near the ground state for particle and photon emission is used, which requires setting `igamma=1-3`. The special statistical decay model has been developed based on GEM; to use it, the operator should specify `nevap=3`. The related parameters are automatically set by specifying `e-mode`=1 or 2 (unless explicitly specified, `igamma=2` is selected). For consistency reason, the combination `igamma=1` and `e-mode`=1 is also supported.

Event generator mode Ver.2 is the improved version of the legacy version (i.e., event generator mode Ver.1). In reactions emitting multiple neutrons, the previous event generator mode sampled the first ejectile neutron from the cross-section data and emission of the subsequent particles was simulated using the statistical decay model. By contrast, ver.2 samples all ejectile neutrons from the cross-section data and the statistical decay model is used merely to simulate prompt gamma-ray production. For capture reactions, the previous event generator mode assumed that the target nucleus absorbs incident neutrons, and particle emission was simulated by the statistical decay model. In this updated version, ejectile particle species are fully determined by the statistical decay model. While ver.2 selects the ejectile particle species depending on the reaction channel (i.e., in the (n, α) reaction, emission of only one alpha particle and gamma-rays is allowed).

Under Event Generator mode, the following new observables, which cannot be detected otherwise, are obtained:

(1) The deposition energy distribution by the low-energy neutrons below `dmax(2)` without the kerma approximation is available in the `[t-deposit]` tally by considering contributions from the secondary charged particles. When `e-mode=0`, the deposition energy by the low-energy neutrons in `[t-deposit]` is calculated with the kerma approximation [73].

(2) In `[t-yield]` and `[t-product]`, the yield and product quantities can be tallied below `dmax(2)`.

(3) DPA values are obtained even for neutrons of energies below `dmax(2)`.

[72] Y. Iwamoto *et al.*, International Conference on Nuclear Data for Science and Technology 2007, DOI: 10.1051/ndata:07417; K. Niita *et al.*, International Conference on Nuclear Data for Science and Technology 2007, DOI: 10.1051/ndata:07398; Y. Iwamoto *et al.*, Prog. Nucl. Sci. Technol. 2, 931-935 (2011).

[73] After ver. 3.05. Before ver. 3.04, the `[t-deposit]` tally cannot calculate the deposition energy with the kerma approximation.

5.3 [Source] section

The source information can be set in this section. The source type is specified by the number `s-type=N` based on its geometry as in Table 5.37.

The energy of the source particle can be given by `e0` for mono-energy or `e-type` for energy spectrum [74]. Unit of energy in `[source]` is MeV/n (MeV per nucleons) except for particles not the constitute of nucleon (photons, electrons, pions, muons etc).

Table 5.37: Source type

source type	explanation
`s-type = 1, (4)`	Cylinder (or circle, pencil)
`s-type = 2, (5)`	Rectangular solid (or rectangle)
`s-type = 3, (6)`	Gaussian (x, y, z independent)
`s-type = 7, (8)`	Generic parabola (x, y, z independent)
`s-type = 9, (10)`	Sphere or spherical shell
`s-type = 11`	Phase-space source based on accelerator beam emittance.
`s-type = 12`	Reading the data from decay-turtle output
`s-type = 13, (14)`	Gaussian (*xy*x plane)
`s-type = 15, (16)`	Generic parabola (*xy* plane)
`s-type = 17`	Reading dump file
`s-type = 18, (19)`	Cone
`s-type = 20, (21)`	Triangle prism
`s-type = 22, (23)`	*xyz*-mesh distribution
`s-type = 24`	Tetra-mesh source
`s-type = 26`	Surface definition source
`s-type = 100`	User definition source Edit the usrsors.f and compile PHITS.

In Table 5.37, numbers in the brackets were used until ver. 2.94. If both `e0` and `e-type` are defined from ver. 2.95, the energy is decided according to the previous procedure of `s-type`. For example, `e0` is used when `s-type=1`, and the definition of `e-type` is used when `s-type=4`.

[74] Until ver. 2.94, `s-type` had to be used depending on the definition of the energy, e.g., `s-type=1` with `e0` defined the cylinder source of the mono-energy, or `s-type=4` with `e-type` defined that of the energy spectrum.

5.3.1 <Source> : Multi-source

Using this multi-source function, multiple sources with different source definitions, such as different particle species, geometries, or energy distributions, can be specified. Each source definition begins with <source>=*number*, which specifies the relative weight of such type of source. totfact= defines a global scaling factor.

Table 5.38: Multi-source

parameter	explanation
<source> =	The relative weight of such source definition. This is necessary when multiple sources are defined.
totfact =	(D=1) Global scaling factor.
iscorr =	(D=0) Multi-source correlation option. 0: Uncorrelated multi-source. 1: Correlated multi-source. In this case, source particles from each multi-source definition are generated as an event. This option is useful for estimating detector response by nuclear reactions producing several secondary particles. Note that the locations of all sources generated in an event are not the same, unlike the case of iscorr = 2,3. 2: Correlated multi-source. In this case, the locations of all sources generated in an event are the same. This mode is useful for simulating a nuclear reaction occurring at an arbitrary location. 3: Correlated multi-source. In this case, the locations of all sources generated in an event are the same, and the direction of the second source is opposite to that of the first. This mode is useful for simulating a particle decaying into two particles at an arbitrary location.

Tally results in PHITS calculations are normalized to per source particle, or per unit weight in a more strict sense. For the totfact setting, if it is given a positive value, number of source particles are generated based on the intensity ratio of each source definition; if negative, same number of source particles are generated for each source definition, but their weights are adjusted to match the intensity ratio of all source definitions. If one source definition has a much lower intensity than the others, yet such source definition contributes significantly to the tally results, it is recommended to set totfact<0 to improve calculation efficiency.

For iscorr=1,2,3, <source> should be defined as an integer absolute value. As for normalization, if you want to consider multiple generated radiation as a single source, you should specify the sum of the numbers defined in <source> with totfact. For example, to simulate the (X, $1p2n$) reaction as a single source, use <source>=1 for the proton source, <source>=2 for the neutron source, and totfact=3. Conversely, if you want to consider each radiation produced as a single source, totfact is the same as for iscorr=0.

5.3.2 Common parameters

Common parameters for each source type are shown below. The order of the parameters in the source section is free. If a parameter has a default value (D=***), it can be omitted. The energies of d, t, α, and nuclei are specified by [MeV/nucleon]. Please see sections 5.3.19, 5.3.20, 5.3.21 for the definitions of energy, angular, and time distributions, respectively for source description.

Table 5.39: Common source parameters (1)

parameter	explanation
`proj =`	Projectile: see Table4.4 for specification.

`sx =`
`sy =`
`sz =`
	(D=0) x-component of spin.
	(D=0) y-component of spin.
	(D=0) z-component of spin.

`reg =`
 	
`ntmax =`
	(D=all) The source region can be restricted to the overlap(s) between the regions defined by `s-type` and those specified by this parameter. The format is `reg = { 1 - 5 } 10 34`. The lattice and universe frame can be used as `reg = ( 6 < 10[1 0 0] < u=3 )`. See the section of tally region specification for more details on the syntax.
	Note that this parameter cannot be written in the line right above `<source> =` if multi-source sub-sections are defined; otherwise it will cause a syntax error.
	(D=1000) Maximum number of re-sampling the source location from the region specified by `s-type` when `reg` is specified.

`trcl =`	(D=none) Transform number, or definition of transform.
`wgt =`	(D=1.0) Weight of source particles.
`factor =`	(D=1.0) Scaling factor fot tally results; all tally results are multiplied by the value of `factor`. When `factor` and `totfact` are both defined, `totfact` × `factor` is the actual scaling factor. However, when multi-source is defined, use `totfact` only; do not use this parameter for scaling.
`izst =`	(D=charge of particle species specified by `proj`=) Charge state of source particle. This value influences the particle motions only in the magnetic and electro-magnetic fields defined in `[magnetic field]` and `[electro magnetic field]` sections. In addition, ATIMA with the fixed charge mode (`ifixchg=1`) calculates the stopping power based on this charge state. The charge number defined with `izst` does not change while the particle travels. Note that this parameter is only valid for source particles, i.e., secondary particles produced from nuclear reactions carry the charges of their atomic numbers.
`cnt(i)=`	(D=0) Initial value of counter `i (i=1-3)` of source particle. You can distinguish the contributions from each multi-source definition by assigning different values to their initial counter values.

The projectile direction is specified by three parameters: `dir, phi,` and `dom`. The relation between these is shown in Fig. 5.2, in which the direction is noted as a thick arrow. `dir` is the direction cosine relative to the z axis. `phi` is the azimuthal angle from the x axis and is given in degrees; if not set by the user, the value of the azimuthal cosine is selected randomly. Using the parameter `dom` spreads the direction determined by `dir` and `phi` by the solid angle $2\pi(1-\cos\psi)$, where ψ = `dom` (in degrees). In the PHITS calculation, the angle is selected randomly within the solid angle.

By setting `dir=all`, the direction of the source beam becomes isotropic. To use a specific angular distribution, a subsection in the distribution based on numerical data or an analytic function starting from `a-type` is required.

In `s-type=9`, the definition of `dir` differs. In `s-type=11` and `12`, `dir` can only bet set = −1 or 1.

To transform coordinates, the `trcl` parameter, which specifies the transform number or the definition of the transformation itself, can be used. The relation of `wgt` and `factor` is reciprocal.

If its spin is not defined or zero, a neutron enters a magnetic field without

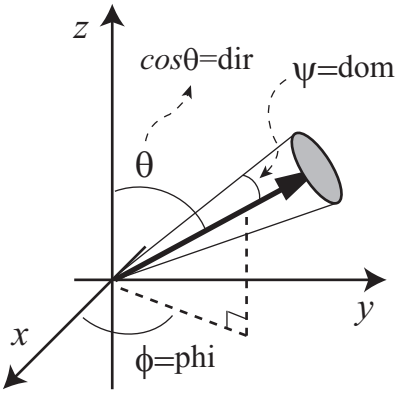


Figure 5.2: Source direction and parameters `dir, phi, dom`.

spin. In this case, the initial spin of the neutron is determined at the entrance
of the magnetic field by the direction of the magnetic field and the polarization factor. If the spin is defined in this section, the neutron enters the magnetic field with the defined spin direction irrespective of the direction of the magnetic field and polarization.

Table 5.40: Common source parameters (2)

parameter	explanation
`ispfs = 0, 1, 2`	(D=0) Neutron sources from spontaneous fission. Note that the RI source function (`e-type=28,29`) can also reproduce spontaneous fission neutrons after version 3.31, so the use of RI source function is recommended to represent mixed fields with fission neutrons and γ-rays. If `ispfs=1 or 2` is specified in the `[source]` section and the following 18 nuclei are chosen as `proj`, neutrons can be defined using spontaneous fission as a source. These nuclei are assumed to be spontaneous fission nuclei: U-238, Pu-238, Pu-240, Pu-242, Cm-242, Cm-244, Cf-252, Th-232, U-232, U-233, U-234, U-235, U-236, Np-237, Pu-239, Pu-241, Am-241, Bk-249. `ispfs=1`: Tally result is normalized by the number of spontaneous fissions. `ispfs=2`: Tally result is normalized by the number of neutrons produced by spontaneous fission. In the case of `ispfs = 1,2`, `e0` and `e-type`, `dir`, and `a-type` are neglected. Unlike the case of RI source (`e-type=28,29`), particle type specified by `proj` is not used, namely, neutrons are generated as source particles.
`ibatch =`	(D=all) Effective batch numbers of each multi-source when `itall=4`. Each batch number should be separated by blank. You can specify successive batch numbers from n1 to n2 in the format of {`n1 - n2`} (n1 should be smaller than n2).

When `ispfs=1,2` is specified, the multiplicity and energy spectrum of neutrons are taken from the reference [75]. The PHITS development team is grateful to Dr. Liem Peng Hong of NAIS, Co., Inc. for his support in developing this function to generate neutron sources.

In the next example, setting `proj`=240Pu and `ispfs`=1 selects neutrons from the spontaneous fission of Pu-240 as source particles. In this case, the value of `e0` is invalid.

```
[ Source ]
    s-type =   1
      proj =   240Pu
        e0 =   1.0
        z0 =   0
        z1 =   0
        r0 =   0.0
       dir =   1.0000
     ispfs = 1
```

[75] J. M. Verbeke, C. Hagmnn, and D. Wright, “Simulation of Neutron and Gamma Ray Emission from Fission and Photofission”, UCRL-AR-228518 (2014).

5.3.3 Cylinder distribution source

Parameters for the cylinder source are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.41: Parameters for cylinder source

s-type = 1	Cylinder or circle source.
x0 =	(D=0.0) *x* coordinate of central axis of cylinder source[cm].
y0 =	(D=0.0) *y* coordinate of central axis of cylinder source[cm].
z0 =	(D=0.0) Minimum *z* of cylinder source[cm].
z1 =	(D=0.0) Maximum *z* of cylinder source[cm], (when z1=z0, circle plane source).
r0 =	(D=0.0) Radius of cylinder source, (when r0=0.0, a pencil source results)[cm].
r1 =	(D=0.0) Inner radius for inner void of cylinder.
dir =	(D=1.0) Direction cosine of projectile against *z* axis.
	If all is set, source is isotropic.
	If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree].
	$= -1$; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

5.3.4 Rectangular solid distribution source

Parameters for a rectangular solid source are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.42: Parameters for rectangular solid source

s-type = 2	Rectangular solid and rectangle source.
x0 =	(D=0.0) Minimum *x* coordinate[cm].
x1 =	(D=0.0) Maximum *x* coordinate[cm].
y0 =	(D=0.0) Minimum *y* coordinate[cm].
y1 =	(D=0.0) Maximum *y* coordinate[cm].
z0 =	(D=0.0) Minimum *z* coordinate[cm].
z1 =	(D=0.0) Maximum *z* coordinate[cm], when (z1=z0), rectangle source.
dir =	(D=1.0) Direction cosine of projectile against *z* axis.
	If all is set, source is isotropic.
	If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree].
	$= -1$; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

5.3.5 Gaussian distribution source (x,y,z independent)

The Gauss distribution is consist of independent Gaussian in each x, y, z direction. Parameters for Gaussian source are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.43: Parameters for Gaussian source

s-type = 3	Gaussian source.
x0 =	(D=0.0) *x* coordinate of Gaussian center[cm].
x1 =	(D=0.0) FWHM in *x* direction[cm].
y0 =	(D=0.0) *y* coordinate of Gaussian center[cm].
y1 =	(D=0.0) FWHM in *y* direction[cm].
z0 =	(D=0.0) *z* coordinate of Gaussian center[cm].
z1 =	(D=0.0) FWHM in *z* direction[cm].
dir =	(D=1.0) Direction cosine of projectile against *z* axis.
	If all is set, source is isotropic.
	If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree].
	$= -1$; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

5.3.6 Generic parabola distribution source (x,y,z independent)

The generic parabola distribution consists of independent parabola in each x, y, z direction. Parameters for generic parabola source are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.44: Parameters for generic parabola distribution

s-type = 7	Generic parabola source.
x0 =	(D=0.0) *x* coordinate of X-parabola center[cm].
x1 =	(D=0.0) X-parabola width[cm].
y0 =	(D=0.0) *y* coordinate of Y-parabola center[cm].
y1 =	(D=0.0) Y-parabola width[cm].
z0 =	(D=0.0) Minimum *z* of parabola[cm].
z1 =	(D=0.0) Maximum *z* of parabola[cm].
rn =	(D=2) Order of generic parabola.
dir =	(D=1.0) Direction cosine of projectile against *z* axis.
	If all is set, source is isotropic.
	If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree].
	$= -1$; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

5.3.7 Gaussian distribution source (x-y plane)

The source of the Gaussian distribution in x-y plane can be set. Parameters for Gaussian source are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.45: Parameters for Gaussian source

s-type = 13	Gaussian source.
x0 =	(D=0.0) *x* coordinate of Gaussian center[cm].
y0 =	(D=0.0) *y* coordinate of Gaussian center[cm].
r1 =	(D=0.0) FWHM in *xy* plane[cm].
z0 =	(D=0.0) Minimum *z* coordinate[cm].
z1 =	(D=z0) Maximum *z* coordinate[cm].
dir =	(D=1.0) Direction cosine of projectile against *z* axis.
	If all is set, source is isotropic.
	If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree].
	= −1 ; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

5.3.8 Generic parabola distribution source (x-y plane)

The source of the generic parabola distribution in x-y plane can be set. Parameters for generic parabola source are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.46: Parameters for generic parabola distribution

s-type = 15	Generic parabola source.
x0 =	(D=0.0) *x* coordinate of X-parabola center[cm].
y0 =	(D=0.0) *y* coordinate of Y-parabola center[cm].
r1 =	(D=0.0) Parabola width in *xy* plane[cm].
z0 =	(D=0.0) Minimum *z* of parabola[cm].
z1 =	(D=0.0) Maximum *z* of parabola[cm].
rn =	(D=2) order of generic parabola.
dir =	(D=1.0) Direction cosine of projectile against *z* axis.
	If all is set, source is isotropic.
	If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree].
	= −1 ; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

5.3.9 Sphere and spherical shell distribution source

Parameters for sphere and spherical shell sources are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.47: Parameters for sphere and spherical shell source

s-type = 9	Sphere and spherical shell source.
x0 =	(D=0.0) x coordinate of sphere center[cm].
y0 =	(D=0.0) y coordinate of sphere center[cm].
z0 =	(D=0.0) z coordinate of sphere center[cm].
r1 =	(D=0.0) Inside radius[cm]. If r1=0, sphere source.
r2 =	(D=0.0) Outside radius[cm].
dir =	(D=1.0) Direction. $-1.0 \leq$dir≤ 1.0: Direction cosine relative to the outward normal line from the center of the sphere. The azimuthal angle is always isotropic. dir = data: Direction cosine distribution relative to the outward normal line from the center of the sphere. a-type subsection is required. The azimuthal angle is always isotropic. dir = all: Isotropic distribution from the surface of the spherical shell. This is used when you want to reproduce the status that radionuclide sources are attached to the spherical surface. dir = -all: Inverse direction against dir=1.0, and with cosine distribution. This is used when you want to irradiate objects isotropically from the outside. By using weights to bias towards the center, this is more efficient for irradiating objects at the center than using dir=iso. However, parameters like ag1 cannot be used. dir = iso: Uniform distribution on a circle of radius r2 on a spherical shell of radius r1 with the direction orthogonal to a circle of radius r2 toward the sphere. Unlike dir=-all, no bias is applied, but parameters like ag1 can be used. This is the recommended setting for cosmic-ray source mode.
ag1 =	(D=-1.0) Minimum cutoff value of the zenith angle (direction cosine relative to the positive z-axis). Only valid when dir=iso.
ag2 =	(D=1.0) Maximum cutoff value for the above.
pg1 =	(D=0.0) Minimum cutoff value of the azimuthal angle (direction in degrees where the source is emitted). Only valid when dir=iso. Can be set within the range of -360 to $+360$.
pg2 =	(D=360.0) Maximum cutoff value for the above.
isbias =	(D=0) Method of handling particles cut off when specifying the angular range using ag1, ag2, pg1, and pg2, or the energy range using eg1 and eg2 in the cosmic-ray source mode (see Sec. 5.3.19.9). =0 Move to the next history by emitting a photon with energy 0. =1 Resample the angle and energy respectively until a particle is generated within their acceptable ranges.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

When you want to isotropically irradiate objects within a certain sphere with radius of r1, set dir=-all (or iso) and r1=r2. In this case, the fluences at all locations inside the sphere will be $1/\pi/(r1)^2$. Source generation locations are uniformly distributed within the spherical shell with inner and outer radii of r1 and r2, respectively. Thus, r1 should be smaller or equal to r2 except for dir=iso, where r1 should be equal to (or larger than) r2.

When a particle with angle below ag1/pg1 or above ag2/pg2 is sampled in the case of isbias=0, it is converted to photon with no energy in order to kill the particles immediately after their generation, i.e., it is counted as a source particle even it is not transported at all. Thus, these parameters are useful for reducing the computational time by terminating particles from unwanted directions. Since this method can be regarded as a kind of variance reduction technique, the normalization factor, totfact, is not necessary to be changed. For example, totfact should be set to the area of the source

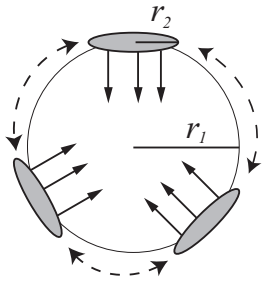


Figure 5.3: Schematic of the source in the case of dir=iso. The actual source generation points are always on the sphere with radius of r1.

circle, π*r2**2, to obtain the normalized tally results per unit fluence irradiation (/cm^2), irrelevant to ag1 and ag2. In the case of isbias=1, PHITS continuously resamples a new particle until the sampled angle meets the request by ag1, ag2, pg1, and pg2. Thus, you can reduce the computational time, but you have to calculate the success rate of the angle sampling and multiply the value with totfact by yourself.

5.3.10 Phase-space source based on accelerator beam emittance

In this mode, sources are defined based on the phase space in vertical to beam axis. This mode is useful for reproducing accelerator sources when its RMS emittance is known. Note that the RMS emittance is not normalized by energy. The followings are the parameters used for defining phase-space source. The order of parameters is free. If a parameter has a default value (D=***), the parameter is optional.

Table 5.48: Parameters for s-type=11

s-type = 11	Uniform distribution in a phase space vertical with beam direction.
x0 =	(D=0.0) x coordinate of beam center[cm].
y0 =	(D=0.0) y coordinate of beam center[cm].
z0 =	(D=0.0) Minimum z[cm].
z1 =	(D=0.0) Maximum z[cm].
rx =	(D=0.0) Gradient of ellipse in a phase space on x direction[rad].
ry =	(D=0.0) Gradient of ellipse in a phase space on y direction[rad].
wem =	(D=0.0) Sampling method from the phase space. (see Fig. 5.4) wem > 0: Uniform distribution. wem represents the RMS emittance in πcm $\times$ mrad. wem = 0: Gaussian distribution.
x1 =	(D=0.0) The ratio of maximum coordinate and angle in cm/mrad for x-axis. (wem>0) Sigma of Gaussian distribution of the x coordinate in cm for rx=0 (wem=0)
y1 =	(D=0.0) The ratio of maximum coordinate and angle in cm/mrad for y-axis. (wem>0) Sigma of Gaussian distribution of the y coordinate in cm for ry=0 (wem=0)
xmrad1 =	(D=0.0) Sigma of Gaussian distribution of the x angle in mrad for rx=0 (Effective only when wem=0)
ymrad1 =	(D=0.0) Sigma of Gaussian distribution of the y angle in mrad for ry=0 (Effective only when wem=0)
x2 =	(D=0.0) Center of the phase space x coordinate in cm.
y2 =	(D=0.0) Center of the phase space y coordinate in cm.
xmrad2 =	(D=0.0) Center of the phase space x angle in mrad.
ymrad2 =	(D=0.0) Center of the phase space y angle in mrad.
dir =	(D=1) Direct cosine. (1 or -1 only).
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

After considering the gradient and center of the phase space, the sampled coordinates of (X, X') and (Y, Y') are transformed to (X_m, X'_m) and (Y_m, Y'_m) as follows:

$$\begin{aligned} X_m &= X\cos(rx) - X'\sin(rx) + x2 \\ X'_m &= X\sin(rx) + X'\cos(rx) + xmrad2 \\ Y_m &= Y\cos(ry) - Y'\sin(ry) + y2 \\ Y'_m &= Y\sin(ry) + Y'\cos(ry) + ymrad2 \end{aligned}$$

The position (x, y) and the direction vector (u, v, w) of the source particle are determined from the following

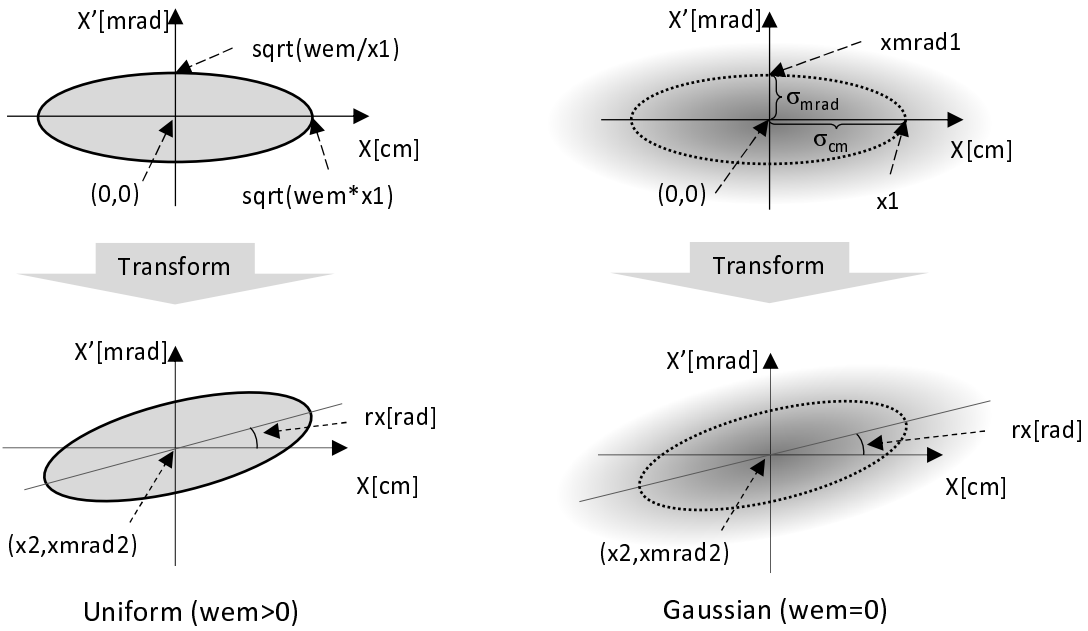


Figure 5.4: Sampling method from the phase space of x-coordinate, X, and angle, X'.

equations:

$$
\begin{aligned}
x &= X_m + x0 \\
y &= Y_m + y0 \\
u &= \frac{\tan\left(X'_m/1000\right)}{\sqrt{1.0 + \tan^2\left(X'_m/1000\right) + \tan^2\left(Y'_m/1000\right)}} \\
v &= \frac{\tan\left(Y'_m/1000\right)}{\sqrt{1.0 + \tan^2\left(X'_m/1000\right) + \tan^2\left(Y'_m/1000\right)}} \\
w &= \frac{1.0}{\sqrt{1.0 + \tan^2\left(X'_m/1000\right) + \tan^2\left(Y'_m/1000\right)}}
\end{aligned}
$$

When `wem=0`, it sets a two-dimensional Gaussian distribution that covers the elliptical area represented by the rms of a distance and the rms of an angle on the phase space diagram. At `wem>0`, source particles are generated in the range within the ellipse on the phase space. This source (`wem>0`) differs from the general beam distribution because it produces particles with a uniform distribution on the ellipse in the phase space. To reproduce the accelerator beam, you should generate a particle source of arbitrary emittance shape with `wem=0`.

5.3.11 s-type = 12

In this source type, decay-turtle output is read as source. Parameters for this source type are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

The input file is rewinded and re-used from the first particle again, if all of source in decay-turtle is read before the calculation finishes.

Table 5.49: Parameters for `s-type=12`

`s-type = 12`	Decay-turtle reading.
`x0 =`	(D=0.0) *x* coordinate offset of beam[cm].
`y0 =`	(D=0.0) *y* coordinate offset of beam[cm].
`z0 =`	(D=0.0) *z* coordinate offset of beam[cm].
`dir =`	(D=1) Direct cosine. (1 or −1 only).
`file =`	Decay-turtle filename (with full pathname).

The format of `decay-turtle` is double precision, and ascii, and each record is as

```
xp, xq, yp, yq, e0, wt0, pz0
```

Table 5.50: Decay-turtle data

variable	explanation
`xp, yp`	Incoming position of beam particle[cm].
`xq, yq`	Angle against vertical face with beam direction[mrad].
`e0`	Momentum of beam particle[GeV/c].
`wt0`	Weight of beam particle.
`pz0`	Polarizing of beam particle (be not in use).

5.3.12 Cone shape

Parameters for cone shape source are shown below. In this case, source particles are generated on the side surface of the cone. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.51: Parameters for cone shape

s-type = 18	Cone shape.
x0 =	(D=0.0) x coordinate offset of top of cone[cm].
y0 =	(D=0.0) y coordinate offset of top of cone[cm].
z0 =	(D=0.0) z coordinate offset of top of cone[cm].
x1 =	(D=0.0) x-component of vector from top to center of bottom circle[cm].
y1 =	(D=0.0) y-component of vector from top to center of bottom circle[cm].
z1 =	(D=0.0) z-component of vector from top to center of bottom circle[cm].
r0 =	(D=0.0) Length of conical generatrix[cm].
r1 =	(D=0.0) Length of the generatrix of the cone to be cut[cm]. When you want the sides of the cut cone to be the source area. The source is not generated between the vertex and the point at r1 from the vertex, and is generated between r1 and r0. Here, r1<r0 should be set.
r2 =	(D=0.0) Angle between generatrix and axis of cone[degree].
dir =	(D=1.0) Direction cosine of projectile against z axis. If all is set, source is isotropic. If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree]. $= -1$; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

5.3.13 Triangle prism shape

Parameters for triangle prism shape source are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.52: Parameters for triangle prism shape

s-type = 20	Triangle prism shape.
x0 =	(D=0.0) *x*-coordinate of origin apex of triangle[cm].
y0 =	(D=0.0) *y*-coordinate of origin apex of triangle[cm].
z0 =	(D=0.0) *z*-coordinate of origin apex of triangle[cm].
x1 =	(D=0.0) *x* vector from the origin to the first apex[cm].
y1 =	(D=0.0) *y* vector from the origin to the first apex[cm].
z1 =	(D=0.0) *z* vector from the origin to the first apex[cm].
x2 =	(D=0.0) *x* vector from the origin to the second apex[cm].
y2 =	(D=0.0) *y* vector from the origin to the second apex[cm].
z2 =	(D=0.0) *z* vector from the origin to the second apex[cm].
x3 =	(D=0.0) *x* vector from the origin to the third apex[cm].
y3 =	(D=0.0) *y* vector from the origin to the third apex[cm].
z3 =	(D=0.0) *z* vector from the origin to the third apex[cm].
exa =	(D=0.0) Attenuation coefficient. exa>0; $\exp(-ax)$, exa=0; uniform.
dir =	(D=1.0) Direction cosine of projectile against *z* axis.
	If all is set, source is isotropic.
	If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree].
	= −1 ; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

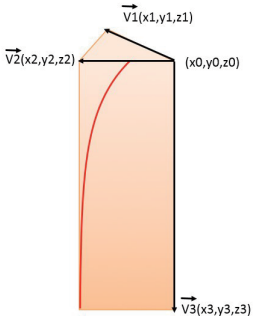


Figure 5.5: Coordinates of triangle prism

5.3.14 xyz-mesh distribution source

By specifying s-type=22, a complex spatial distribution of sources can be set. In stype=22, mono-energetic particles can be generated. The spatial distribution can be defined by x,y,z-type subsections and data of relative intensities $I_{ijk}, (i = 1, \cdots, n_x; j = 1, \cdots, n_y; k = 1, \cdots, n_z)$. The data should be written below the subsections. The direction of the sources can be specified by dir, phi, and dom. Note that the direction in each xyz-bin cannot be changed. Parameters for xyz-mesh distribution source are shown in Table 5.53. The order of parameters is free except for the data of the relative intensities. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.53: Parameters for xyz-mesh distribution source

s-type = 22	*xyz*-mesh distribution source.
mesh = xyz	Mesh type. Only xyz can be set.
x-type =	Mesh type for *x*-axis. x-type subsection is needed below this option.
y-type =	Mesh type for *y*-axis. y-type subsection is needed below this option.
z-type =	Mesh type for *z*-axis. z-type subsection is needed below this option.
	Put data of relative intensities I_{ijk} below x, y, z-type subsections. The change of the data element should be in the order of x, y, z. Note that the change in x, y, z axes should be ascending order if nx, ny, nz are positive, and it should be descending order if nx, ny, nz are negative.
dir =	(D=1.0) Direction cosine of projectile against *z* axis.
	If all is set, source is isotropic.
	If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree].
	$= -1$; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].
isbias =	(D=0) Represent the source distribution by changing the number of generated particles from each mesh.
	= 1 Represent the source distribution by changing the source weight of each mesh. The source location is determined in random from all meshes.
	= 2 Represent the source distribution by changing the source weight of each mesh. The equal numbers of sources are generated from each mesh; once a mesh is selected as the source-generation mesh, it will not be selected again until all meshes are selected. For the MPI mode, PHITS memorizes the source-generation meshes in each node, and thus, the same mesh might be selected from different nodes.

When all of nx,ny,nz are positive, the order of the relative intensities I_{ijk} is given as

$$I_{111}, I_{211}, \cdots, I_{n_x 11}, I_{121}, I_{221}, \cdots, I_{n_x n_y 1}, I_{112}, I_{212}, \cdots, I_{n_x n_y n_z}, \tag{3}$$

where $n_x =$|nx|, $n_y =$|ny|, $n_z =$|nz|. When nx and ny are positive and nz is negative, the order is given as

$$I_{11n_z}, I_{21n_z}, \cdots, I_{n_x 1n_z}, I_{12n_z}, I_{22n_z}, \cdots, I_{n_x n_y n_z}, I_{11n_z-1}, I_{21n_z-1}, \cdots, I_{n_x n_y 1}. \tag{4}$$

In the latter case, the data with $k = n_z =$|nz| should be first put.

An example using s-type=22 is shown below.

Example 1: Example of xyz-mesh distribution source

```
1:   [ Source ]
2:    s-type = 22
3:      proj = neutron
4:        e0 = 1.0
5:       dir = all
6:      mesh = xyz
```

```
 7:    x-type = 2
 8:        nx = 3
 9:      xmin = -10
10:      xmax =  10
11:    y-type = 2
12:        ny = -3
13:      ymin = -10
14:      ymax =  10
15:    z-type = 2
16:        nz = -2
17:      zmin = -10
18:      zmax =  10
19:        1 2 3
20:        4 5 6
21:        7 8 9
22:
23:        1 0 3
24:        0 5 0
25:        7 0 9
```

In this case, the source region is [-10cm$\leq x, y, z \leq$10cm], and the ranges in x, y, and z-axes are divided by 3, 3, and 2 bins, respectively. Because `ny` and `nz` are negative, the relative intensities I_{ijk} should be written in descending order for j and k. Therefore, in the 19th-21st lines of the example, I_{ijk} with $k = 2$, i.e. in the region of [0cm$\leq z \leq$10cm], are given as

$$\begin{pmatrix} I_{132} & I_{232} & I_{332} \\ I_{122} & I_{222} & I_{322} \\ I_{112} & I_{212} & I_{312} \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, \tag{5}$$

and I_{ijk} with $k = 1$, i.e. in the region of [$-$10cm$\leq z \leq$0cm], are given in the 23rd -25th lines as

$$\begin{pmatrix} I_{131} & I_{231} & I_{331} \\ I_{121} & I_{221} & I_{321} \\ I_{111} & I_{211} & I_{311} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 3 \\ 0 & 5 & 0 \\ 7 & 0 & 9 \end{pmatrix}. \tag{6}$$

Figure 5.6 shows the sources distribution of the example, which are obtained by `[t-product]` with `output=source`. The left and right panels are results in the regions of [0cm$\leq z \leq$10cm] and [$-$10cm$\leq z \leq$0cm], respectively. In the left panel, the regions of 1, 2, 3, 4, 5, 6, $\cdots$ correspond to $I_{132}, I_{232}, I_{332}, I_{122}, I_{222}, I_{322}, \cdots$, respectively. As the region number increases, the intensity in each region gradually increases. In the right panel, the sources don't generate in the regions of 2, 4, 6, and 8, which correspond to the elements of 0 in the right side of Eq. 6.

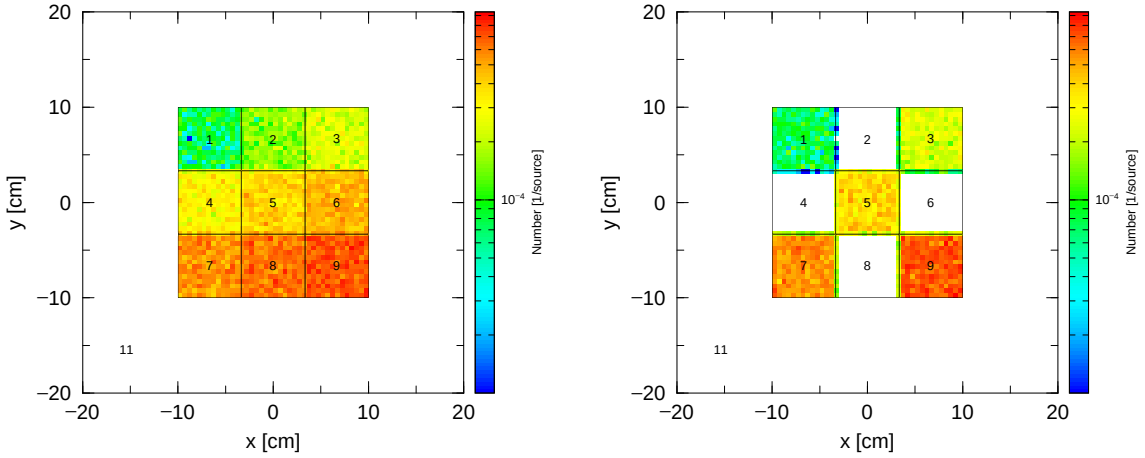


Figure 5.6: Results of example 1 in the region of [0cm$\leq z \leq$10cm] (left panel) and [$-$10cm$\leq z \leq$0cm] (right panel).

5.3.15 Tetra-mesh source

By specifying s-type=24, particle generation from tetrahedrons defined in tetrahedron geometry can be considered (sec. 5.6.5.4). Specification of the cell number of the universe where tetrahedrons belong is needed. Particles are produced uniformly from each tetrahedron which belongs to the specified universe with a rate proportional to its volume. Here only one cell number can be specified and therefore multi-source (sec. 5.3.1) have to be used to select more than one cell. The direction of the sources can be specified by dir, phi, and dom. Parameters for tetra-mesh source are shown in Table 5.54. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted. Please refer also ppt files and sample input files in \utility\TetraGEOM.

Table 5.54: Parameters for tetra-mesh source

s-type = 24	Tetra-mesh source.
tetreg =	The cell number of the universe where tetrahedrons belong. Only one cell number can be specified.
dir =	(D=1.0) Direction cosine of projectile against z axis. If all is set, source is isotropic. If data is set, an a-type subsection is necessary.
phi =	(D=none; random) Azimuthal angle[degree].
dom =	(D=0.0) Solid angle[degree]. $= -1$; $\cos^2$ bias distribution.
e0 =	(For mono-energy) Projectile energy[MeV/n].
e-type =	(For energy spectrum) Projectile energy[MeV/n].

An example using s-type=24 is shown below.

Example 2: Example of tetra-mesh source

```
1:   [ Source ]
2:    s-type = 24
3:      proj = proton
4:    tetreg = 201
5:       dir = all
6:        e0 = 70
```

In this case, source particles are produced uniformly from the region of the cell number 201 specified by tetreg. This specified cell number should belong the universe as the 13th line of the example 43.

5.3.16 Surface definition source

Parameters for the surface source are shown below. Direction specified by `dir` and `phi` is defined in the coordinate system that the direction of the *z*-axis is the same as that of the normal vector to the surface at the point of source particles. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.55: Parameters for surface definition source

`s-type = 26`	Surface source.
`suf =`	Surface number for generating sources defined in `[surface]` section. Only one surface can be defined for this parameter. The types of the surfaces that can be used for this parameter are `PX, PY, PZ, SO, S, SX, SY, SZ, C/X, C/Y, C/Z, CX, CY, CZ`.
`cut =`	Surface numbers to restrict the source generation area on the surface. The format is the same as `[cell]` section, but operators such as `( )` and `:` cannot be used. For example, if you define `cut = 1 -2 3 -4`, the source area is restricted to positive side (or outside) of surface 1 & 3, and negative side (or inside) of surface 2 & 4. Multiple planes can be defined, but only one sphere or cylinder can be defined. The area restricted by the specified surfaces should be a closed region, so it is necessary to define the minimum and maximum values using planes (`px,py,pz`) when the positive side (i.e., outer region) of a certain sphere (or cylinder) is specified.
`dir =`	(D=1.0) Direction cosine against the normal vector to the surface at the point of source particles. If `all` is set, source is isotropic. If `data` is set, an `a-type` subsection is necessary.
`phi =`	(D=all) Azimuthal angle[degree].
`dom =`	(D=0.0) Solid angle[degree]. $= -1$; $\cos^2$ bias distribution.
`e0 =`	(For mono-energy) Projectile energy[MeV/n].
`e-type =`	(For energy spectrum) Projectile energy[MeV/n].

Surfaces defined in `[surface]` section with the transform operator (`trcl`) should not be specified in `suf` and `cut` parameters. On the other hand, trcl can be used in `[source]` section even when `s-type=26`.

5.3.17 Reading dump file

By setting `s-type=17`, information on particles recorded in a file can be used as sources. The "dump" file can be obtained by using the dump option in the `[t-cross]`, `[t-product]`, and `[t-time]` tallies. There are two alternative methods for treating the dump data, which can be switched between using the `idmpmode` parameter: by setting `idmpmode=0`, each particle's data recorded in the dump file can be treated as an independent history; by setting `idmpmode=1`, the correlations between particles at the time of data dump can be taken into account via statistical processing based on history information recorded in the dump file. For continuous calculation using the dump file based on the history information from the previous step, the use of `idmpmode=1` is recommended. [76]

Parameters for `s-type=17` are shown below. The order of parameters is free. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.56: Parameters for dump file source (1)

`s-type = 17`	Reading dump file.
`file =`	Dump filename (with full pathname).
`dump =`	Number of dump data entries. If given as a negative number, the data should be written as ASCII. If `dump=0`, the dump data IDs will be read from the standard tally output file, and it is not necessary to provide the IDs in the following line.
(next line)	Dump data IDs.
`jpsf =`	Special option for Phase Space File. Beam starting point `z0` is necessary. =1: w/o weight, 2: w weight.
(optional)	If the following parameters are specified, these values have priority over the dump data. If the dump data do not include the following data, the relevant parameters should be specified.
`x0 =`	Minimum *x* coordinate[cm].
`x1 =`	Maximum *x* coordinate[cm].
`y0 =`	Minimum *y* coordinate[cm].
`y1 =`	Maximum *y* coordinate[cm].
`z0 =`	Minimum *z* coordinate[cm].
`z1 =`	Maximum *z* coordinate[cm].
`sx =`	(D=0) *x*-component of spin.
`sy =`	(D=0) *y*-component of spin.
`sz =`	(D=0) *z*-component of spin.
`dir =`	Direction cosine of projectile against *z* axis. If `all` is set, source is isotropic. If `data` is set, an `a-type` subsection is necessary.
`phi =`	(D=none; random) Azimuthal angle[degree].
`dom =`	(D=0.0) Solid angle[degree]. = −1 ; $\cos^2$ bias distribution.
`e0 =`	(For mono-energy) Projectile energy[MeV/n].
`e-type =`	(For energy spectrum) Projectile energy[MeV/n].
`wgt =`	(D=1.0) Weight of source particle.

[76] The option `idmpmode=1` and re-used calculation using the option `dmpmulti` were introduced by referring to the presentation "Estimation of uncertainty in multi-step Monte Carlo calculation" (N50) by Dr. Y. Namito *et al.* at the 2015 annual meeting of AESJ (Hitachi, Japan).

Table 5.57: Parameters for dump file source (2)

(optional)	The following parameters can also be specified.
factor =	(D=1.0) normalization of source particle.
t-type =	(D=0) time distribution.
reg =	(D=all) specify the region.
ntmax =	(D=1000) maximum re-try number when reg is specified.
trcl =	(D=none) transform number, or definition of transform.
idmpmode =	Control parameter to select dump source mode. (D=0) (nocas and nobach) not included in the case history information dump data. (D=1) (nocas and nobach) included in the case history information dump data. =0: Each particle data recorded in the dump file is treated as an independent history. =1: Particle data in the dump file are processed by taking account of particle correlations in the previous step.
dmpmulti =	Control parameter for reuse of dump file. (D=0.0) With idmpmode=0. (D=1.0) With idmpmode=1. =0.0: The dump file is rewound and re-used until total histories are finished, as specified by (maxcas*maxbch). However, this mode cannot be used with idmpmode=1. >0.0: The dump file is re-used the number of times specified by this parameter. The digits after the decimal point reflect the probability with which the Russian roulette routine will adopt another reproduction of the particle data.

To use idmpmode=1, the history number (nocas) and batch number (nobch) must be contained in the dump data. Using the PHITS tally dump option, a dump file named "***_dmp" is created in addition to the normal tally file with the file name "***" specified by file=***. Both files are required to use idmpmode=1 and must be placed in the same folder. The total number of histories of the previous step is adopted from the values of maxcas and maxbch written in this tally file and the history numbers, maxcas and maxbch, given in the PHITS input file are neglected. The normalization factor totfact is ignored when idmpmode=1. To use the normalization factor, totfact should be applied to the previous step for creating the dump file, at which point it will be reflected in the weights of particles recorded in the dump file. The option idmpmode=1 is incompatible with multi-source processing. In addition, the calculation should not be terminated by using stdcut parameter or editing batch.out.

The option dmpmulti controls the number of re-uses of the dump file. For instance, in the case of dmpmulti=2.0, the dump file will be reused twice, i.e., information for each particle recorded in the dump file is reproduced twice and the particle is treated as two particles having half-weights based on a different random number. The digits after the decimal point indicate the probability of creating another reproduction of the particle as stochastically determined by running the Russian roulette procedure for each particle. For example, in the case of dmpmulti=2.3, the particle data are reproduced twice with probability 70 % and three times with probability 30 %. As a special mode, setting dmpmulti=0.0 rewinds and re-uses the dump file until the total histories specified by (maxcas*maxbch) in the PHITS input file are finished. This mode is valid only with idmpmode=0.

The restart calculation (istdev<0) is essentially not applicable for dump source calculation; it only allowed with dmpmulti=0.0, in which case it should be noted that the calculation might give biased results unless the dump file contains sufficiently large particle data. Thus, re-calculation using a larger dmpmulti factor is recommended to achieve better statistics; if the particle data contained in the dump file are not statistically suffcient, the statistical uncertainties may not satisfactorily diminish even if a large dmpmulti factor is selected. In this case, it will be necessary to re-calculate from the previous step to create suffcient dump data.

The parameter dump specifies the number of dump data points per record. If this number is 0, the dump data IDs will be read from the standard tally output file, and it is not necessary to provide the IDs in the following line. If this number is positive, the data are read as binary data; setting dump as a negative value will cause the data to be read as ascii data. In the line following the dump= assignment, the data sequence of the record is input. The respective dump file physical quantities are encoded with the id numbers shown in Tables 5.58 and 5.59.

Table 5.58: Dump data id number (1)

physical quantities	kf	x	y	z	u	v	w	e	wt	time	c1	c2	c3	sx	sy	sz
id number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Table 5.59: Dump data id number (2)

physical quantities	name	nocas	nobch	no
id number	17	18	19	20

In the table, kf is the particle kf-code (see Table 4.4), x, y, and z are coordinates [cm], u, v, and w are the unit vectors of particle coordinate system, e is the energy ([MeV], or [MeV/n] for nuclei), wt is the weight, time is the initial time [ns], c1, c2, and c3 are the counter values, and sx, sy, and sz are the unit vectors of the respective spin directions. In Table 4.43, which encodes history-related ids, name is the particle's collision number, nocas is the batch history number, nobch is the batch number, and no is the cascade id contained in the history. These are assumed to in real*8 format for binary data and in *n*(1p1e24.15) data format for ascii data.

For an example, a record containing nine data points is given by

```
kf  e  wt  x  y  z  u  v  w
```

To read these data, the parameters are written as

```
dump = 9
1  8  9  2  3  4  5  6  7
```

5.3.18 User definition source

An original source function can be used by editing "usrsors.f," and setting `s-type=100`. If the following parameters are set, these values have the priority. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.60: Parameters can be specified in s-type=100

`s-type = 100`	User definition source.
	If below parameters are specified, these values have priority over the user defined data.
`x0 =`	Minimum *x* coordinate[cm].
`x1 =`	Maximum *x* coordinate[cm].
`y0 =`	Minimum *y* coordinate[cm].
`y1 =`	Maximum *y* coordinate[cm].
`z0 =`	Minimum *z* coordinate[cm].
`z1 =`	Maximum *z* coordinate[cm].
`sx =`	(D=0) *x*-component of spin.
`sy =`	(D=0) *y*-component of spin.
`sz =`	(D=0) *z*-component of spin.
`dir =`	Direction cosine of projectile against *z* axis.
	If `all` is set, source is isotropic.
	If `data` is set, an `a-type` subsection is necessary.
`phi =`	(D=none; random) Azimuthal angle[degree].
`dom =`	(D=0.0) Solid angle[degree].
	$= -1$; $\cos^2$ bias distribution.
`e0 =`	(For mono-energy) Projectile energy[MeV/n].
`e-type =`	(For energy spectrum) Projectile energy[MeV/n].
`wgt =`	(D=1.0) Weight of source particle.
`factor =`	(D=1.0) Normalization of source particle.
`t-type =`	(D=0) Time distribution.
`reg =`	(D=all) Specify the region.
`ntmax =`	(D=1000) Maximum re-try number when `reg` is specified.
`trcl =`	(D=none) Transform number, or definition of transform.

User-defined variables `c1-c99` by setting `set` can be used in "usrsors.f" as `cval(1-99)`. Note that the last setting of `set` is available if the same variable is defined several times.

A sample program of "usrsors.f" is shown as following. In the first comment part, there is a list of the variables which is necessary to define the source. Next there is a list of kf-code which specifies the source particle. In the last part of the comment, the random number functions, one is a uniform random number, the other is a Gaussian random number, are shown. The first part of the program is an example of the initialization, which describes the open and close the data file. The remaining part shows a list of the variables which user should define in this subroutine.

File 1: usrsors.f

```
      ************************************************************************
            subroutine usrsors(x,y,z,u,v,w,e,wt,time,name,kf,nc1,nc2,nc3,
           &                   sx,sy,sz)
      *        sample subroutine for user defined source.                    *
      *        variables :                                                   *
      *           x, y, z : position of the source.                          *
      *           u, v, w : unit vector of the particle direction.           *
      *           e       : kinetic energy of particle (MeV).                *
      *           wt      : weight of particle.                              *
      *           time    : initial time of particle. (ns)                   *
      *           name    : usually = 1, for Coulmb spread.                  *
      *           kf      : kf code of the particle.                         *
      *           nc1     : initial value of counter 1                       *
      *           nc2     : initial value of counter 2                       *
      *           nc3     : initial value of counter 3                       *
      *           sx,sy,sz : spin components                                 *
      *-----------------------------------------------------------------------*
      *        kf code table                                                 *
      *           kf-code: ityp :  description                               *
      *             2212 :   1  :  proton                                    *
      *             2112 :   2  :  neutron                                   *
      *              211 :   3  :  pion (+)                                  *
      *              111 :   4  :  pion (0)                                  *
      *             -211 :   5  :  pion (-)                                  *
      *              -13 :   6  :  muon (+)                                  *
      *               13 :   7  :  muon (-)                                  *
      *              321 :   8  :  kaon (+)                                  *
      *              311 :   9  :  kaon (0)                                  *
      *             -321 :  10  :  kaon (-)                                  *
      *           kf-code of the other transport particles                   *
      *               12 :         nu_e                                      *
      *               14 :         nu_mu                                     *
      *              221 :         eta                                       *
      *              331 :         eta'                                      *
      *             -311 :         k0bar                                     *
      *            -2112 :         nbar                                      *
      *            -2212 :         pbar                                      *
      *             3122 :         Lanbda0                                   *
      *             3222 :         Sigma+                                    *
      *             3212 :         Sigma0                                    *
      *             3112 :         Sigma-                                    *
      *             3322 :         Xi0                                       *
      *             3312 :         Xi-                                       *
      *             3334 :         Omega-                                    *
      *-----------------------------------------------------------------------*
      *        available function for random number                          *
      *           unirn(dummy) : uniform random number from 0 to 1           *
      *           gaurn(dummy) : gaussian random number                      *
      *                          for exp( - x**2 / 2 / sig**2 ) : sig = 1.0  *
      ************************************************************************
            implicit real*8 (a-h,o-z)
      *-----------------------------------------------------------------------
            parameter ( pi  = 3.141592653589793d0 )
```

```
54:         data ifirst / 0 /
55:         save ifirst
56:         character filenm*50
57:   *-----------------------------------------------------------------------
58:   *        example of initialization
59:   *-----------------------------------------------------------------------
60:           if( ifirst .eq. 0 ) then
61:   c             filenm = 'input.dat'
62:   c             inquire( file = filenm, exist = exex )
63:   c             if( exex .eqv. .false. ) then
64:   c                write(*,*) 'file does not exist =>  ', filenm
65:   c                call parastop( 887 )
66:   c                end if
67:   c             open(71, file = file(i), status = 'old' )
68:
69:   c             close(71)
70:                 ifirst = 1
71:           end if
72:   *-----------------------------------------------------------------------
73:   *        example for 3 GeV proton with z-direction
74:   *-----------------------------------------------------------------------
75:              x = 0.0
76:              y = 0.0
77:              z = 0.0
78:              u = 0.0
79:              v = 0.0
80:              w = 1.0
81:              e = 3000.0
82:             wt = 1.0
83:           time = 0.0
84:           name = 1
85:             kf = 2212
86:            nc1 = 0
87:            nc2 = 0
88:            nc3 = 0
89:             sx = 0.d0
90:             sy = 0.d0
91:             sz = 0.d0
92:   *-----------------------------------------------------------------------
93:         return
94:         end
```

5.3.19 Definitions for energy distribution

The energy-distributed source can be defined by `e-type` subsection instead of by `e0`. Note that `e0` have to be deleted or set to invalid by comment marks when the `e-type` subsection is given. Attention should also be paid to the unit of different kinds of source particles. Throughout this section, the unit is [/(MeV/n)] for ion sources (d, t, α, and nuclei), and [/MeV] for other types of sources.

Available types of the source energy distributions are shown in Table 5.61. There are integral and differential types as the format of the distribution; for the integral type, the source intensity should be given as an energy-integrated value, while for the differential type, the intensity should be given as the energy differential value expressed in [particles/MeV] (particles/MeV/n for ions). As generation method of the source particles, two methods can be selected. In the first one, the number of particles generated in a bin is proportional to the intensity, and the specified energy distribution is statistically described. In the second one, the number of source particles generated in each bin is the same for all energy bins, but the weight values of source particles are adjusted to be proportional to the intensity.

Table 5.61: Types of source energy distribution. There are two methods for generation of source particles; one is to adjust statistically the number of particles in each energy bin with the constant weight value, while the other is to change the weight value of the source particle in each bin with the constant number of particles. When setting the numbers in the brackets to `e-type`, the energy can be given by wave length (Å).

Type	Generation method	`e-type`	Explanation
Continuous energy distribution with integral type	Number of particles Weight value	1, (11) 4, (14)	A continuous energy distribution can be specified by providing the data set of lower limits of energy bins and integrated values of the particle generation probabilities in each bin.
Continuous energy distribution with differential type	Number of particles Weight value	21, (31) 24, (34)	A continuous energy distribution can be specified by providing the data set of the minimum of energy bins and differential probabilities of the particle generation in each bin.
Discrete energy distribution	Number of particles Weight value	8, (18) 9, (19)	A discrete energy distribution can be specified by giving data set of energy points and probabilities of the particle generation at each points.
Discrete+Continuous energy distribution with integral type	Number of particles Weight value	22, (32) 23, (33)	Any energy distribution can be specified by providing the data set of the minimum and maximum of energy bins, and integrated values of particle probabilities in each bin.
Gaussian distribution with differential type	Number of particles	2, (12)	A differential spectrum of Gaussian distribution can be specified by giving a mean value and FWHM of Gaussian.
Maxwellian distribution with differential type	Number of particles Weight value	3 7	A differential spectrum of Maxwellian distribution can be specified by giving a temperature parameter of Maxwellian.
Analytical function with differential type	Number of particles Weight value	5, (15) 6, (16)	A differential spectrum can be specified by giving analytical functions.
Energy distribution for RI source	Number of particles Weight value	28 29	The α, β, γ-rays, and fission neutrons of radioisotope (RI) decay are generated by simply specifying the activity and names of the RIs.
Energy distribution of tally results	Number of particles	20	A energy distribution can be specified by using results of tallies.
Cosmic-ray source mode	Number of particles Weight value	25 26	The cosmic-ray energy and angular distributions in space and in the atmosphere are reproduced considering their absolute values.

Input formats and parameters in each `e-type` will be explained below. If a parameter has a default value (D=***), the parameter are optional.

5.3.19.1 Continuous energy distribution with integral value

Table 5.62: Parameters for source energy distribution (1)

parameter	explanation
e-type = 1, (11)	Any energy distribution can be specified by providing the data set of energy bins e(i) and the integrated values of the particle generation probabilities w(i) by hand. The number of particles generated in a bin is proportional to w(i), and the specified energy distribution is statistically described. For case 11, the energy is given by the wave length (Å).
ne =	Number of energy groups. If this is a positive number, source particles are generated so that the energy differential fluxes in units of [1/MeV] become constant in each bin. If negative, the fluxes in units of [1/Lethargy] become constant for each bin. Data must be provided in the following line using the format (e(i), w(i), i=1, ne), e(ne+1). The integrated numbers of particles generated in each energy bin is proportional to w(i).
e-type = 4, (14)	The same energy distribution as in the case of e-type = 1, (11) can be specified, except with the energy bins e(i) and weights of the source particle w(i) entered by hand. The number of source particles generated in each bin is the same for all energy bins, but integrated values of the weights of source particles are adjusted to be proportional to w(i). The number of source particles generated in each bin can also be changed by specifying p(i). For case 14, the energy is given by wave length (Å).
ne =	Number of energy groups. If this is a positive number, source particles are generated so that the energy differential fluxes in units of [1/MeV] become constant in each bin. If negative, the fluxes in units of [1/Lethargy] become constant for each bin. Data must be provided in the following line using the format (e(i), w(i), i=1, ne), e(ne+1). In the default (p-type = 0), equal numbers of particles are generated in each cell. The integrated number of source particles generated in each bin is proportional to p(i).
p-type = 0, 1	(D = 0) generation option. For 0, p(i) = 1 for all i is assumed without the following data. For 1, p(i) must be given from the next line using the format (p(i),i=1,ne).

An alternative option for neutron optics has been developed to specify the energy as a wavelength: if e-type is set to = 11, 12, 14, the wavelength (Å) is used as the energy unit. In the other cases, the expression e0 = 8.180425e-8/13**2 is used, which corresponds to the energy of a neutron with a 13Å wavelength.

In the case of `e-type=1`, the format is as follows.

```
e-type = 1
    ne = n
       e(1)   w(1)
       e(2)   w(2)
       e(3)   w(3)
       ....   ...
       e(n-1) w(n-1)
       e(n)   w(n)
       e(n+1)
```

In this case, the energy distribution is given as

```
e(1)-e(2)   w(1)
e(2)-e(3)   w(2)
e(3)-e(4)   w(3)
.......     ...
e(n-1)-e(n) w(n-1)
e(n)-e(n+1) w(n)
```

As an example, use the following code to set the strengths of three energy bins between 0-2 MeV, 2-4 MeV, and 4-6 MeV as 0.2, 0.6, and 0.2, respectively:

```
e-type = 1
    ne = 3
       0   0.2
       2   0.6
       4   0.2
       6
```

5.3.19.2 Continuous energy distribution with differential value

Table 5.63: Parameters for source energy distribution (2)

parameter	explanation
e-type = 21, (31)	Any energy distribution can be specified by giving data set of energy bins e(i) and differential probabilities of the particle generation dφ/dE(i) by hand. The integrated number of the particle generation in the bin is proportional to dφ/dE(i)*{e(i+1)-e(i)}, and the specified energy distribution is statistically described. For 31 case, energy is given by wave length (Å).
ne =	Number of energy group. If it is given by positive number, source particles are generated so that the energy differential fluxes in the unit of [1/MeV] become constant in each bin. On the other hand, if ne is negative, the fluxes in the unit of [1/Lethargy] become constant in each bin. Data must be given from the next line by the format as (e(i),dφ/dE(i),i=1,ne), e(ne+1). The integrated number of the particle generation in the each energy bin is proportional to dφ/dE(i)*{e(i+1)-e(i)}.
e-type = 24, (34)	The same energy distribution as in the case of e-type=21,(31) can be specified. Unlike e-type=21,(31), the distribution is described by giving data set of energy bins e(i) and weights of the source particle w(i) by hand. The number of source particles generated in each bin is the same for all energy bin, but integrated values of the weight of source particles are adjusted to be proportional to w(i)*{e(i+1)-e(i)}. The number of source particles generated in each bin can also be changed by specifying p(i). For 34 case, energy is given by wave length (Å).
ne =	Number of energy group. If it is given by positive number, source particles are generated so that the energy differential fluxes in the unit of [1/MeV] become constant in each bin. On the other hand, if ne is negative, the fluxes in the unit of [1/Lethargy] become constant in each bin. Data must be given from the next line by the format as (e(i),w(i),i=1,ne), e(ne+1). In default (p-type=0), equal number of particle is generated in each cell. The integrated number of source particles generated in each bin is proportional to p(i).
p-type = 0, 1	(D=0) generation option. For 0, p(i)=1 for all i is assumed without the following data. For 1, p(i) must be given from the next line by the format as (p(i),i=1,ne).

5.3.19.3 Discrete energy distribution

Table 5.64: Parameters for source energy distribution (3)

parameter	explanation
`e-type = 8, (18)`	Any energy distribution can be specified by giving data set of energy points `e(i)` and probabilities of the particle generation `w(i)` by hand. The number of the particle generation at the point is proportional to `w(i)`, and the specified energy distribution is statistically described. For 18 case, energy is given by wave length (Å).
`ne =`	Number of energy points. Data must be given from the next line by the format as `(e(i),w(i),i=1,ne)`. The number of the particle generation at the each energy point is proportional to `w(i)`.
`e-type = 9, (19)`	The same energy distribution as in the case of `e-type=8,(18)` can be specified. Unlike `e-type=8,(18)`, the distribution is described by giving data set of energy points `e(i)` and weights of the source particle `w(i)` by hand. The number of source particles generated in each point is the same for all energy point, but integrated values of the weight of source particles are adjusted to be proportional to `w(i)`. The number of source particles generated in each bin can also be changed by specifying `p(i)`. For 19 case, energy is given by wave length (Å).
`ne =`	Number of energy points. Data must be given from the next line by the format below. In default (`p-type=0`), equal number of particle is generated at each point. `(e(i),w(i),i=1,ne)` The number of source particles generated in each point is proportional to `p(i)`.
`p-type = 0, 1`	(D=0) generation option. For 0, `p(i)=1` for all `i` is assumed without the following data. For 1, `p(i)` must be given from the next line by the format as `(p(i),i=1,ne)`.

5.3.19.4 Energy distribution by free format

Table 5.65: Parameters for source energy distribution (4)

parameter	explanation
e-type = 22, (32)	Any energy distribution can be specified by providing the data set of the minimum emin(i) and maximum emax(i) of energy bins, and the integrated values of the particle probabilities w(i) by hand. To define the same discrete distribution as e-type=18, (28), set the maximum to the same value as the minimum. The number of particles generated in a bin is proportional to w(i), and the specified energy distribution is statistically described. For case 32, the energy is given by the wave length (Å).
ne =	Number of energy groups. If this is a positive number, source particles are generated so that the energy differential fluxes in units of [1/MeV] become constant in each bin. If negative, the fluxes in units of [1/Lethargy] become constant for each bin. Data must be provided in the following line using the format: (emin(i),emax(i),w(i),i=1,|ne|). The integrated numbers of particles generated in each energy bin is proportional to w(i).
e-type = 23, (33)	The same energy distribution as in the case of e-type=22,(32) can be specified, except with the weights of the source particle w(i) entered by hand. The number of source particles generated in each bin is the same for all energy bins, but integrated values of the weights of source particles are adjusted to be proportional to w(i). The number of source particles generated in each bin can also be changed by specifying p(i). For case 33, the energy is given by wave length (Å).
ne =	Number of energy groups. If this is a positive number, source particles are generated so that the energy differential fluxes in units of [1/MeV] become constant in each bin. If negative, the fluxes in units of [1/Lethargy] become constant for each bin. Data must be provided in the following line using the format: (emin(i),emax(i),w(i),i=1,|ne|). In the default (p-type=0), equal numbers of particles are generated in each cell. The integrated number of source particles generated in each bin is proportional to p(i).
p-type = 0, 1	(D=0) generation option. For 0, p(i)=1 for all i is assumed without the following data. For 1, p(i) must be given from the next line by the format as (p(i),i=1,ne).

An alternative option for neutron optics has been developed to specify the energy as a wavelength: if e-type is set to = 32, 33, the wavelength (Å) is used as the energy unit. In the other cases, the expression e0 = 8.180425e-8/13**2 is used, which corresponds to the energy of a neutron with a 13Åwavelength.

In the case of `e-type = 22`, the format is as follows:

```
e-type = 22
    ne = n
       emin(1) emax(1)   w(1)
       emin(2) emax(2)   w(2)
       emin(3) emax(3)   w(3)
       ....   ...
       emin(n-1) emax(n-1) w(n-1)
       emin(n) emax(n)   w(n)
```

As an example, use the following code to set the strengths of three energy bins between 0-2 MeV, 2-4 MeV, and 4-6 MeV as 0.2, 0.6, and 0.2, respectively, and the discrete one of mono-energy of 5.6 MeV as 0.4:

```
e-type = 22
    ne = 4
       0   2   0.2
       2   4   0.6
       4   6   0.2
       5.6 5.6 0.4
```

5.3.19.5 Gaussian and Maxwellian energy distributions

Table 5.66: Parameters for source energy distribution (5)

parameter	explanation
e-type = 2, (12)	Differential spectrum dφ/dE(i) is given by Gaussian distribution. For 12 case, energy is given by wave length (Å).
eg0 = eg1 = eg2 = eg3 =	center of Gaussian distribution [MeV/n]. FWHM of Gaussian distribution [MeV/n]. minimum cut off for Gaussian distribution [MeV/n]. maximum cut off for Gaussian distribution [MeV/n].
e-type = 3	Differential spectrum dφ/dE(i) is given by Maxwellian distribution: $f(E) = E^a \exp(-E/T)$
nm =	(D=-200) number of energy group. If it is given by positive number, linear interpolation is assumed in a bin. If negative, logarithmic interpolation is assumed in a bin.
et0 = et1 = et2 = et3 =	Temperature parameter T [MeV] given by $T = kt$. Here, Boltzmann constant is $k = 8.617 \times 10^{-11}$[MeV/K] and temperature t is in units of [K]. minimum cut off for Maxwellian distribution [MeV/n]. maximum cut off for Maxwellian distribution [MeV/n]. (D=0.5): Power index of energy, i.e. parameter a in the equation above.
e-type = 7	The same energy distribution as in the case of e-type=3 can be specified. Unlike e-type=3, the number of source particles generated in each bin is the same for all energy bin, but integrated values of the weight of source particles are adjusted to be proportional to $f(E) = E^a \exp(-E/T)$. The number of source particles generated in each bin can also be changed by specifying p(i).
nm =	(D=-200) Number of energy group. If it is given by positive number, linear interpolation is assumed in a bin. If negative, logarithmic interpolation is assumed in a bin. In the default (p-type=0), equal numbers of particles are generated in each cell. The integrated number of source particles generated in each bin is proportional to p(i).
et0 = et1 = et2 = et3 =	temperature parameter T [MeV]. minimum cut off for Maxwellian distribution [MeV/n]. maximum cut off for Maxwellian distribution [MeV/n]. (D=0.5): Power index of energy, i.e. parameter a in the equation above.
p-type = 0, 1	(D=0) generation option. For 0, p(i)=1 for all i is assumed without the following data. For 1, p(i) must be given from the next line by the format as (p(i),i=1,nm).

The energy distribution of Maxwellian with $a = 0.5$, $f(E) = \sqrt{E}\exp(-E/T)$, is related with the usual velocity distribution $f(v)$; $f(E)dE \propto f(v)dv$. Using $f(v) = Av^2 \exp(-mv^2/kt)$ and $mvdv = dE$, which is given by $mv^2/2 = E$, we obtain $f(v)dv = A(E/m)\exp(-E/T)\cdot(1/m\sqrt{E})dE = (A/m^2)\sqrt{E}\exp(-E/T)$. Here, $T = kt$.

5.3.19.6 Energy distribution defined by analytical functions

Table 5.67: Parameters for source energy distribution (6)

parameter	explanation
`e-type = 5, (15)`	Differential spectrum `dφ/dE(i)` is given by $f(x)$. For 15 case, energy is given by wave length (Å).
`f(x) =`	Any analytical function of x, Fortran style. x denotes energy [MeV/n]. One can use intrinsic functions and constants `C`, e.g., `f(x) = exp(-c1*x**2)`.
`nm =`	number of energy group. If it is given by positive number, linear interpolation is assumed in a bin. If negative, logarithmic interpolation is assumed in a bin. Integrated number of source particles generated in each cell is proportional to $f(x)$.
`eg1 =` `eg2 =`	minimum cut off for energy distribution [MeV/n]. maximum cut off for energy distribution [MeV/n].
`e-type = 6, (16)`	The same energy distribution as in the case of `e-type=5,(15)` can be specified. Unlike `e-type=5,(15)`, the number of source particle generated in each bin is the same for all energy bin, but integrated values of the weight of source particles are adjusted to be proportional to $f(x)$. The number of source particles generated in each bin can also be changed by specifying `p(i)`. For 16 case, energy is given by wave length (Å).
`f(x) =`	Any analytical function of x, Fortran style. x denotes energy (MeV/n). One can use intrinsic functions and constants `C`, e.g., `f(x) = exp(-c1*x**2)`.
`nm =`	number of energy group. If it is given by positive number, linear interpolation is assumed in a bin. If negative, logarithmic interpolation is assumed in a bin. In the default (`p-type=0`), equal numbers of particles are generated in each cell. The integrated number of source particles generated in each bin is proportional to `p(i)`.
`eg1 =` `eg2 =`	minimum cut off for energy distribution [MeV/n]. maximum cut off for energy distribution [MeV/n].
`p-type = 0, 1`	(D=0) generation option. For 0, `p(i)=1` for all `i` is assumed without the following data. For 1, `p(i)` must be given from the next line by the format as `(p(i),i=1,nm)`.

5.3.19.7 Energy distribution of RI source

Table 5.68: Parameters for source energy distribution (7)

parameter	explanation
e-type= 28,29	The α, β (including Auger electrons), and γ-rays of radioisotope (RI) decay are generated by simply specifying the activity (in Bq) and names of the RIs. All kinds of radiation from decay (i.e., α-, β-, γ-rays, and fission neutrons) are generated when proj=all is set. The DECDC[77]nuclear decay database is used to obtain the energy spectra. This database is equivalent to ICRP Publication 107 and contains decay databases for almost all radionuclides (including metastables) with half-lives of more than one minute. To use this function, the directory containing the DECDC data file "RIsource.dat" must be specified by setting file(24) (D=c:\phits\data\) in the [parameters] section. When e-type=28, spectra are expressed by changing the probabilities of generating particle energy spectra obtained from DECDC. When e-type=29, spectra are expressed by changing the weights of the source particles while maintaining constant probabilities for all source energy.
ni=	Number of RIs. RI names and activity must be given in the next line using the format (RI(i),A(i),i=1, ni). RI(i) can be defined using one of two formats, e.g., as 137Cs or Cs-137. A(i) is specified in units of Bq.
dtime=	(D=−10.0) Option for time evolution [s]. dtime>0 (in s): The energy spectrum is determined based on the decays of specified RIs including contributions from their daughter nuclides at the time at which dtime has passed; this changes the activity of each RI from its specified value. For example, setting 100 Bq for an RI with a half-life of 1 min and dtime=60 results in an RI activity of 50 Bq in the PHITS calculation. dtime=0: The energy spectrum is determined based on the decays of specified RIs. The γ-rays emitted from metastables of their daughter nuclides with a half-life of more than 1 minute are not included, because no time evolution is considered. dtime<0: The energy spectrum is determined based on the decays of the specified RIs including the contributions from their daughter nuclides at the time at which the half-life×|dtime| has passed. Unlike the case of dtime>0, the activities of the respective RI are unchanged from their specified values. For example, setting 100 Bq for an RI with a half-life of 1 min and dtime=−1.0 results in an unchanged RI activity of 100 Bq in the PHITS calculation. When no information on the time necessary to attain radioactive equilibrium is present, a large negative value, e.g., dtime=−10.0, must be set. It is not possible to specify both parent and daughter nuclides simultaneously for dtime<0.

To use this function with multi-sources, set <source>=1.0 and totfact as follows:

- when all sources are defined by e-type=28,29, set totfact as the number of <source>;
- when the multi-sources include sources defined by e-type≠28,29, set each <source> as the absolute intensity of each RI and set totfact as the sum of all values of <source>.

If the totfact is given as a negative value, the same particle is generated in each multi-source section with weights adjusted according to the ratio of the activities.

[77] A. Endo, Y. Yamaguchi and K.F. Eckerman, Nuclear decay data for dosimetry calculation - Revised data of ICRP Publication 38, JAERI 1347 (2005).

Table 5.69: Parameters for source energy distribution (8)

parameter	explanation
	(continued in the cases of `e-type= 28,29`)
`actlow=`	(D=1.0e-10) Option for lower limit of the activity [Bq]. If the activity following the passage of the specified time is smaller than `actlow`, the decay of the corresponding RI is not included in the sources.
`norm=`	(D=0) Option for normalization of tally results. 0: Tally results are normalized in units of [/s]. In this case, the source intensity is determined by the specified activity (if `dtime>0`, its decrease owing to decay is also taken into account). 1: Tally results are normalized in units of [/source]. This is equivalent to the general normalization of PHITS.
`iaugers=`	(D=0) Option for the production of Auger & internal conversion (IC) electrons. (only for `proj=electron`) 0: All electrons (β-rays, Auger & IC electrons) are generated. 1: Only β-rays are generated. 2: Only Auger & IC electrons are generated.
`iannih=`	(D=1) Option for the production of annihilation photons (only for `proj = photon`) 0: Consider the production of annihilation photons. 1: Ignore the production of annihilation photons. This option should be selected when you explicitly consider the production of positrons by defining another <source> with `proj = positron`, otherwise annihilation photons will be counted twice.
`icharctx=`	(D=0) Option for the production of characteristic X-rays. (only for `proj=photon`) 0: All photons (γ-rays and characteristic X-rays) are generated. 1: Only γ-rays are generated. 2: Only characteristic X-rays are generated.

The following is an example using `e-type = 28,29`:

```
[ Source ]
 totfact = 2.0
 <source> = 1.0
   s-type =   1
     proj =  photon
      dir =   all
       r0 =   0.
       z0 =   0.
       z1 =   0.
   e-type =   28
       ni =   1
            Cs-137  100.
    dtime =  -10.0
   actlow =   1.0
 <source> = 1.0
   s-type =   1
     proj =  photon
      dir =   all
       r0 =   0.
       z0 =   0.
       z1 =   0.
   e-type =   28
       ni =   1
            Cs-134  100.
    dtime =  -10.0
   actlow =   1.0
```

In this case, Cs-137 and Cs-134 after reaching radioactive equilibrium at 100 Bq are both defined as photon sources. Furthermore, setting `actlow=1` ensures that activity smaller than 1 Bq is ignored.

Furthermore, for example, to consider both γ and β-rays by decay of Cs-137, make a multi-sources having two <source> sections with `proj=photon` and `proj=electron`. Because both the production rates of the γ and β-rays are the same, their ratios can be specified as 1.0 (<source>=`1.0`). In this case, `totfact` should be 2.0, which is the sum of the <source> sections.

5.3.19.8 Energy distribution of result of tally

Table 5.70: Parameters for source energy distribution (9)

parameter	explanation
e-type= 20	Results of tallies can be used as an energy distribution by specifying file. Only results of [t-track], [t-cross], [t-point], [t-product], [t-time], and [t-interact] tallies with axis=eng can be used. A file name (***.out), which was specified in the tally section as an output file name, should be set in file. There are two kinds of tally results, energy differential and integrated values, in accordance with unit given in the tally section. The energy distribution of the source particle is expressed on the basis of the kind. Note that time (nsec) and angular (sr) derivatives are not taken into account. The distribution of the source is normalized to the integrated value of the tally result. In the default setting, the unit of energy in tally results is MeV, while that in [source] section is MeV/n. To overcome this inconsistency, iMeVperU should be set to 1 in [parameters] of the 1st simulation in order to output the tally results in MeV/n in the cases of heavy ions.
file=	Output file name of tallies.

The first result specified by part in the tally section is used as the distribution of the source particle. Note that particles of its kind are not generated. The parameter proj should be set to specify the kind of the source particle.

An example using e-type=20 is as follows. This function can be used by setting only e-type=20 and file.

```
[ Source ]
   s-type =   1        # axial source with energy spectrum
     proj =  neutron   # kind of incident particle
      dir =   1.0      # z-direction of beam [cosine]
       r0 =   0.       # radius [cm]
       z0 =   0.       # minimum position of z-axis [cm]
       z1 =   0.       # maximum position of z-axis [cm]
   e-type =   20       # energy distribution given by tally output
     file = cross.out  # file name of tally output
```

5.3.19.9 Cosmic-ray source mode

Table 5.71: Parameters for source energy distribution (10)

parameter	explanation
e-type= 25,26	These types reproduce the cosmic-ray energy and angular distributions in space and in the atmosphere, considering their absolute values. When e-type=28, the numbers of particles generated in each bin are proportional to the calculated cosmic-ray spectra, while their weights are constant. When e-type=29, the weights of particles generated in each bin are proportional to the calculated cosmic-ray spectra, while maintaining constant probabilities for all source energy bins. In the cosmic-ray source mode, <source> of each multi-source should be 1.0. If totfact is set to the source area, e.g. $\pi(r_2)^2$ for s-type=9, the tally results are automatically normalized to (/sec). If totfact is given as a negative value, the same particle is generated in each multi-source section with weights adjusted according to the ratio of the cosmic-ray fluxes. Note that e-type as well as icenv should be same for all multi-sources in the case of at least one multi-source having e-type=25 or 26.

Table 5.72: Parameters for source energy distribution (11)

parameter	explanation
	(continued in the cases of `e-type= 25,26`)
`icenv=`	(D=0) Cosmic-ray environment parameter. 1–5: Terrestrial galactic cosmic-ray (GCR) mode based on PARMA/EXPACS[78]. Each `icenv` represents the cosmic-ray environments in the ideal atmosphere (=1), on the ground surface (=2), at the pilot and cabin locations in aircraft (=3 & 4), and the black-hole mode (=5). In the cases of `icenv`=2–4, the local environment must be specified by setting the `environ` parameter. The blackhole mode (=5) is useful for the simulations with ground because all albedo neutrons from the ground are excluded in the mode. Please see "phits/sample/source/Cosmicray/GCR-blackhole/readme.txt" in more detail. The type of projectile (`proj`) should be neutron, proton, ions with $Z \leq 28$ (Ni), muons, photon, electron, and positron in this mode. Note that the results obtained from `icenv`=1–5 are the same except for `proj`= neutron because the local environments influence only on the neutron fluxes. Energy-dependent angular distributions determined by PARMA/EXPACS can be also reproduced in this mode in the case of `s-type=9` and `dir=iso`, where the top of the atmosphere should be z+ direction. 0: GCR-in-space mode based on the DLR model[79]. The type of projectile (`proj`) should be proton and ions with $Z \leq 28$ (Ni) in this mode. The influence of the Earth's magnetosphere[80] on typical Low-Earth-Orbits (LEO)[81] can be considered when `alti` is specified. -1: Free-space solar energetic particle (SEP) mode based on the Tylka model[82]. Currently, the time-integral SEP fluences for 4 historically large solar particle events (SPE), which were occurred in Feb. 1956, Nov. 1960, Aug. 1972, and Oct. 1989, can be reproduced. The type of projectile (`proj`) should be proton in this mode. If `totfact` is set to the source area, the tally results are automatically normalized to (/SPE) instead of (/sec). -2: Trapped proton (TP) on typical LEO[81] mode based on the AP-8 model[83] implemented in SPENVIS[84]. The type of projectile (proj) should be proton in this mode.

[78] T. Sato, Analytical Model for Estimating Terrestrial Cosmic Ray Fluxes Nearly Anytime and Anywhere in the World: Extension of PARMA/EXPACS, PLOS ONE 10(12): e0144679 (2015) DOI: 10.1371/journal.pone.0144679

T. Sato, Analytical Model for Estimating the Zenith Angle Dependence of Terrestrial Cosmic Ray Fluxes. PLOS ONE 11(8): e0160390 (2016). DOI: 10.1371/journal.pone.0160390

[79] Matthia D, Berger T, Mrigakshi AI, Reitz G, A ready-to-use galactic cosmic ray model. Adv Space Res 51: 329-338 (2013).

[80] An anti-proton trace model coupled with the static geomagnetic field model T89 (Tyganenko (1989), Planetary and Space Science, 37(1), 5-20) was used in the calculation of geomagnetic transmission function. Kp index was assumed to be 3. For more detail, please refer Sato et al. (2018) Radiat. Prot. Dosim. 180, 146-149.

[81] Orbits of International Space Station (ISS) with the inclination angle of 51.6 degree and the mean altitude range between 340 – 420 km are assumed.

[82] Tylka, A.J., Dietrich, W.F., A new and comprehensive analysis of proton spectra in ground-level enhanced (GLE) solar particle events, the 31st International Cosmic Ray Conference. Universal Academy Press, Poland (2009).

[83] Sawyer D, Vette J (1976) AP-8 trapped proton environment for solar maximum and solar minimum. National Space Science Data Center, Report 76-06, Greenbelt, Maryland.

[84] The Space Environment Information System (2010). `http://www.spenvis.oma.be/`

Table 5.73: Parameters for source energy distribution (12)

parameter	explanation
	(continued in the cases of e-type= 25,26)
nm=	(D=−200) number of energy group. If it is given by positive number, linear interpolation is assumed in a bin. If negative, logarithmic interpolation is assumed in a bin.
eg1=	Minimum cut off for energy distribution [MeV/n]. The default value is 1.0e-8 for proj=neutron. For other projectiles, the default values are 1.0e-2 and 1.0 for the terrestrial mode (icenv>0) and free-space mode (icenv≤0), respectively.
eg2=	Maximum cut off for energy distribution [MeV/n]. Default values are 1.0e4 for proj= neutron, photon, electron, positron, 1.0e8 for proj= muon+, muon-, 1.0e5 for the SEP mode, and 1.0e6 for other projectiles.
ag1=	(D=-1.0) The minimum zenith angle for calculating terrestrial cosmic-ray fluxes in $\cos\theta$.
ag2=	(D=1.0) The maximum zenith angle for calculating terrestrial cosmic-ray fluxes in $\cos\theta$. In the case of s-type=9, the zenith angles of actual source particles are also limited between ag1 and ag2. For other cases, ag1 and ag2 influence only on the energy spectrum, and the direction of the source particles are determined from dir, phi, and dom parameters. Please see "phits/sample/source/cosmicray/GCR-parallel" in more detail.
solarmod=	(D=0) Solar modulation potential, so-called W-index, used in the calculation of both terrestrial and free-space GCR fluxes. It is effective only for icenv≥0. The GCR fluxes are anti-correlated with the W-index whose values are approximately 0 and 150 for the solar minimum and maximum conditions, respectively. If this parameter is not specified, the W-index is automatically determined based on the daily count rates of several ground-level neutron monitors[85]on the date of interest specified by icyear, icmonth, and icday. The W-index of the default date (Oct. 20, 2020) was 0.1. If the specified date is before year 1951, i.e. before ground-level neutron monitors in operation, W-index is determined from the reconstructed cosmic-ray intensity based on the sun spot number by Usoskin et al.[86]W-index before year 1611 or during the Maunder Minimum (1647–1699) are not available because of the lack of the sunspot number data. W-index for the last several months cannot be automatically determined by PHITS neither because the daily W-index database contained in the PHITS package is not so frequently updated, whereas the latest W-index data are available from WASAVIES website.[87]
icyear=	(D=2009) Year for calculating W-index. It is effective only when solarmod is not specified and If the period 1614-1950 is specified, monthly average solar activity estimated from sunspot observation data is used; if 1951 onwards, daily average solar activity estimated from ground-based neutron monitor count rates is used. icenv≥0.
icmonth=	(D=10) Month for calculating W-index. It is effective only when solarmod is not specified and icenv≥0.
icday=	(D=20) Day for calculating W-index. It is effective only when solarmod is not specified and icenv≥0.
depatom=	Atmospheric depth in g/cm^2. It is effective only for the terrestrial mode (icenv>0). If this parameter is not specified, the atmospheric depth is automatically determined from the altitude specified by alti, using the US Standard Atmosphere 1986.[88]The value of depatom obtained from the default alti (at sea level) is approximately 1033 g/cm^2.
alti=	(D=0) Altitude in km for calculating the atmospheric depth when depatom is not specified and icenv>0. The mean altitude of the orbit in km when icenv=0 or -2, where only Low-Earth-Orbit (LEO, 340 – 420 km) can be specified.

[85] NMDB: Real-Time Database for high-resolution Neutron Monitor measurements, http://www01.nmdb.eu/

[86] I.G. Usoskin, K. Mursula, S.K. Solanki, M. Schuessler, and G.A. Kovaltsov, A physical reconstruction of cosmic ray intensity since 1610, J. Geophys. Res. 107(A11), 1374, (2002).

[87] Warning System for Aviation Exposure to Solar Energetic Particle, NICT, https://wasavies.nict.go.jp/FFPday.txt

[88] U.S. Standard Atmosphere, 1976, U.S. Government Printing Office, Washington, D.C., 1976.

Table 5.74: Parameters for source energy distribution (13)

parameter	explanation
	(continued in the cases of `e-type= 25,26`)
`rigid=`	The cut-off rigidity in GV, which indicates the minimum rigidity of cosmic-ray that can penetrate the magnetosphere. For the terrestrial mode (`icenv>0`), it indicates the vertical cut-off rigidity whose range is approximately between 0 and 18 GV. If this parameter is not specified, the vertical cut-off rigidity is automatically determined from the latitude and longitude specified by `glat` and `glong`, respectively, using the database constructed from MAGNETOCOSMICS.[89]For the free-space mode (`icenv`≤0), all particles with rigidity below this value are not generated as source.
`glat=`	(D=90) Geographic latitude for calculating the vertical cut-off rigidity. Positive and negative values indicate the north and south latitude, respectively, in degree. It is effective only when `rigid` is not specified and `icenv>0`.
`glong=`	Geographic longitude for calculating the vertical cut-off rigidity. Positive and negative values indicate the east and west longitude, respectively, in degree. It is effective only when `rigid` is not specified and `icenv>0`.
`environ=`	The parameter for representing the local environments used in the calculation of the terrestrial neutron fluxes (`icenv`=2–4), or the event index for the SEP mode (`icenv`=−1). For reproducing neutron fluxes on ground (`icenv`=2), this parameter indicates the fraction of water in ground. For considering the disturbance of neutron fluxes due to aircraft structures (`icenv`= 3 & 4), the mass of aircraft in 100 ton should be provided. Note that both parameters are NOT pure physical quantities, and can be adjustable. For the SEP mode (`icenv`=−1), this parameter should be an integer from 1 to 4, which corresponds to SPE occurred in Feb. 1956 (=1, default), Nov. 1960 (=2), Aug. 1972 (=3), Sep. 1989 (=4), and Jan. 2005 (=5).

[89] `http://cosray.unibe.ch/~laurent/magnetocosmics/`

5.3.20 Definitions for angular distribution

When setting dir=data, angular distribution parameters are required as shown below. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.75: Parameters for source angular distribution (1)

parameter	explanation
a-type = 1, (11)	Any angular distribution can be specified by giving data set of angle bins a(i) and integrated values of the particle generation probability w(i) by hand. The number of the particle generation in the bin is proportional to w(i), and the specified angular distribution is statistically described. For 1 case, angle is given by cosine, and for 11 case, given by degree.
na =	Number of angular group. Data must be given from the next line by the format as (a(i),w(i),i=1,na), a(na+1).
a-type = 4, (14)	The same angular distribution as in the case of a-type=1,(11) can be specified. Unlike a-type=1,(11), the distribution is described by giving data set of angle bins a(i) and weights of the source particle w(i) by hand. The number of source particles generated in each bin is the same for all angle bin, but integrated values of the weight of source particles are adjusted to be proportional to w(i). The number of source particles generated in each bin can also be changed by specifying q(i). For 4 case, angle is given by cosine, and for 14 case, given by degree.
na =	Number of angular group. Data must be given from the next line by the format as (a(i),w(i),i=1,na), a(na+1). In default (q-type=0), equal number of particle is generated in each cell. The integrated number of source particles generated in each bin is proportional to q(i).
q-type = 0, 1	(D=0) generation option. For 0, q(i)=1 for all i is assumed without the following data. For 1, q(i) must be given from the next line by the format as (q(i),i=1,na).

Table 5.76: Parameters for source angular distribution (2)

parameter	explanation
a-type = 5, (15)	Angular distribution dφ/dΩ(i) is given by $g(x)$. For 5 case, angle is given by cosine, for 15 case, given by degree.
g(x) =	Any analytical function of x, Fortran style. x denotes angle. One can use intrinsic functions and constants C, e.g., g(x) = exp(-c1*x**2).
nn =	Number of angular group.
ag1 =	minimum cut off for angular distribution.
ag2 =	maximum cut off for angular distribution.
a-type = 6, (16)	The same angular distribution as in the case of a-type=5,(15) can be specified. Unlike a-type=5,(15), the number of source particle generated in each bin is the same for all angle bin, but integrated values of the weight of source particles are adjusted to be proportional to $g(x)$. The number of source particles generated in each bin can also be changed by specifying q(i). For 6 case, angle is given by cosine, for 16 case, given by degree.
g(x) =	any analytical function of x, Fortran style. One can use intrinsic functions and constants C.
nn =	Number of angular group. In default (q-type=0), equal number of particle is generated in each cell. The integrated number of source particles generated in each bin is proportional to q(i).
q-type = 0, 1	(D=0) generation option. For 0, q(i)=1 for all i is assumed without the following data. For 1, q(i) must be given from the next line by the format as (q(i),i=1,nn).

5.3.21 Definition for time distribution

For t-type=0,1,2,3,4,5,6, the time distribution function can be set to define the time spectrum of the source particles. If a parameter has a default value (D=***), the parameter can be omitted.

Table 5.77: Parameters for time distribution (1)

parameter	explanation
t-type = 0, 1, 2	(D=0) time distribution. 0: no time-distribution, t=0.0. 1: rectangle distribution. 2: Gaussian distribution.
t0 =	(D=0.0) center of time when t-type = 1 [ns].)
tw =	width of time distribution [ns]. When t-type=1, it means full width. When t-type=2, it means FWHM of Gaussian distribution.
tn =	number of time distribution.
td =	interval of time distribution [ns].
tc =	(D=10×tw) cut off time when Gaussian distribution t-type=2 [ns]
t-type = 3	Any time distribution can be specified by giving data set of time bin t(i) and integrated values of the particle generation probability w(i), and specified time distribution is statistically described.
ntt =	Number of time group. Data must be given from the next line by the format as (t(i),w(i),i=1,ntt), t(ntt+1). The integrated number of the particle generation in the each time bin is proportional to w(i).
t-type = 4	The same time distribution as in the case of t-type=1 can be specified. Unlike t-type=1, the distribution is described by giving data set of time bins t(i) and weights of the source particle w(i) by hand. The number of source particles generated in each bin is the same for all time bin, but integrated values of the weight of source particles are adjusted to be proportional to w(i). The number of source particles generated in each bin can also be changed by specifying o(i).
ntt =	Number of time group. Data must be given from the next line by the format as (t(i),w(i),i=1,ntt), t(ntt+1). In default (o-type=0), equal number of particle is generated in each cell. The integrated number of source particles generated in each time bin is proportional to o(i).
o-type = 0, 1	(D=0) generation option. For 0, o(i)=1 for all i is assumed without the following data. For 1, o(i) must be given from the next line by the format as (o(i),i=1,ntt).

Table 5.78: Parameters for time distribution (2)

parameter	explanation
t-type = 5	Differential spectrum ($d\varphi/dt$) is given by $h(t)$.
$h(x)$	Any analytical function of x[ns], Fortran style. One can use intrinsic function and constants C.
ll	Number of time group. If it is given by positive number, linear interpolation is assumed in a bin. If negative, logarithmic interpolation is assumed in a bin. Integrated number of source particles generated in each cell is proportional to $h(x)$.
tg1	Minimum cut off for time distribution [ns].
tg2	Maximum cut off for time distribution [ns].
t-type = 6	The same time distribution as in the case of t-type=5 can be specified. Unlike t-type=5, the number of source particle generated in each bin is the same for all time bin, but integrated values of the weight of source particles are adjusted to be proportional to $h(x)$. The number of source particles generated in each bin can also be changed by specifying o(x).
$h(x)$	Any analytical function of x[ns], Fortran style. One can use intrinsic function and constants C.
ll	Number of time group. If it is given by positive number, linear interpolation is assumed in a bin. If negative, logarithmic interpolation is assumed in a bin. In default (o-type=0), equal number of particle is generated in each cell. The integrated number of source particles generated in each time bin is proportional to o(i).
tg1	Minimum cut off for time distribution [ns].
tg2	Maximum cut off for time distribution [ns].
o-type = 0, 1	(D=0) generation option. For 0, o(i)=1 for all i is assumed without the following data. For 1, o(i) must be given from the next line by the format as (o(i),i=1,ntt).

When setting t-type=100, an original distribution can be used by defining its function in subroutine “tdis01” of “sors.f,” which a source file of PHITS.

Table 5.79: Parameters for time distribution (3)

parameter	explanation
t-type = 100	User define time distribution. User can write any type of time distribution in the source program of “sors.f”.
tg1	Minimum cut off for time distribution [ns].
tg2	Maximum cut off for time distribution [ns].

5.3.22 Example of multi-source

The following is an example of multi-source, which includes energy distribution and angular distribution described by analytic functions.

Example 3: Example of multi-source

```
 1:  [ Source ]
 2:   totfact = 3
 3:  <source> = 9.72
 4:    s-type = 1
 5:      proj = proton
 6:        z0 = 2
 7:        z1 = 29
 8:        r0 = 5
 9:        r1 = 4
10:       dir = 0.0
11:    e-type = 6
12:       eg1 = 1.e-6
13:       eg2 = 1.e-3
14:       nm  = -200
15:      set: c10[1.e-4]
16:      f(x) = x**(1.5)*exp(-x/c10)
17:  <source> = 1
18:    s-type = 1
19:      proj = photon
20:        z0 = 1
21:        z1 = 2
22:        r0 = 5
23:       dir = -1
24:    e-type = 5
25:       eg1 = 1.e-3
26:       eg2 = 5.e-1
27:       nm  = 200
28:       set: c10[1.e-1]
29:       set: c20[1.e-1/2.35482]
30:      f(x) = exp(-(x-c10)**2/2/c20**2)
31:  <source> = 1
32:    s-type = 1
33:      proj = neutron
34:        z0 = 29
35:        z1 = 30
36:        r0 = 5
37:    e-type = 6
38:       eg1 = 1.e-2
39:       eg2 = 1.e+3
40:       nm  = -200
41:       set: c10[92.469]
42:       set: c20[5.644e+10]
43:      f(x) = c10/c20*exp(-sqrt(x*(x+1876))/c10)*(x+938)/sqrt(x*(x+1876))
44:       dir = data
45:    a-type = 5
46:       ag1 = 0
47:       ag2 = 1
48:       nn  = 200
49:      g(x) = exp(-(x-1)**2/0.3**2)
```

In this example, there are three source subsections started from <source>. In the first source subsection, a cylinder source from z =2 cm to z =29 cm with 5 cm radius is defined, and r1=4 is set. This r1=4 means that the region inside the cylinder with radius 4 cm is not included. In the next source, it is also a cylinder source from z =1 cm to z =2 cm with 5 cm radius without r1. This is a normal thin cylinder. The last one is also a thin cylinder from z =29 cm to z =30 cm with 5 cm radius. The numbers defined after each <source> denote the relative weight of the multi-source. In this example, the relative weight is determined by the relative volume ratio of each source. This means that the source particles are generated uniformly in each source volume. The coordinate distribution of the generated source particles is shown in Fig. 5.7 using [t-product] tally with output=source, and icntl=6.

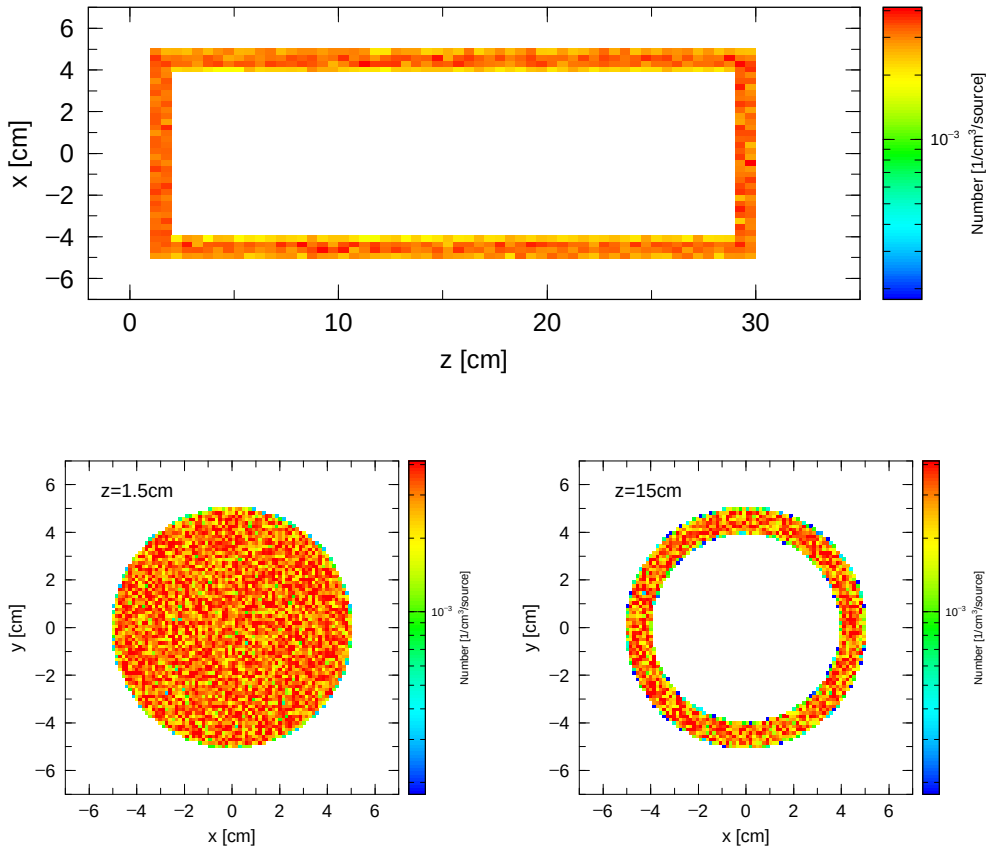


Figure 5.7: Multi-source, coordinate distribution

The source particles of the multi-source are proton, photon and neutron. In each subsection, the energy distribution of the source particle is defined as Maxwellian, Gaussian, and user defined analytical function by using the expression of those function with Fortran style. The first Maxwellian distribution is just equivalent to the expression by `e-type=7` as

```
e-type = 7
   et0 = 1.e-4
   et1 = 1.e-6
   et2 = 1.e-3
```

The second Gaussian distribution is also equivalent to the expression by e-type=2 as

```
e-type = 2
   eg0 = 1.e-1
   eg1 = 1.e-1
   eg2 = 1.e-4
   eg3 = 5.e-1
```

These energy distributions are shown below by using `[t-product]` tally with `output=source`, and `icntl=6`. The result of each particle is shown in Fig. 5.8 with different colors.

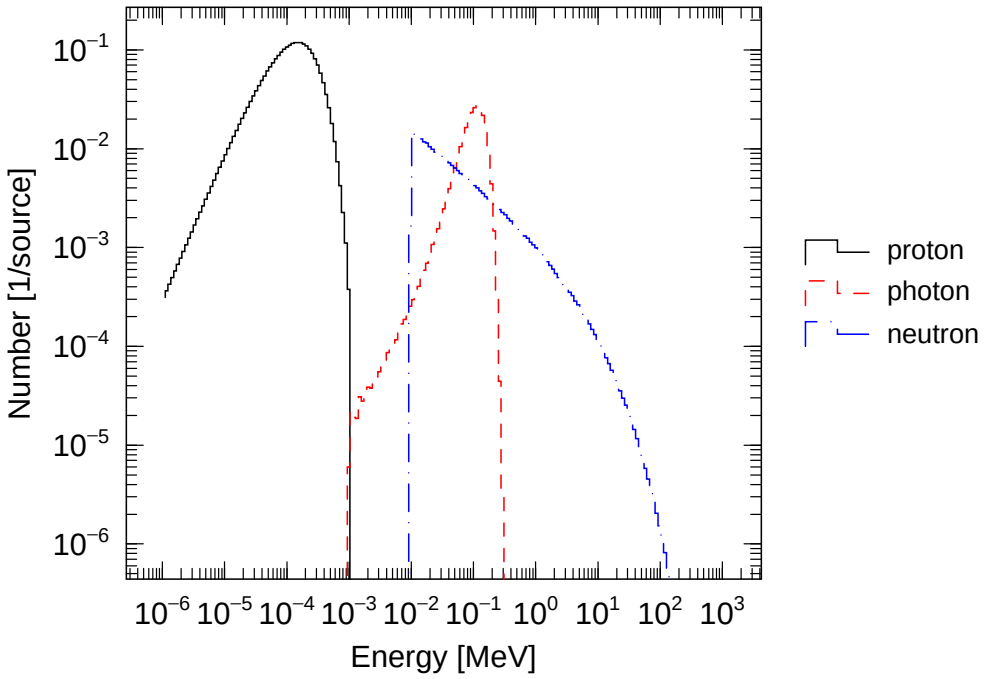


Figure 5.8: Multi-source, energy distribution

The first source has an angular distribution defined by `dir=0`, which means 90 degrees direction with respect to z-axis, the second one has `dir=-1`, 180 degrees direction, and the third one has an angular distribution defined by `a-type` description in which an analytic function is used for an angular distribution. The angular distribution of the third one is shown in Fig. 5.9 by using `[t-cross]` tally.

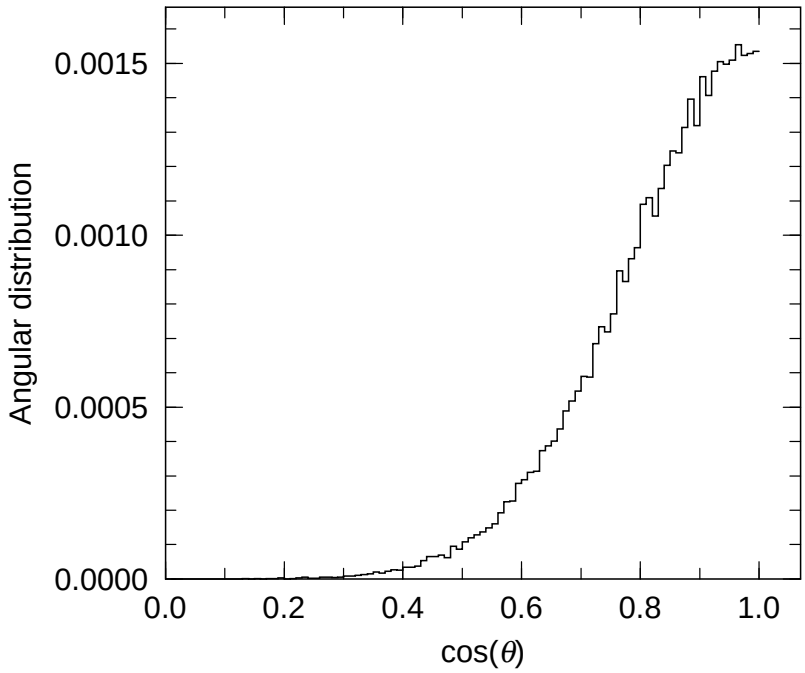


Figure 5.9: Multi-source, angular distribution

5.3.23 Duct source option

For the simulation of neutrons through the long beam-line from the moderator of spallation neutron source or the reactor to the detector room, the following duct source options can be used to reduce the variance of the calculations. The beam current transported through the beam-line decreases proportional to the inverse square of the distance from the moderator. This means that the current crossing the wall of the beam-line, which is called as 'wall current,' at 100 m point is six order of magnitude smaller than that at 1 m point from the source if an isotropic distribution of the source direction is assumed. To reduce this variance, a special option of the source function cab be used. When using this option, the wall current of the simulation particles is equalized at any point of the beam-line by changing the importance weight of the particles to simulate a real situation of the current inside the beam-line.

The duct source option is available when setting `dom=-10` in the circle or rectangle source, which are set by `s-type=1` or `2`, respectively. The parameters for the duct source options are summarized in Table 5.80.

Table 5.80: Parameters for duct source options

parameter	explanation
`dom = -10`	specify the duct source
`dl0 =`	starting z position of the beam-line from z0[cm].
`dl1 =`	starting z position of the duct source from z0[cm].
`dl2 =`	ending z position of the duct source from z0[cm].
`dpf =`	portion of pass through particles at dl2
`drd =`	radius of circle beam line for s-type = 1[cm].
`dxw =`	x size of rectangle beam line for s-type = 2[cm].
`dyw =`	y size of rectangle beam line for s-type = 2[cm].

A shape of the beam-line is assumed to be circle or rectangle for `s-type=1` or `2`, respectively. The setting of `z1=z0` and `dir=1` is also assumed. The latter parameter `dir` means the direction of the beam-line. To change the direction of the beam-line, use the transformation by setting `trcl=`*number of transformation*. The source particles are generated within the circle or rectangle region at `z0` defined by `r0` or `x0, x1, y0, y1`, for `s-type=1` or `2`, respectively. The direction of the particle is determined by the wall position where it reaches within `dl1` and `dl2` so as to equalize the wall current at any point within this region changing the importance of the particle. Overall normalization factor is defined as a number of the source particles which pass the entrance of beam-line at `dl0` originated within the same region at the source position `z0` as that at `dl0`. If the source region at `z0` can be seen at all duct wall positions from `dl0` to `dl2`, the normalization factor is set to be 1. If the source region at `z0` is larger than the area of the beam-line at `dl0`, the source particle from the outer region at `z0` is not counted as the normalization number at `dl0`. This means that the extra region at `z0` increases the current in the beam-line without changing the normalization factor. In the above argument, the angular distribution of the source particle is assumed to be isotropic within the small solid angle which covers the whole beam-line.

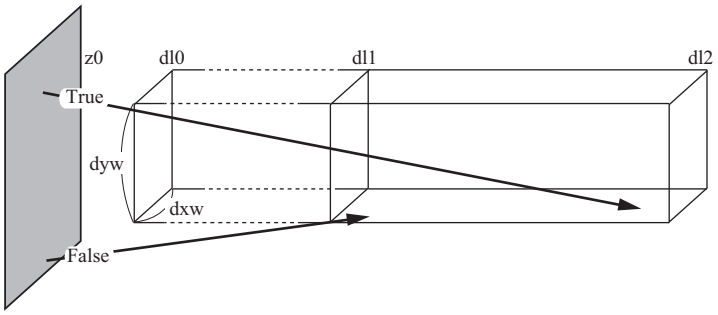


Figure 5.10: Schematic image of the duct source.

Some example of the duct source option are shown below.

In the first example, the rectangle source and beam-line, the same size of the source and beam-line dimensions, are defined. The input for the duct source option is as follows:

Example 4: duct source option, example (1)

```
 1    [ Source ]
 2:
 3:   set: c1[200]    $dl0
 4:   set: c2[500]    $dl1
 5:   set: c3[5000]   $dl2
 6:   set: c4[5.0]    $x*2 at z0
 7:   set: c5[5.0]    $y*2 at z0
 8:   set: c10[5.0]   $dxw
 9:   set: c20[5.0]   $dyw
10:   set: c30[0.001] $dpf
11:
12:   s-type = 2
13:   proj   =  neutron
14:   e0     =  20.0
15:   x0     = -c4/2
16:   x1     =  c4/2
17:   y0     = -c5/2
18:   y1     =  c5/2
19:   z0     =  0.0
20:   z1     =  0.0
21:   dir    =  1.0
22:   phi    =  0.0
23:
24:   dom = -10
25:   dl0 = c1
26:   dl1 = c2
27:   dl2 = c3
28:   dxw = c10
29:   dyw = c20
30:   dpf = c30
```

In the first part of above source section, some constants are set to define the duct source option, `dl0, dl1, dl2`, and size of source, `dxw, dyw, dpf`. In the second part, the position and xy region of the source, direction of the beam-line and the energy of source particle are defined. In the third part, the duct source options are defined. The setting gives the calculaton of the particle transport in the beam-line from 5 m up to 50 m by this duct source and the current, wall current by using the cross tally. The results are shown in Fig. 5.11 compared with an ideal case in which the current and the wall current are proportional to $1/L^2$ and $1/L^3$, respectively. The cross marker in the figure indicates the position of `dl0` and show that the current at this point is unit. The results of the duct source option agree very well with the analytical results.

In the next example, only the size of the source is changed from the previous example.

Example 5: duct source option, example (2)

```
 1    [ Source ]
 2:
 3:   set: c1[200]    $dl0
 4:   set: c2[500]    $dl1
 5:   set: c3[5000]   $dl2
 6:   set: c4[10.0]   $x*2 at z0
 7:   set: c5[10.0]   $y*2 at z0
 8:   set: c10[5.0]   $dxw
 9:   set: c20[5.0]   $dyw
10:   set: c30[0.001] $dpf
11:
12:   s-type = 2
13:   proj   =  neutron
14:   e0     =  20.0
15:   x0     = -c4/2
16:   x1     =  c4/2
```

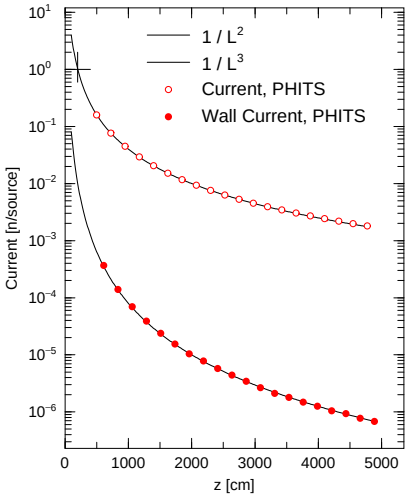


Figure 5.11: Duct source option, example 1

```
17:   y0     = -c5/2
18:   y1     =  c5/2
19:   z0     =  0.0
20:   z1     =  0.0
21:   dir    =  1.0
22:   phi    =  0.0
23:
24:   dom = -10
25:   dl0 = c1
26:   dl1 = c2
27:   dl2 = c3
28:   dxw = c10
29:   dyw = c20
30:   dpf = c30
```

Figure 5.12 shows how the extra region of the source increases the current and the wall current. By this function, the margin area of the moderator to the size of the cross-section of beam-line can be automatically treated.

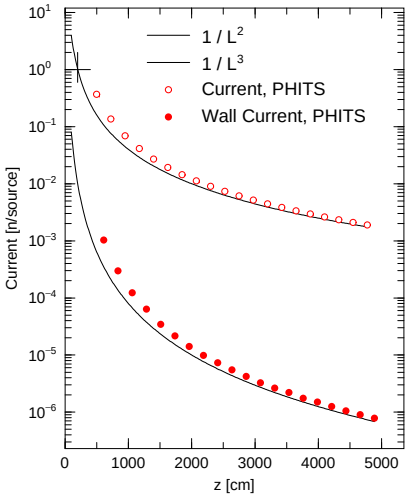


Figure 5.12: Duct source option, example 2

5.4 [Material] section

5.4.1 Formats

Materials, which are used to define 3-dimensional geometry, are defined in this section. The defined materials are refered to some sections such as [cell] section.

The [material] section is defined by material number, elements (or nuclides), and their composition ratios. Following a material number, element symbols and their ratios should be alternately written to define a compound or mixture. The format is:

```
[ Material ]
MAT[n]     element   ratio     element   ratio     ...    ...
```

Two formats to specify material numbers can be used as follows: MAT[*n*] and M*n*, where *n* can be specified up to a material number of 99,999 unless it is over-defined. Note that a blank space cannot be set between MAT and [in the format MAT[*n*].

The following comment marks can be used: #, %, !, $. Although version 2.88 or before, c␣ (␣ is a space) can be also used as comment marks, after version 2.89, it is not permitted in the default setting. To use c␣ as comment marks in this section, set icommat=1 in the [parameters] section.

5.4.2 Element (nuclide) definition

The element in the above format can be specified using element symbols. To define nuclide (isotope), add its mass number to the symbol as follows: 208Pb, Pb-208, or 82208. For instance, hydrogen is defined as 1H, H-1, or 1001. The natural abundance of isotopes can be defined using only an element symbol or a no-mass style as, e.g., Pb, 82000. Note that the natural isotope ratio format cannot be used for an isotope not included in the nuclear data library JENDL-4.0.

To use nuclear data libraries other than JENDL-4.0, copy all addresses of the new library and then paste it at the end of the address file file(7) (D=xsdir). When setting the new library of low-energy neutrons (below 20MeV) other than JENDL-4.0, specify the "library id" in each material number by NLIB parameter, which is shown in Table 5.81. The "library id" consists of the library number (double-digit) and data class (character a-z). For example, 50c for the low-energy neutron data library of JENDL-4.0. The libraries of photons, electrons, and protons can be also specified by PLIB, ELIB, HLIB, respectively. The library id can be specified by nuclide instead of by material number following the element definition and a period as, e.g., 208Pb.50c, Pb-208.50c, or 82208.50c. In PHITS, only one type of the library should be used for each incident particle, and the id should be specified by the NLIB, PLIB, ELIB, HLIB parameters. Only for neutrons, the second library can be used; in this case, after specifying the library id by NLIB, define the nuclide by the expression of 208Pb.50c. Note that only neutron data library can be specified by the extension after each substance. For example, when you use proton data libraries of JENDL-4.0/HE for ^{6}Li and ^{7}Li, please specify them with HLIB as shown below:

```
6Li 0.001
7Li 0.999
hlib=51h
```

If the library id is not specified by the user, PHITS searches the address file file(7) (D = xsdir) from the top line for the library number id corresponding to the nuclide and uses the corresponding data library. Information on the data library used in a PHITS calculation is written in the summary output file file(6) (D=phits.out) when kmout=1 is set in the [parameters] section.

When using EGS5, restrict the number of elements (not nuclides) in one material number to 20.

5.4.3 Composition ratio definition

The composition ratios of the defined elements are given in ratio in the above format. Two ways are available for the ratio definition: if ratio takes a positive value, it means atomic ratio; for negative value, it does mass ratio.

For instance, water (H_2O) can be defined as follows:

```
MAT[1] H 2  O 1
```

or

```
MAT[1] H -2/18  O -16/18
```

In the latter case, the mass ratios of hydrogen and oxygen are given as about 2/18 and 16/18, respectively, because the molecular mass of the water is about 18.

The material densities used in transport calculations are given in the `[cell]` section. Note that material densities must be defined in the `[material]` section instead of composition ratios when using `letmat` parameter in `[t-deposit], [t-deposit2], [t-let], [t-sed]` sections. In this case, if it takes a positive value, the particle density is [10^{24} atoms/cm^3], and for negative value, it is given as a mass density [g/cm^3].

5.4.4 Material parameters

For regions in which nuclear data are used, the material parameters for each material can be set using the format *keyword=value*. The parameters can be set anywhere in the material subsection. The full set of material parameters is listed in Table 5.81.

Table 5.81: Material parameters

parameter	value	explanation
GAS	D = 0	Density effect correction to electron stopping power.
	= 0	Appropriate for materials in the condensed (solid or liquid) state used.
	= 1	Appropriate for material in the gaseous state.
ESTEP	= *n*	Make sub step number *n* for electron transport: ignored when *n* is smaller than the built-in default value.
NLIB	= *id*	Change default neutron library number *id*.
PLIB	= *id*	Change default photon library number *id*.
ELIB	= *id*	Change default electron library number *id*.
HLIB	= *id*	Change default proton library number *id*.
COND		Conductor settings.
	< 0	Non-conductor.
	= 0	(default) Non-conductor if there exists at least 1 non-conductor; otherwise, conductor.
	> 0	Conductor if there exists at least 1 conductor.

5.4.5 Direct specification of stopping power

In general, PHITS automatically calculates the stopping power of each material using models such as ATIMA and SPAR specified by `ndedx` parameter in `[parameters]` section. If you need to use certain databases of the stopping power such as those given in ICRU90, you can use them by specifying the filename of the database as follows:

```
dedxfile = filename
```

This database file should be contained in the folder specified by `file(29)` whose default setting is `file(1)/data/dedx`. In the default folder, the databases of stopping powers of protons and α particles in several materials calculated by PSTAR and ASTAR[90], respectively, are contained. Note that this function is effective only for particles whose stopping power is set to be calculated by ATIMA. The energy and angular straggling can be considered according to `nedisp` and `nspred` parameters, respectively, in the same as the model calculation.

[90] https://www.nist.gov/pml/stopping-power-range-tables-electrons-protons-and-helium-ions

5.4.5.1 Format of stopping-power database

In the stopping-power database, $ and # can be used as the comment remark. All letters are case-insensitive in the same as the PHITS input file. Only two parameters can be specified in the database, which are `unit` and `kf` for defining the unit of particle energy and particle type, respectively. `unit` should be defined before specifying kf parameter.

```
unit = 1: MeV (Total kinetic energy, default)
     = 2: MeV/u (Kinetic energy per atomic mass unit)
     = 3: MeV/n (Kinetic energy per nucleon)
```

`kf` indicates the kf code of the radiation. After specifying `kf`, the database of the stopping power should be given as

Energy Stopping_power

in the ascending order of the energy. The unit of the stopping power should be MeV/(g/cm2). The stopping power of radiation having an energy outside the defined region or having kf code that is not defined in the file is calculated by ATIMA. You can define stopping power databases for more than 1 radiation type in one file, but you cannot use more than 1 stopping power database file for a material.

5.4.6 $S(\alpha,\beta)$ settings

In the transport of low-energy neutrons, the library of the thermal scattering law data $S(\alpha,\beta)$ may be required. This library plays an important role in describing the transport of thermal neutrons. This library can be set as follows:

```
MTn materialID
```

where `n` is the material number and `materialID` is the ID number, such as `lwtr.20t`, written in "xsdir". For example, the library for water at room temperature (at 296K) can be set as follows:

```
M1   H    2.0
     O    1.0
MT1  lwtr.20t
```

See "\phits\XS\tsl\tsl-table" for detailed information for these data.

5.4.7 Chemical form specification for track structure calculation

Chemical form of the material is important for track-structure calculation. Specification of chemical form is not necessary for proton/carbon ion track-structure calculation by KURBUC and electron track structure calculation because the cross section is calculated by scaling the cross section of liquid water. However, track-structure simulation of protons and ions by ITSART (Ion Track Structure calculation model for Arbitrary Radiation and Targets) can consider the elemental composition of the materials. Moreover, the molecular structure of following substances is considered if specified in `[material]` section.
H_2O, CO_2, NH_2, NH_3, SF_6, TeF_6, CH_4, CH_3, C_2H_2, C_2H_4, C_2H_6, C_6H_6,$(CH_3)_2NH$

The chemical form of a compound is specified as follows.

```
M1   H   2.0    O 1.0
chem = H2O
```

Mixture of compounds and pure materials is written in this way using molar ratios.

```
M1   H  1.62  O  0.01  N 1.6  C 0.4  Ar  0.1
chem = H2O   0.01  N2   0.8    CH4   0.4    Ar   0.1
```

The molar ratio inside `Chem` is automatically normalized therefore the magnitude is not important.

5.4.8 Examples

Some examples using the materials parameter are shown below.

Example 6: material example (1)

```
 1:  [ Material ]
 2:  MAT[ 1 ]
 3:     1H        1.0000000E-04
 4:   208Pb       1.7238000E-02
 5:   204Pb       4.6801000E-04
 6:   206Pb       7.9430000E-03
 7:   207Pb       7.2838000E-03
 8:  MAT[ 2 ]
 9:     1H        1.0000000E-09
10:    14N        4.6801000E-05
11:    16O        7.9430000E-06
```

By default, the order is element, then ratio; these can be specified in reverse by putting den and nuc as,

Example 7: material example (2)

```
 1:  [ Material ]
 2:     den            nuc      <------
 3:  MAT[ 1 ]
 4:  1.0000000E-04      1H
 5:  1.7238000E-02    208Pb
 6:  4.6801000E-04    204Pb
 7:  7.9430000E-03    206Pb
 8:  7.2838000E-03    207Pb
 9:  MAT[ 2 ]
10:  1.0000000E-09      1H
11:  4.6801000E-05     14N
12:  7.9430000E-06     16O
```

Example 8: material example (3)

```
 1:  [ Material ]
 2:  m1     80196.49c  5.9595d-5
 3:         80198.49c  3.9611d-3
 4:         80199.49c  6.7025d-3
 5:         80200.49c  9.1776d-3
 6:         80201.49c  5.2364d-3
 7:         80202.49c  1.1863d-2
 8:         80204.49c  2.2795d-3
 9:  c      ...Be...
10:  m3     4009.37c  1.2362E-1
11:  mt3    be.01
12:  c      ...h2o (25C)...
13:  m4     1001.37c  6.6658d-2      8016.37c  3.3329d-2
14:  mt4    lwtr.01
15:  c      ...b4c (natural boron; 25%-density)...
16:  m5     6012.37c  6.8118d-3
17:         5011.37c  2.1825d-2
18:  c      ...liquid-h2 (20K)...
19:  m6     1001.49c  3.1371d-2      1011.49c  1.0457d-2
20:  mt6    orthoh.00   parah.00
```

5.5 [Surface] section

5.5.1 Formats

In the [surface] section, surfaces can be set to define cells in the [cell] section. Several surfaces, including planes, sphere surfaces, cylindrical surfaces, etc., can be set in terms of the parameters in the equations defining their respective shapes. Only $ and c␣ (␣ is a space) can be used as comment marks in [surface]. To use continuation lines, at least five blanks can be placed at the line head in place of the line sequential mark at the end of line.

The [surface] section is defined in the order data: surface number, coordinate transform number, surface symbol, and surface definition; these are explained in Table 5.82. The format is as follows:

```
[ Surface ]
surface number     transform number     surface symbol     surface definition
```

The surfaces expressed by the equations in Table 5.83 or the macro body in Table 5.83 can be referred to using their surface symbols. In addition, mathematical expressions and user defined variables can be used for surface definition. Reflective and white boundary conditions can be set by writing * and +, respectively, before the surface number. For example, *10 indicates that surface 10 is a reflective boundary. The reflective boundary condition is useful to develop an infinite repeated structure. However, the function of the reflection boundary does not work in tetra-mesh geometry. Instead by preparing a physically meaningless thin region in the tetra-mesh geometry and defining reflection boundary inside, the reflection boundary in tetra-mesh geometry can be realized.

Table 5.82: Surface definition format

item	explanation
surface number	Use any number from 1 to 999,999.
transform number	Specifies the number *n* of TR*n* defined in the [transform] section. Does not specify if not using the coordinate transform.
surface symbol	Specifies a surface symbol in Tables 5.83 and 5.84.
surface definition	Species parameters of 1 ~ 15 inputs, which depend on the surface symbol.

Table 5.83: Surfaces expressed by equations

Surface Symbol	Type	Explanation	Equation	Parameters
P	plane	multi-purpose	$Ax + By + Cz - D = 0$	$A,\ B,\ C,\ D$
PX		vertical with X-axis	$x - D = 0$	D
PY		vertical with Y-axis	$y - D = 0$	D
PZ		vertical with Z-axis	$z - D = 0$	D
P	plane	defined by 3 coordinates		$x_1, y_1, z_1,\ x_2, y_2, z_2,$ x_3, y_3, z_3
SO	sphere	origin is center	$x^2 + y^2 + z^2 - R^2 = 0$	R
S		multi-purpose	$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 - R^2 = 0$	$x_0,\ y_0,\ z_0,\ R$
SX		center on X-axis	$(x - x_0)^2 + y^2 + z^2 - R^2 = 0$	$x_0,\ R$
SY		center on Y-axis	$x^2 + (y - y_0)^2 + z^2 - R^2 = 0$	$y_0,\ R$
SZ		center on Z-axis	$x^2 + y^2 + (z - z_0)^2 - R^2 = 0$	$z_0,\ R$
C/X	cylinder	parallel with X-axis	$(y - y_0)^2 + (z - z_0)^2 - R^2 = 0$	$y_0,\ z_0,\ R$
C/Y		parallel with Y-axis	$(x - x_0)^2 + (z - z_0)^2 - R^2 = 0$	$x_0,\ z_0,\ R$
C/Z		parallel with Z-axis	$(x - x_0)^2 + (y - y_0)^2 - R^2 = 0$	$x_0,\ y_0,\ R$
CX		on X-axis	$y^2 + z^2 - R^2 = 0$	R
CY		on Y-axis	$x^2 + z^2 - R^2 = 0$	R
CZ		on Z-axis	$x^2 + y^2 - R^2 = 0$	R
K/X	cone	parallel with X-axis	$\sqrt{(y - y_0)^2 + (z - z_0)^2} \mp \lvert t\rvert\,(x - x_0) = 0$	$x_0,\ y_0,\ z_0,\ \lvert t\rvert^2,\ k$
K/Y		parallel with Y-axis	$\sqrt{(x - x_0)^2 + (z - z_0)^2} \mp \lvert t\rvert\,(y - y_0) = 0$	$x_0,\ y_0,\ z_0,\ \lvert t\rvert^2,\ k$
K/Z		parallel with Z-axis	$\sqrt{(x - x_0)^2 + (y - y_0)^2} \mp \lvert t\rvert\,(z - z_0) = 0$	$x_0,\ y_0,\ z_0,\ \lvert t\rvert^2,\ k$
KX		on X-axis	$\sqrt{y^2 + z^2} \mp \lvert t\rvert\,(x - x_0) = 0$	$x_0,\ \lvert t\rvert^2,\ k$
KY		on Y-axis	$\sqrt{x^2 + z^2} \mp \lvert t\rvert\,(y - y_0) = 0$	$y_0,\ \lvert t\rvert^2,\ k$
KZ		on Z-axis	$\sqrt{x^2 + y^2} \mp \lvert t\rvert\,(z - z_0) = 0$	$z_0,\ \lvert t\rvert^2,\ k$ k is ±1 or unspecified
SQ	ellipse, hyperboloid, paraboloid	parallel with X-, Y-, or Z- axis	$A(x - x_0)^2 + B(y - y_0)^2 + C(z - z_0)^2$ $+2D(x - x_0) + 2E(y - y_0) + 2F(z - z_0)$ $+G = 0$	$A,\ B,\ C,\ D,\ E,$ $F,\ G,\ x_0,\ y_0,\ z_0$
GQ	cylinder, cone, ellipse, hyperboloid, paraboloid	non parallel with X-, Y- and Z-axis	$Ax^2 + By^2 + Cz^2 + Dxy + Eyz +$ $Fzx + Gx + Hy + Jz + K = 0$	$A,\ B,\ C,\ D,\ E,$ $F,\ G,\ H,\ J,\ K$
TX	ellipse torus torus	parallel with X-, Y-, or Z-axis	$(x - x_0)^2/B^2 +$ $(\sqrt{(y - y_0)^2 + (z - z_0)^2} - A)^2/C^2 - 1 = 0$	$x_0,\ y_0,\ z_0,\ A,\ B,\ C$
TY			$(y - y_0)^2/B^2 +$ $(\sqrt{(x - x_0)^2 + (z - z_0)^2} - A)^2/C^2 - 1 = 0$	$x_0,\ y_0,\ z_0,\ A,\ B,\ C$
TZ			$(z - z_0)^2/B^2 +$ $(\sqrt{(x - x_0)^2 + (y - y_0)^2} - A)^2/C^2 - 1 = 0$	$x_0,\ y_0,\ z_0,\ A,\ B,\ C$

When using the surfaces defined in the `[surface]` section in the `[cell]` section, the interior of the shape should essentially be specified in a 'negative sense,' while the exterior should be specified in a 'positive sense.' Note that, in the case of unclosed shapes such as planes, it is necessary to distinguish which side of the plane is 'positive' and which is 'negative' by calculating a value of $f(x, y, z)$ at a point (x_0, y_0, z_0), where $f(x, y, z)$ is the equation of the unclosed shape. If $f(x_0, y_0, z_0) > 0$, the side including the point is 'positive'; if $f(x_0, y_0, z_0) < 0$, the side is 'negative.' For example, for shape symbol PY with $D = 5$, the side including the origin $(0, 0, 0)$ is 'negative' because $f(0, 0, 0) = -5 < 0$.

A plane can be defined by assigning the symbol `P` to the x, y, z-coordinates of three points. In this case, the region including the origin is a negative region.

A cone can be defined in terms of its vertex point x_0, y_0, and z_0 and aligned with any of the x, y, or z axes. By default, the cone will have two sheets extending from the vertex (see Example 13); If parameter k is set = 1, the upper sheet is used, while the lower sheet is used when $k = -1$. If k is not specified, both sheets are used.

Table 5.84: Macro body

Symbol	Type	Parameters	Explanation
BOX	Optional BOX (all angles are 90°)	$x_0,\ y_0,\ z_0,$ $A_x,\ A_y,\ A_z,$ $B_x,\ B_y,\ B_z,$ $C_x,\ C_y,\ C_z$	Base point coordinate Vector from base point to first surface. Vector from base point to second surface. Vector from base point to third surface.
RPP	Rectangular solid (Each surface is vertical with x, y, z axes)	$x_{\min},\ x_{\max},$ $y_{\min},\ y_{\max},$ $z_{\min},\ z_{\max}$	Minimum and Maximum of x. Minimum and Maximum of y. Minimum and Maximum of z.
SPH	Sphere (the same with general sphere S)	$x_0,\ y_0,\ z_0,$ R	Center coordinate. Radius.
RCC	Cylinder	$x_0,\ y_0,\ z_0,$ $H_x,\ H_y,\ H_z,$ R	Center coordinate of bottom face. Height vector from center of bottom face to that of top face. Radius.
RHP or HEX	Optional hexagonal prism Prism	$x_0,\ y_0,\ z_0,$ $H_x,\ H_y,\ H_z,$ $A_x,\ A_y,\ A_z,$ $B_x,\ B_y,\ B_z,$ $C_x,\ C_y,\ C_z$	Base point coordinate. Height vector from base point. Vector from base point to first surface. Vector from base point to second surface. Vector from base point to third surface.
REC	Right elliptical cylinder	$x_0\ y_0\ z_0,$ $H_x,\ H_y,\ H_z,$ $A_x,\ A_y,\ A_z,$ $B_x,\ B_y,\ B_z$	Center coordinate of bottom face. Height vector from center of bottom face (***H***). Major axis vector of ellipse orthogonal to ***H*** (***A***). Minor axis vector of ellipse orthogonal to ***H*** and ***A*** (***B***).
TRC	Truncated right-angle cone	$x_0,\ y_0,\ z_0,$ $H_x,\ H_y,\ H_z,$ $R_1,$ R_2	Center coordinate of bottom face of cone. Height vector from center of bottom. Radius of bottom face. Radius of top face.
ELL	Ellipsoid of revolution (Spheroid)	If $R > 0$, $x_1,\ y_1,\ z_1,$ $x_2,\ y_2,\ z_2,$ R If $R < 0$, $x_0,\ y_0,\ z_0,$ $A_x,\ A_y,\ A_z,$ R	 Coordinate of first focus. Coordinate of second focus. Radius of major axis. In this case, the ellipsoid formed by a rotation on the major axis. The major axis can be set to be shorter than the minor axis. Center coordinate of ellipsoid. Major axis vector. Radius of minor axis.
WED	Wedge	$x_0,\ y_0,\ z_0,$ $A_x,\ A_y,\ A_z,$ $B_x,\ B_y,\ B_z,$ $H_x,\ H_y,\ H_z$	Coordinate of top. Vector to first side of triangle (***A***). Vector to second side of triangle (***B***). Height vector (***H***).

If the top radius (R_2) is set to 0, it is automatically changed to R_2= 1.0e-5. This is because a cone with a vertex cannot be defined using a `TRC` surface.

The bottom of a wedge defined by `WED` must be a right triangle. To make an arbitrary triangle, right triangles of various sizes must be combined.

5.5.2 Examples

This section shows examples of the surface definitions listed in Tables 5.83 and 5.84.

Example 9: Example of Plane (1)

```
1:   [surface]
2:     1   PY     5
```

PX,PY,PZ define planes perpendicular to the x, y, z axes, respectively. In this example, a plane of surface number 1, corresponding to $y = 5$ and perpendicular to y axis, is defined by setting the symbol to PY and the parameter to $D = 5$. Surface number 1 corresponds to the plane shown as a dashed line in Fig. 5.13.

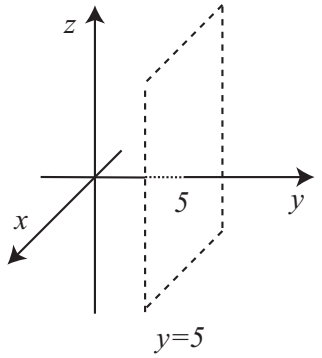


Figure 5.13: The plane $y = 5$ defined in Example 9.

Example 10: Example of Plane (2)

```
1:   [surface]
2:     1   P      2 2 1  10
3:     2   P      5 0 0  0 5 0  0 0 10
```

Arbitrary planes can be defined by specifying P as the symbol. In this case, two types of description can be used to define the plane: four parameters, A, B, C, D, can be used to specify the plane equation $Ax + By + Cz - D = 0$; otherwise, the coordinates of three points, $(x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3)$, can be specified. The former method is useful when the normal vector (s, t, u) and coordinates (x_0, y_0, z_0) of one point on the plane are known. An equation of the plane satisfying these conditions would be given as

$$s(x - x_0) + t(y - y_0) + u(z - z_0) = 0. \quad (7)$$

Then, because

$$sx + ty + uz - (sx_0 + ty_0 + uz_0) = 0, \quad (8)$$

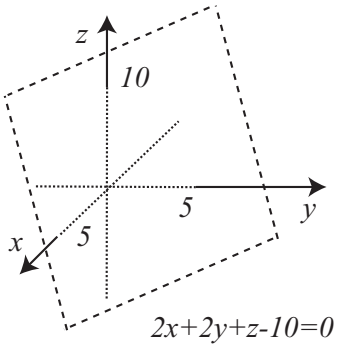


Figure 5.14: The plane defined in Example 10.

the plane can be defined by setting $A = s, B = t, C = u, D = sx_0 + ty_0 + uz_0$. The surface number 1 shown in Example 10 is defined as a plane having the normal vector $(2, 2, 1)$ and containing the coordinate $(5, 0, 0)$. This surface corresponds to the plane denoted by the dashed line in Fig. 5.14. As an illustration of the latter method, surface number 2 is defined as a plane passing through the three coordinates $(5, 0, 0), (0, 5, 0), (0, 0, 10)$. This surface coincides with surface 1 in Figure 5.14.

Example 11: Example of Sphere Surface

```
1:   [surface]
2:     1   SO     5
3:     2   SZ     10  3
4:     3   S      10 10 0  3
```

To define a sphere surface, the symbols `SO,SX,SY,SZ,S` can be used. `SO` is used to define a sphere surface centered at the origin by specifying its radius R as a parameter. In the second line of Example 11, a sphere surface centered at the origin with a radius of 5 cm is defined as surface number 1. The symbols `SX,SY,SZ` can be used to define sphere surfaces centered on the x, y, z axes, respectively. To use these symbols, two parameters —a coordinate value on the appropriate axis, and the radius of the sphere— must be specified. In the third line of Example 11, a sphere surface centered at the coordinate $(0, 0, 10)$, which is a point on the z axis, with a radius of 3 cm is defined as surface number 2. `S` can be used to define a sphere surface centered at an arbitrary coordinate (x_0, y_0, z_0). In this case, four parameters —the coordinate values x_0, y_0, z_0 and the radius R— must be specified. In the fourth line of Example 11, a sphere surface centered at the coordinate $(10, 10, 0)$ with a radius of 3 cm is defined as surface number 3. The three sphere surfaces defined in Example 11 are shown in Fig. 5.15.

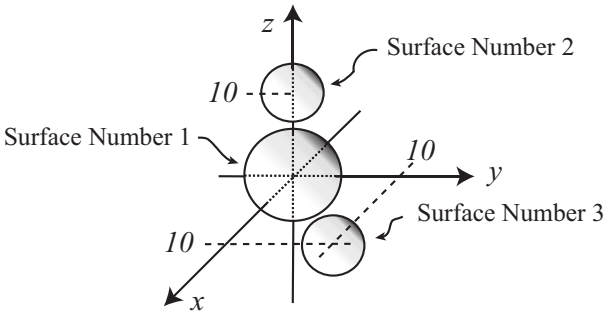


Figure 5.15: The three sphere surfaces defined in Example 11.

Example 12: Example of Cylinder Surface

```
1:   [surface]
2:     1   CY     5
3:     2   C/Y    15 0  3
```

To define a cylinder surface having a central axis along either the x, y, z axes, the symbols `CX,CY,CZ`, respectively, can be specified. In this case, the radius R must be specified as a parameter. In the second line of Example 12, a cylinder surface with a central axis on the y axis with a radius of 5 cm is defined. This surface corresponds to the surface of the cylinder shown in the upper-right of Fig. 5.16. The symbols `C/X,C/Y,C/Z`, can be used to define a cylinder surface with a central axis parallel to the x, y, z axes, respectively. In this case, two coordinate values and the radius of the cylinder must be specified. The third line of Example 12 defines a cylinder surface with a central axis parallel to the y axis. The coordinates $x_0 = 15, z_0 = 0$ are specified as the first and second parameters, respectively, and the radius is given by the third parameter. This surface corresponds to the cylinder in the lower-left side of Fig. 5.16.

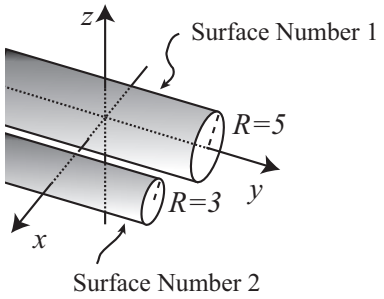


Figure 5.16: The two cylinder surfaces defined in Example 12.

To define a cylinder surface with a central axis that is NOT parallel to any of the x, y, z axes, the symbol `GQ` can be used. However, it is easier to use `CX,CY,CZ,C/X,C/Y,C/Z` and then perform a coordinate transformation using the `[transform]` section than it is to use `GQ`.

Example 13: Example of Cone Surface

```
1:   [surface]
2:     1   KZ     0  1
3:     2   K/Z    0 20 0  1/3  1
```

A cone surface with a central axis along the x, y, z axes can be defined by specifying the symbols `KX,KY,KZ`, respectively. Parameters defining the coordinate of the vertex, an angle parameter, $|t|^2$, and a sheet parameter, k (optional), must also be set. An example in which `KZ` is specified is shown in the second line of Example 13. The first parameter is the z-coordinate of the vertex of the cone, $z_0 = 0$, and the second defines $|t|^2 = 1$. The equation of `KZ` is

$$\sqrt{x^2 + y^2} \mp |t|\,(z - z_0) = 0. \tag{9}$$

After substituting each of the parameters, the following equations are obtained:

$$\sqrt{x^2 + y^2} - z = 0,\ (z > 0) \tag{10}$$

$$\sqrt{x^2 + y^2} + z = 0,\ (z < 0) \tag{11}$$

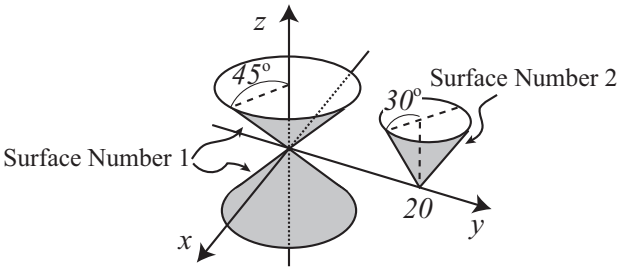


Figure 5.17: The cone surface defined in Example 13.

This formulation corresponds to surface number 1, which is defined as the cone surface having a central axis on the z axis with its vertex at $(0, 0, 0)$. This surface is shown on the left side of Fig. 5.17. The cone in the region $z > 0$ is represented by Eq. (10), while that in $z < 0$ is given by Eq. (11); these surfaces are specified by $k = 1$ and $k = 1$, respectively. Note that, as k is not specified in the second line of the example, both surfaces are defined. The parameter $|t|^2$ is given by the relation $|t|^2 = \tan^2\theta$, where θ is the angle between the central axis and the generating line. In the case of surface number 1, $\tan\theta = 1$, and therefore $\theta = 45°$. To define a cone surface with a central axis parallel to either the x, y, z axes and a vertex at (x_0, y_0, z_0), the symbols `K/X,K/Y,K/Z`, respectively, can be used. The third line of Example 13 shows an example in which `K/Z` is specified. The coordinate of the vertex is $(0, 20, 0)$, and the cone has a central axis parallel to the z axis. Its angle θ is $= 30°$, corresponding to $|t|^2 = 1/3$, or $\tan\theta = 1/\sqrt{3}$. In this case, only one cone surface in the region of $z > 0$ is defined by specifying $k = 1$. Note that one sheet cone ($k = \pm 1$) should not have a skew transformation. If you want to rotate a one-sheet cone surface, please specify another surface to cut a two-sheet cone instead of defining k parameter.

Example 14: Example of Ellipsoid Surface

```
1:   [surface]
2:     1   SQ     1/9**2 1/6**2 1/3**2  0 0 0  -1  0 0 0
```

To define an ellipsoid surface as shown in Fig. 5.18, the symbol `SQ` is specified. The general equation of an ellipsoid surfaces is

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \tag{12}$$

where a, b, c are the lengths of the semi-principal axes along the x, y, z axes, respectively. An ellipsoid surface can therefore be defined by specifying the first to the seventh parameters of `SQ` as $A = 1/a^2, B = 1/b^2, C = 1/c^2$, $D = E = F = 0$, and $G = -1$, respectively. In Example 14, an ellipsoid surface with lengths $a = 9, b = 6, c = 3$ cm is defined. To define a spheroid surface, the two lengths along the two axes that are not the rotational axis must have same values. For example, if the x axis is the rotational axis, b must be set $= c$, namely, $B = C$. The eighth, ninth, and tenth parameters of `SQ` correspond to the coordinates of the center (x_0, y_0, z_0, respectively). In Example 14, the center of the ellipsoid is the origin $(0, 0, 0)$. Note that if D, E, F are set to non-zero values, a quadric surface other than an ellipsoid is defined.

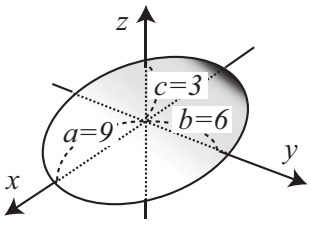


Figure 5.18: The ellipsoid surface defined in Example 14.

Example 15: Example of Hyperboloid Surface

```
1:   [surface]
2:     1   SQ     1/6**2 1/3**2 -1/5**2  0 0 0  -1  0 0 0
3:     2   SQ     -1/6**2 -1/3**2 1/5**2  0 0 0  -1  0 20 0
```

The symbol SQ can be specified to define hyperboloid surfaces, as shown in the center and right-hand side of Fig. 5.19. The surface in the center of the figure is called the hyperboloid of one sheet; those on the right-hand side are hyperboloids of two sheets. Equations of hyperboloids of one and two sheets are given as, respectively,

$$\frac{x^2}{a^2}+\frac{y^2}{b^2}-\frac{z^2}{c^2} = 1, \quad (13)$$

$$-\frac{x^2}{a^2}-\frac{y^2}{b^2}+\frac{z^2}{c^2} = 1. \quad (14)$$

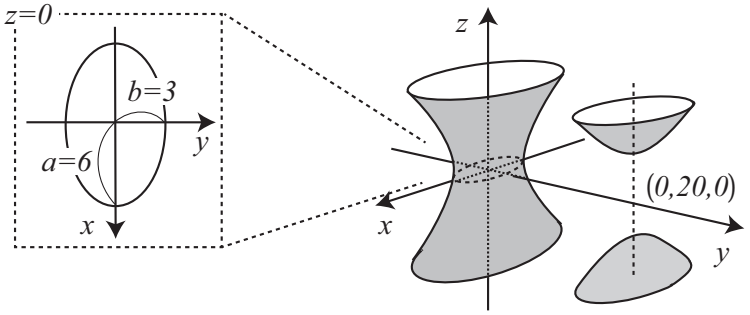


Figure 5.19: The hyperboloid surface defined in Example 15.

Here, the central axis is the z axis. If $a = b$, the shape is called a hyperboloid of revolution. The surface number 1 defined in the second line of Example 15 corresponds to the hyperboloid of one sheet shown in the center of Fig. 5.19. In the second line, the first to the seventh parameters of SQ are given as $A = 1/6^2, B = 1/3^2, C = -1/5^2$, $D = E = F = 0$, and $G = -1$, respectively. The central axis, which corresponds to the z axis, passes through the origin because the eighth to tenth parameters are set to $x_0 = 0, y_0 = 0, z_0 = 0$, respectively. The intersection of this hyperboloid with a plane vertical to the z axis is an ellipse. The figure in the left side of Fig. 5.19 shows the intersection line of surface number 1 with the plane $z = 0$. This ellipse has the major and minor radii of $a = 6$ cm and $b = 3$ cm, respectively. The surface defined in the third line of Example 15 corresponds to the hyperboloid of two sheets shown in the right-hand side of Fig. 5.19. In this case, the signs of a, b, c of surface number 1 are reversed, and $D = E = F = 0$ and $G = -1$ are specified to define a hyperboloid of two sheets. Note that the central axis passes through the coordinate $(0, 20, 0)$ as (x_0, y_0, z_0) is set = $(0, 20, 0)$ in the eighth, ninth, and tenth parameters, respectively. When a surface of a hyperboloid of two sheets is used to define a cell in the [cell] section, one region of the coordinate system (x_0, y_0, z_0) represents the 'negative sense,' while the remaining space represents the 'positive sense.' The interiors of the two sheets shown in the right-hand side of Fig. 5.19 should be specified as 'positive.'

Example 16: Example of Paraboloid Surface

```
1:   [surface]
2:     1   SQ     1 1/2**2 0  0 0 -1  0  0 0 0
```

To define a paraboloid surface as shown in Fig. 5.20, the symbol SQ can be specified as described in Example 16. The equation of a paraboloid surface with a central axis coinciding with the z axis is given as

$$z = \frac{x^2}{a^2}+\frac{y^2}{b^2}. \quad (15)$$

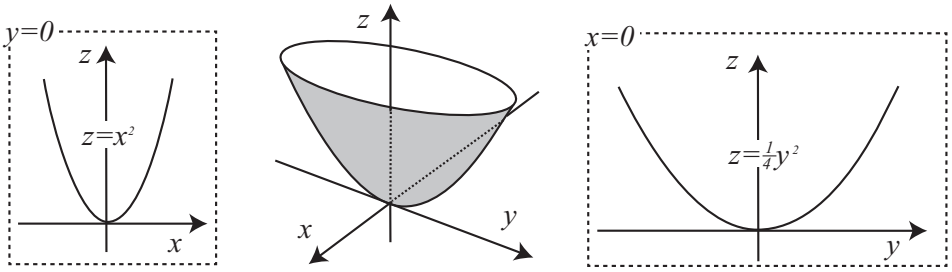


Figure 5.20: The paraboloid surface defined in Example 16.

This surface can be defined by specifying $A = 1/a^2, B = 1/b^2, C = 0$, $D = E = 0, F = -1$, and $G = 0$ as the first to seventh parameters, respectively, of SQ. In the example, a paraboloid with $a = 1, b = 2$ is defined. The intersection of this paraboloid with a plane including the z axis is a parabola. The intersection lines of the paraboloid with $y = 0$ and $x = 0$ are shown in the left and right panels of Fig. 5.20, respectively. The coordinates of the vertex are specified by eighth through tenth parameters; in this case, $x_0 = 0, y_0 = 0, z_0 = 0$.

Example 17: Example of GQ

```
1:   [surface]
2:   set: c1[30]
3:   set: c2[cos(c1/180*pi)]
4:   set: c3[sin(c1/180*pi)]
5:     1   GQ     c2**2 1/2**2 c3**2  0 0 -2*c2*c3  -c3 0 -c2  0
```

Arbitrary surfaces expressed by quadratic equations in x, y, z by can be defined specifying the symbol GQ. Although there similar surfaces such as CX,KX,SQ are also expressed by quadratic equations, GQ can be used to represent quadratic surfaces with a central axis that is NOT parallel to the x, y, z axes. Note that it is easier to use apply a coordinate transformation in the [transform] section to CX,KX,SQ than it is to use GQ. Example 17 represents a paraboloid surface obtained by rotating the surface in Example 16 by an angle of 30° around the y axis. The result is shown in Fig. 5.21. The parameters of this example were obtained by calculating the following transform defined by rotating the coordinate (x, y, z) by an angle of θ around y axis:

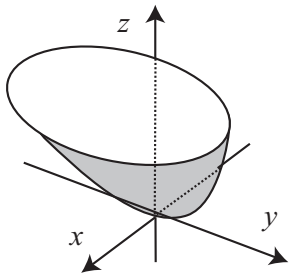


Figure 5.21: The paraboloid surface defined in Example 17.

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}. \tag{16}$$

This relation gives

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} x'\cos\theta - z'\sin\theta \\ y' \\ x'\sin\theta + z'\cos\theta \end{pmatrix}. \tag{17}$$

After substitution of this result into Eq. (15), the following equation of x', y', z' is obtained:

$$\begin{aligned} x'\sin\theta + z'\cos\theta &= \frac{(x'\cos\theta - z'\sin\theta)^2}{a^2} + \frac{y'^2}{b^2}, \\ x'\sin\theta + z'\cos\theta &= \frac{x'^2\cos^2\theta + z'^2\sin^2\theta - 2x'z'\cos\theta\sin\theta}{a^2} + \frac{y'^2}{b^2}, \\ x'\sin\theta + z'\cos\theta &= \frac{\cos^2\theta}{a^2}x'^2 + \frac{\sin^2\theta}{a^2}z'^2 - \frac{2\cos\theta\sin\theta}{a^2}x'z' + \frac{y'^2}{b^2}, \\ 0 &= \frac{\cos^2\theta}{a^2}x'^2 + \frac{1}{b^2}y'^2 + \frac{\sin^2\theta}{a^2}z'^2 - \frac{2\cos\theta\sin\theta}{a^2}x'z' - x'\sin\theta - z'\cos\theta \end{aligned} \tag{18}$$

Comparing this result with the equation of GQ in Table 5.83 and specifying the corresponding parameters defines a paraboloid surface rotated by an angle θ around y axis. In this example, $\theta = 30°$ is given as the constant c1, while c2 and c3 are defined as $\sin\theta$ and $\cos\theta$, respectively.

Example 18: Example of Torus Surface

```
1:   [surface]
2:     1   TZ     0 0 0  10 3 5
```

A torus surface, as shown in the center of Fig. 5.22, can be obtained by rotating an ellipse around a rotational axis outside of the ellipse. In PHITS, torus surfaces with rotational axes corresponding to the x, y, z axes can be specifying the symbols `TX`, `TY`, `TZ`, respectively. The equation of a torus surface with z axis as the rotational axis is given as

$$\frac{\left(\sqrt{x^2+y^2}-R\right)^2}{a^2}+\frac{z^2}{b^2}=1. \tag{19}$$

In this case, the ellipse has major and minor radii z and b, respectively, and a distance between the center of the ellipse and the rotational axis given by R. The center of the torus surface in Example 18 is set at the origin by setting the first, second, and third parameters of `TZ` to zero. The fourth, fifth, and sixth parameters are $A = R = 10$ cm, $B = b = 3$ cm, and $C = a = 5$ cm, respectively. The intersection line of the torus with the plane $z = 0$ is shown in the left panel of Fig. 5.22. The distance between the z axis and the center of the ellipse is 10 cm and the width of the ring is $5 \cdot 2 = 10$ cm. The intersection of the torus with the plane $y = 0$ is shown in the right panel of Fig. 5.22: note that only the region $x > 0$ is shown. The major and minor radii of the ellipse are 5 and 3 cm, respectively.

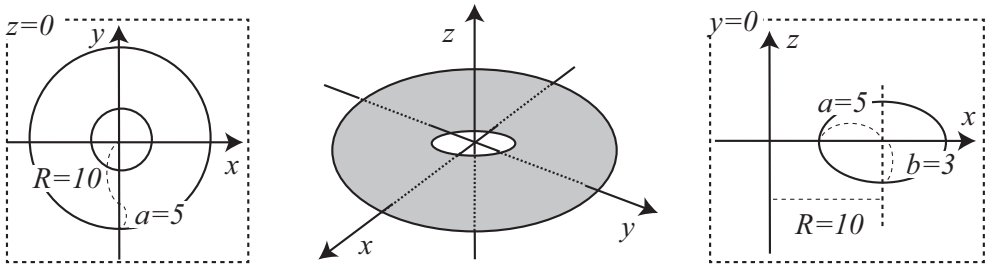


Figure 5.22: The torus surface defined in Example 18.

Example 19: Example of `BOX`

```
1:   [surface]
2:     1   BOX    5 5 0  9 0 0  0 6 0  0 0 3
```

The symbol `BOX` can be specified to define the surfaces of an arbitrary rectangular solid. As parameters of `BOX`, the coordinates of the base point $P(x_0, y_0, z_0)$ and each component of three vectors $\boldsymbol{A}(A_x, A_y, A_z)$, $\boldsymbol{B}(B_x, B_y, B_z)$, and $\boldsymbol{C}(C_x, C_y, C_z)$ must be defined. Figure 5.23 shows the relation between the basepoint and the three vectors. In Example 19, the coordinate of the basepoint is $(5, 5, 0)$ and the three vectors are given as $\boldsymbol{A} = (9, 0, 0)$, $\boldsymbol{B} = (0, 6, 0)$, and $\boldsymbol{C} = (0, 0, 3)$. In using `BOX`, vectors $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ do not have to be parallel to the x, y, z axes, but they should be vertical to each other.

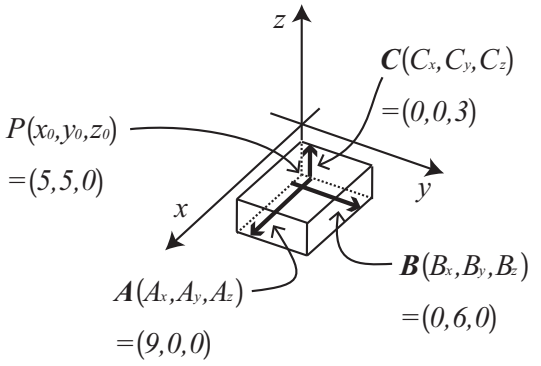


Figure 5.23: The rectangular surface defined in Example 19.

Example 20: Example of RPP

```
1:   [surface]
2:     1   RPP    5 14  5 11  0 3
```

To define the surfaces of a rectangular solid surrounded by three sets of two planes parallel to the xy, yz, zx planes, the symbol RPP can be used. Unlike BOX, an arbitrary rectangular solid with surfaces not parallel to the x, y, z axes CANNOT be defined. The six parameters of RPP must be set by the user as the maximum and minimum values in the x, y, z coordinate system. In Example 20, $x_{\min} = 5, x_{\max} = 14$, $y_{\min} = 5, y_{\max} = 11$, and $z_{\min} = 0, z_{\max} = 3$ are specified as the six parameters. The result is the same rectangular solid defined in Example 19 and is shown in Fig. 5.24.

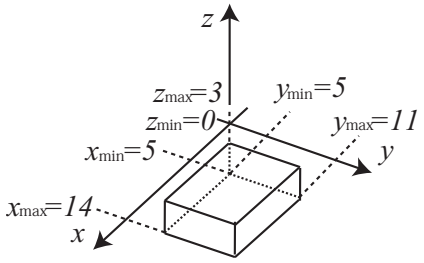


Figure 5.24: The rectangular surface defined in Example 20.

Example 21: Example of SPH

```
1:   [surface]
2:     1   SPH    5 5 5  5
```

Symbol SPH can be used to define a sphere surface with an arbitrary coordinate as its center. The first, second, and third parameters are components of the coordinate (x_0, y_0, z_0) and the fourth parameter is the radius of the sphere. A sphere surface centered at $(5, 5, 5)$ with a radius of 5 cm is defined in Example 21. There is no functional difference between the symbols SPH and S.

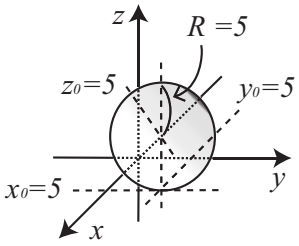


Figure 5.25: The sphere surface defined in Example 21.

Example 22: Example of RCC

```
1:   [surface]
2:     1   RCC    5 5 0  0 0 10  5
```

The symbol RCC can be used to define an arbitrary cylinder surface. The parameters of RCC are the coordinates of the center of the bottom of the cylinder, $P(x_0, y_0, z_0)$, a vector from the bottom to the top, $\boldsymbol{H}(H_x, H_y, H_z)$, and the radius of the cylinder, R. The relation between these parameters is shown in Fig. 5.26. In Example 22, the coordinate $P(5, 5, 0)$ is the center of the bottom circle, the vector is $\boldsymbol{H} = (0, 0, 10)$, and the radius is 5 cm. Unlike symbols such as CX and C/X, it is useful to define a cylinder surface with a central axis NOT parallel to x, y, z axes.

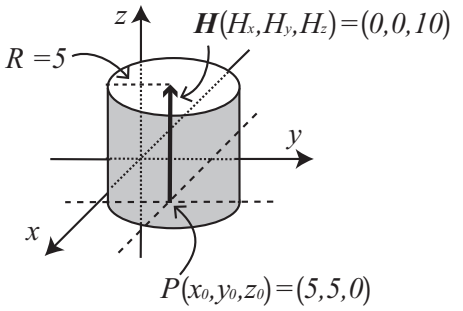


Figure 5.26: The cylinder surface defined in Example 22.

Example 23: Example of RHP

```
1:   [surface]
2:     1   RHP    0 0 0  0 0 10  5 0 0  2 -5 0  -2 -5 0
```

The symbols RHP or HEX can be used to define hexagonal prism surfaces as shown in the right panel of Fig. 5.27. As parameters, the center coordinate of the bottom of the prism, $P(x_0, y_0, z_0)$, a vector from the bottom to the top, $\boldsymbol{H}(H_x, H_y, H_z)$, and three vectors, $\boldsymbol{A}(A_x, A_y, A_z)$, $\boldsymbol{B}(B_x, B_y, B_z)$, and $\boldsymbol{C}(C_x, C_y, C_z)$ must be specified. Vectors $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ are required to define the hexagon comprising the bottom and top surfaces. The left panel of Fig. 5.27 shows the relation between point P and the three vectors; namely, the lengths and directions of the perpendicular lines between P and the three contiguous sides of the hexagon determine the three vectors. In Example 23, the origin is at the center of the bottom hexagon and the height along the z axis of the prism is 10 cm. The components of the origin $(0, 0, 0)$ are specified as the first, second, and third parameters of RHP, respectively, and those of the vector $\boldsymbol{H}(H_x, H_y, H_z) = (0, 0, 10)$ are given as the fourth, fifth, and sixth parameters, respectively. The seventh to fifteenth parameters are the x, y, z components of $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$, respectively. The defined hexagon shown in the left panel of the figure is symmetric with respect to the y axis.

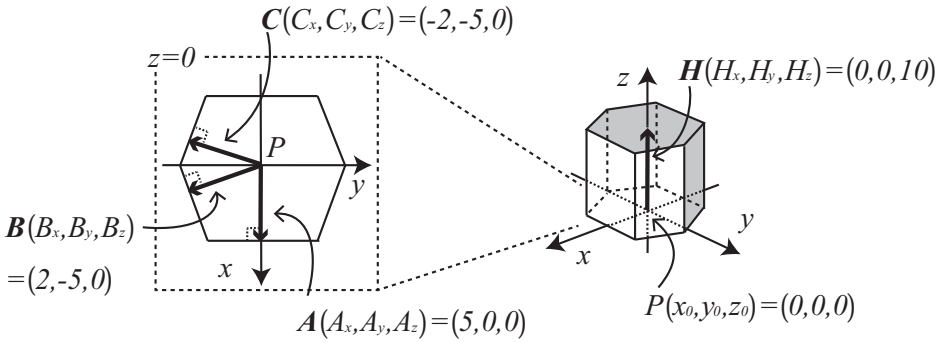


Figure 5.27: The hexagonal prism surface defined in Example 23.

Example 24: Example of REC

```
1:   [surface]
2:     1   REC    0 0 0  0 0 10  5 0 0  0 2 0
```

The symbol REC defines elliptical cylinder surfaces as shown in the right panel of Fig. 5.28. The parameters of REC are the coordinates of the center coordinate of the bottom surface, $P(x_0, y_0, z_0)$, the components of a vector from the bottom to the top, $\boldsymbol{H}(H_x, H_y, H_z)$, and the components of two vectors, $\boldsymbol{A}(A_x, A_y, A_z)$ and $\boldsymbol{B}(B_x, B_y, B_z)$. The relation between the point P and the two vectors $\boldsymbol{A}$ and $\boldsymbol{B}$ is shown in the left panel of Fig. 5.28. The major and minor axis vectors $\boldsymbol{A}$ and $\boldsymbol{B}$ must both be defined using initial point, P, which is the center of the ellipse. The surfaces of an elliptical cylinder with a height along the z axis of 10 cm are defined in Example 24. The center of the bottom is the origin; the first, second, and third parameters of REC are the coordinates of the origin $(0, 0, 0)$, and the fourth, fifth, and sixth parameters are those of vector $\boldsymbol{H}(H_x, H_y, H_z) = (0, 0, 10)$. The x, y, z components of $\boldsymbol{A}$ and $\boldsymbol{B}$ are given in the seventh to twelfth parameters. The vectors $\boldsymbol{A}$ and $\boldsymbol{B}$ are used to form the ellipse shown in the left panel of the figure.

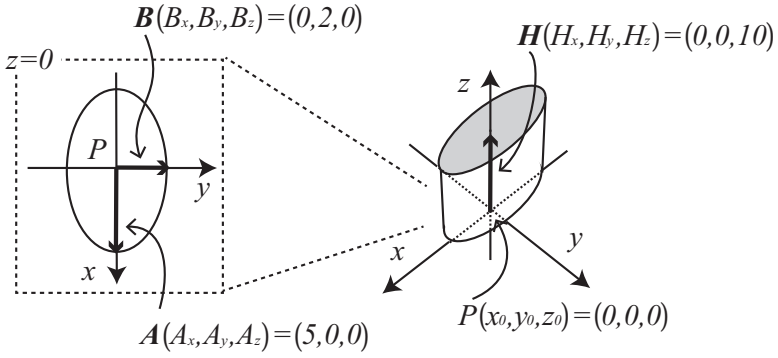


Figure 5.28: The elliptical cylinder surface defined in Example 24.

Example 25: Example of TRC

```
1:   [surface]
2:     1   TRC    0 0 0  0 0 10  5  2
```

The surfaces of a truncated right-angle cone, as shown in Fig. 5.29, can be defined by specifying the symbol TRC. As parameters of TRC, each component of the center coordinate, $P(x_0, y_0, z_0)$, of the cone bottom, the components of a vector from the bottom to the top, $\boldsymbol{H}(H_x, H_y, H_z)$, and two radii of the bottom and top circles, R_1 and R_2, must be specified. In Example 25, the surfaces of the truncated right-angle cone of height 10 cm along the z axis are defined. The center of the bottom circle is the origin. The coordinates of the origin $(0, 0, 0)$ are specified as the first, second, and third parameters, respectively, and those of vector $\boldsymbol{H}(H_x, H_y, H_z) = (0, 0, 10)$ are given as the fourth through sixth parameters. The radius of the bottom is $R_1 = 5$ cm and that of the top is $R_2 = 2$ cm with R_1 and R_2 specified as the seventh and eighth parameters, respectively.

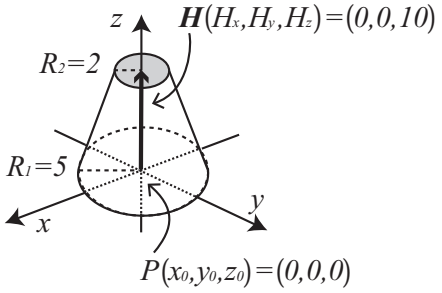


Figure 5.29: The cut cone surface defined in Example 25.

R_2 cannot be set to zero when defining TRC. To set a typical cone, i.e., a non-truncated cone, using TRC, R_2 should be set to a small value.

Example 26: Example of ELL (1)

```
1:   [surface]
2:     1   ELL    3 3 0  -3 -3 0  9
```

The surface of an ellipsoid of revolution (a spheroid) can be set by defining the symbol ELL. After defining an ellipse by specifying the coordinates of two focal points or the coordinate of its center, the ellipsoid of revolution is defined by rotating the ellipse around its major axis. Note that the meanings of the first six parameters of ELL depend on the sign of the seventh parameter, R. Example 26 is an example with $R > 0$. In this case, the coordinates of the two focal points and the radius of the major axis must be specified. The coordinate of the first focus, $P_1(x_1, y_1, z_1)$, is given by the first, second, and third parameters, while that of the second, $P_2(x_2, y_2, z_2)$, is given by the fourth, fifth, and sixth parameters. The seventh parameter is the radius of the major axis, R. The surface of the ellipsoid of revolution defined in Example 26 is shown on the right side of Fig. 5.30. The first and second focal points are $P_1(x_1, y_1, z_1) = (3, 3, 0)$ and $P_2(x_2, y_2, z_2) = (-3, -3, 0)$, respectively. The ellipse is on the xy plane, as shown in the left panel of the figure, with an angle between the major axis and the x axis of 45°. The length of the major axis is $2R = 18$ cm. The ellipsoid is defined by rotating this ellipse around the major axis.

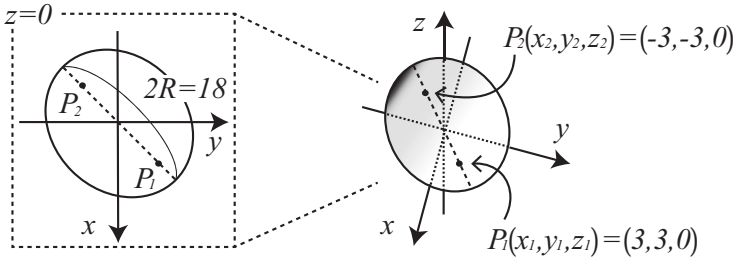


Figure 5.30: The spheroid surface defined in Example 26.

Although an ellipsoid surface can also be defined using SQ, it is easier to define an ellipsoid with an inclined rotational axis using ELL. Note that only surfaces of ellipsoids of revolution can be defined using ELL.

Example 27: Example of ELL (2)

```
1:   [surface]
2:     1   ELL    0 0 0  6 6 0  -6
```

Example 27 apples to the case $R < 0$ in ELL. In this case, the center coordinate, P, major axis, $\boldsymbol{A}$, and radius of the minor axis of the ellipse must be defined. In this example, the center of the ellipse corresponds to the origin. The first, second, and third parameters of ELL are given as $P(x_0, y_0, z_0) = (0, 0, 0)$, while the fourth, fifth, and sixth parameters are $\boldsymbol{A}(A_x, A_y, A_z) = (6, 6, 0)$. The ellipse defined by these parameters is shown in the left panel of Fig. 5.31. The absolute value of the seventh parameter, R, which corresponds to the radius of the minor axis, is 6 cm. The ellipsoid surface obtained by rotating this ellipse around the major axis is shown in the right side of Fig. 5.31.

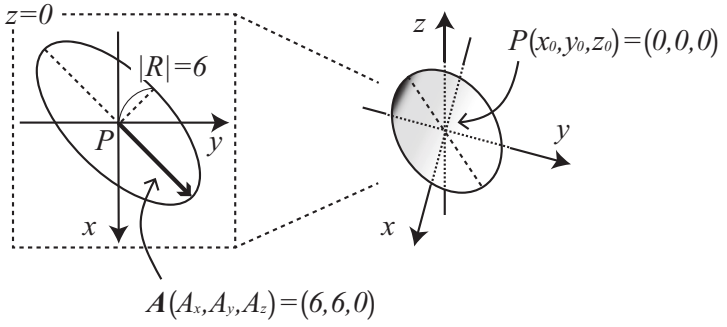


Figure 5.31: The spheroid surface defined in Example 27.

Example 28: Example of WED

```
1:   [surface]
2:     1   WED    0 0 0  10 0 0  0 10 0  0 0 5
```

A wedge surfaces, as shown in Fig. 5.32, can be obtained by specifying the symbol WED. Note that only a right triangle can be used as the bottom of a wedge defined by WED. In Example 28, one vertex of the bottom triangle $P(x_0, y_0, z_0) = (0, 0, 0)$ is given in the first, second, and third parameters of WED, respectively. Two vectors $\boldsymbol{A}(A_x, A_y, A_z) = (10, 0, 0)$ and $\boldsymbol{B}(B_x, B_y, B_z) = (0, 10, 0)$ with initial point P are specified in the fourth to ninth parameters. The triangle defined by the two vectors is a right triangle. The height vector of the wedge, $\boldsymbol{H}(H_x, H_y, H_z) = (0, 0, 5)$ is given in the tenth through twelfth parameters.

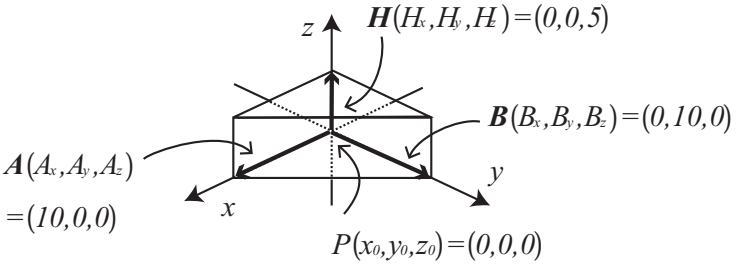


Figure 5.32: The wedge surface defined in Example 28.

5.5.3 Surface definition by macro body

In using a surface defined by a macro body in the cell definition, '-' refers to the inside of the macro body and '+' is the outside. Each surface comprising a macro body can be used in the cell definition, in which case the macro body surface number should be written as '.' plus the macro body surface number.

The surface numbers are listed in Table 5.85.

Table 5.85: Macro Body Surface Numbers

Symbol	Macro Body Surface Number	Explanation
BOX	1	Surface vertical with the end of (A_x, A_y, A_z).
	2	Surface vertical with the origin of (A_x, A_y, A_z).
	3	Surface vertical with the end of (B_x, B_y, B_z).
	4	Surface vertical with the origin of (B_x, B_y, B_z).
	5	Surface vertical with the end of (C_x, C_y, C_z).
	6	Surface vertical with the origin of (C_x, C_y, C_z).
RPP	1	Surface at x_{max}.
	2	Surface at x_{min}.
	3	Surface at y_{max}.
	4	Surface at y_{min}.
	5	Surface at z_{max}.
	6	Surface at z_{min}.
SPH		Only a user-defined surface number is used.
RCC	1	Side face of cylinder.
	2	Surface vertical with the end of (H_x, H_y, H_z).
	3	Surface vertical with the origin of (H_x, H_y, H_z).
RHP or HEX	1	Surface vertical with the end of (A_x, A_y, A_z).
	2	Opposite face for surface 1.
	3	Surface vertical with the end of (B_x, B_y, B_z).
	4	Opposite face for surface 3.
	5	Surface vertical with the end of (C_x, C_y, C_z).
	6	Opposite face for surface 5.
	7	Surface vertical with the end of (H_x, H_y, H_z).
	8	Surface vertical with the origin of (H_x, H_y, H_z).
REC	1	Side face of cylinder.
	2	Surface vertical with the end of (H_x, H_y, H_z).
	3	Surface vertical with the origin of (H_x, H_y, H_z).
TRC	1	Side face of cone.
	2	Surface vertical with the end of (H_x, H_y, H_z).
	3	Surface vertical with the origin of (H_x, H_y, H_z).
ELL		Only a user-defined surface number is used.
WED	1	Surface including the top and bottom hypotenuses.
	2	Surface including ***B*** and ***H***.
	3	Surface including ***A*** and ***H***.
	4	Surface including ***A***, ***B***, and the end of ***H***.
	5	Surface including ***A***, ***B***, and the origin of ***H***.

5.6 [Cell] section

5.6.1 Formats

In the [cell] section, cells can be defined in terms of surfaces described in the [surface] section. A cell should be set as a closed space, and a virtual space for particle transport calculation can be generated by combining defined cells. In PHITS, an 'outer void' must be explicitly defined as a cell.

Only $ and c␣ (␣ is a space) can be used as comment marks in this section; in particular, # cannot be used as a comment mark as this character is used for the cell definition. To use continuation lines, it suffices to place at least five blanks at the line head instead of using the line sequential mark at the end of the line.

The [cell] section is defined in this data order: cell number, material number, material density, cell definition, and cell parameter as keyword style. These are explained in Table 5.86. The format is shown below.

```
[ Cell ]
cell number     mat.  number     mat.  density     cell def.      cell parameter
```

In the LIKE *n* BUT format, the cell parameter format and repeated structures with lattices can be used: see Sec. 5.6.5 for some examples showing how to use this format. The cell parameters are listed and explained in Table 5.87.

Table 5.86: Cell definition format

item	explanation
cell number	Any number from 1 to 999,999 can be used.
material number	Set 0 for void, −1 for outer void, or use a material number defined in the [material] section.
material density	If the cell is void or outer void, no input. When the given value is positive or negative, it is a particle density [10^{24} atoms/cm^3] or mass density [g/cm^3], respectively. A composition ratio defined in the [material] section is used. Thus, different density materials with the same composition as the original can be set in this section. A new parameter, matadd, is used to add different material numbers.
cell definition	Cell geometry is defined by both surface numbers in the [surface] section and the Boolean operators, ␣(blank)(AND), :(OR), and #(NOT). Parentheses (and) can be also used. See Sec. 5.6.2 for details.
LIKE *n* BUT	A cell using this format is the same as an *n* cell except for using only parameters described after BUT.
cell parameter	The format is keyword=value. As a keyword, VOL(volume), TMP(temperature), TRCL(transform), U(universe), LAT(lattice), or FILL can be used. In the LIKE *n* BUT format, MAT(material) and RHO(density) can be used as well.

When defining cells with the same material number but different densities, the cells assume material numbers that differ from that of the first cell.

In operation of the cell definition, ␣(blank) has a higher priority than :.

Table 5.87: Cell parameters

item	explanation
VOL	Volume [cm^3] of the cell is given.
TMP	Temperature [MeV] of the material in the cell is given.
TRCL	Coordinate transform for position of the cell is performed using the defined coordinate transform number defined in the [transform] section or the transform format.
U	Universe number; using this, the number of the universe including the cell is defined. Any number value can be used from 1 to 999,999. See Sec. 5.6.3 for details.
LAT	Lattice number. Setting LAT=1 or 2 defines a quadratic or hexangular prism, respectively: see Sec. 5.6.4 for details.
FILL	Set universe numbers to fill the cell with the universe.
MAT	This is used with the LIKE *n* BUT MAT=*m* format. Using MAT, a cell can be duplicated except with its material number changed to *m*.
RHO	This is used with the LIKE *n* BUT RHO=*x* format. Using RHO, a cell can be duplicated except with its density changed to *x*.

5.6.2 Description of cell definition

Cells are defined by treating regions divided by surfaces defined in the [surface] section. When giving a cell definition, in some cases the concept of 'surface sense' must be used to create a distinction between two regions divided by the surface corresponding to an equation, $f(x,y,z) = 0$; in other cases, the Boolean operators, ␣(blank)(AND), :(OR), and #(NOT) must be used to treat the regions.

'Surface sense' defines one region including a point (x_0, y_0, z_0) by treating $f(x_0, y_0, z_0) > 0$ in a 'positive sense' and the region $f(x_0, y_0, z_0) < 0$ in a 'negative sense.' To define a region in the positive sense, a positive surface number is written in the cell definition space; conversely, a negative surface number is used in the definition to denote a negative sense. An example of the definition of sense is shown below.

Example 29: [cell] section example (1)

```
1:   [ Cell ]
2:     1   0  -10
3:     2  -1   10
4:   [ Surface ]
5:     10  SZ  3  5
```

The tenth surface represents a sphere with a radius of 5 cm. Because the inside of this sphere is in the negative sense, the first cell is defined using a negative number -10. The outer void is explicitly defined as the second cell. This example produces the virtual space shown in Fig. 5.33.

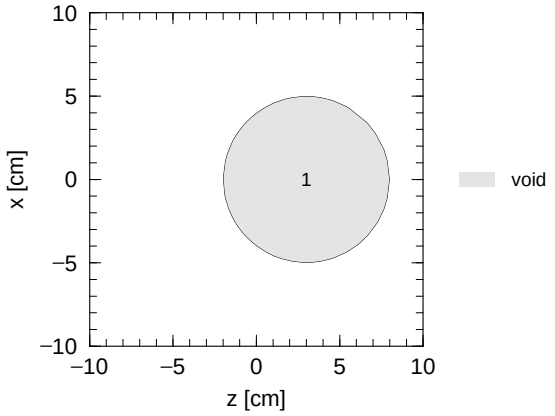


Figure 5.33: Result of [cell] section example (1).

In some cases, the treatment of a region in the cell definition involves the use of Boolean operators. The symbols ␣(blank), `:`, and `#` denote the intersection (AND), union (OR), and complement (NOT), operators, respectively. When using `#all`, it excludes all cells except for those defined as the outer void or in other universes, making it convenient for defining the entire phase space. Parentheses, e.g., `(` and `)`, can be used to combine some regions. The second example in this section uses ␣(blank) and `#`.

Example 30: `[cell]` section example (2)

```
 1:   [ Cell ]
 2:     1   0  11  -12  13  -14  15  -16
 3:     2  -1  #1
 4:   [ Surface ]
 5:     11  PX  -6
 6:     12  PX   6
 7:     13  PY  -6
 8:     14  PY   6
 9:     15  PZ  -6
10:     16  PZ   6
```

In the cell definition in the second line, the three numbers without minus signs correspond to the positive sense regions of the eleventh, thirteenth, an fifteenth surfaces, while those with minus signs correspond to the negative sense regions of the twelfth, fourteenth, and sixteenth surfaces. A region surrounded by these surfaces is defined with ␣(blank) as the first cell, which is the interior of a 12-cm cube. The outside of the cube is defined by the complement operator `#` as the outer void. Figure 5.34 shows the result of this example.

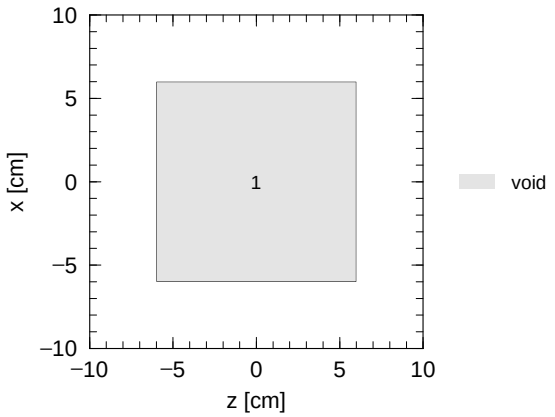


Figure 5.34: Result of `[cell]` section example (2).

The next example uses `:` and parentheses to combine the sphere in the first example and the cube in the second example.

Example 31: `[cell]` section example (3)

```
 1:   [ Cell ]
 2:     1   0  -10 : (11  -12  13  -14  15  -16)
 3:     2  -1  #1
 4:   [ Surface ]
 5:     10  SZ   3  5
 6:     11  PX  -6
 7:     12  PX   6
 8:     13  PY  -6
 9:     14  PY   6
10:     15  PZ  -6
11:     16  PZ   6
```

The numbers surrounded by the parentheses in the second line correspond to the region of the first cell in example (2). In this example, a region combining the inside of the cube with the inside of the sphere in example (1) is defined using the union operator `:` as the first cell. The result is shown in Fig. 5.35.

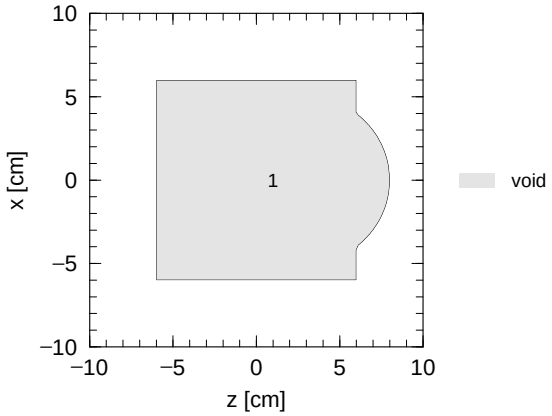


Figure 5.35: Result of `[cell]` section example (3).

The next example shows the division of a cube into two regions by a spherical surface.

Example 32: `[cell]` section example (4)

```
 1:   [ Material ]
 2:     mat[1]  1H 2  16O 1
 3:   [ Cell ]
 4:     1   0       -10
 5:     2   1  -1.0   10 (11  -12  13  -14  15  -16)
 6:     3  -1        #1 #2
 7:   [ Surface ]
 8:     10  SZ   3  5
 9:     11  PX  -6
10:     12  PX   6
11:     13  PY  -6
12:     14  PY   6
13:     15  PZ  -6
14:     16  PZ   6
```

This `[surface]` section is the same as in example (3). In the fifth line, the second cell is defined with ␣(blank) as an overlap region between the outside of the sphere, which is the tenth surface, and the inside of the cube defined by the parentheses. The cell is filled with water as defined in the `[material]` section and the result is shown in Fig. 5.36. The interior of the sphere is the first cell, which is filled with void.

Even empty areas (such as vacuum or air) must be defined in a way that avoids overlapping definitions with other regions. In such cases, use of `#all` command, which excludes all other regions except for those for outer void and in other universe (see next section), is convenient. Note that only one `#all` can be defined per input file.

5.6.3 Universe frame

In PHITS, the cell parameter `U` can be used to define a region of the main space used for particle transport calculation that can be translated to a corresponding region in any universe. This function is very useful for setting the repeated structures introduced in Sec. 5.6.5.

An example using three spaces (one main space and two universes) and shown in Fig. 5.37 is explained below. The main space comprises two rectangular solids. One universe includes a cylinder filled with water, while the other has an iron cylinder surrounded by water. The first cell is filled with a region of universe 1, while the second cell is filled with a region of universe 2.

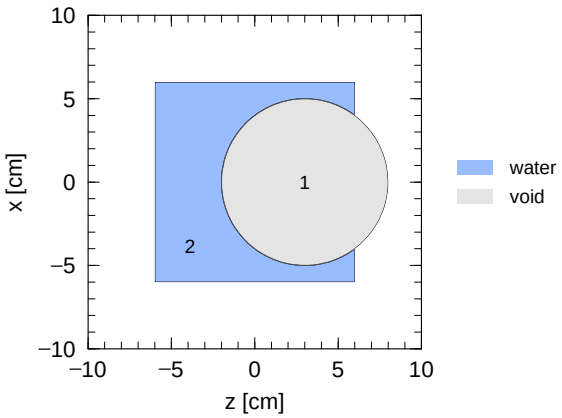


Figure 5.36: Result of `[cell]` section example (4). The first and second cells are filled with void and water, respectively.

Example 33: `[cell]` section example (5)

```
 1:   [ Material ]
 2:     mat[1]  1H 2  16O 1
 3:     mat[2]  Fe 1
 4:   [ Cell ]
 5:       1  0        11  -12  13  -14  15  -17  FILL=1
 6:       2  0        11  -12  13  -14  17  -16  FILL=2
 7:     101  1  -1.0  -10  13  -14  U=1
 8:     102  0       #101  U=1
 9:     201  2 -10.0  -10  13  -14  U=2
10:     202  1  -1.0  #201  U=2
11:       9 -1       #1 #2
12:   [ Surface ]
13:     10  CY   5
14:     11  PX  -6
15:     12  PX   6
16:     13  PY  -6
17:     14  PY   6
18:     15  PZ  -6
19:     16  PZ   6
20:     17  PZ   0
```

Universes `1` and `2` are defined in the seventh and eighth lines and in the ninth and tenth lines, respectively, using cell parameter `U`. These universes have similar structures in which a cylinder is placed at the origin of the coordinate space, but their components inside and outside the cylinder differ, as shown in Fig. 5.37. In the fifth and sixth lines, the first and second cells are defined respectively as regions filled with the corresponding part of each universe using the cell parameter `FILL`. The result of this example is shown in Fig. 5.38, in which it can be seen that the first cell comprises the 101st and 102nd cells in universe 1, while the second cell comprises the 201st and 202nd cells in universe 2.

It is not possible to use an undefined region from one of the universes. If the 102nd cell is not defined in the eighth line as a void region, the first cell cannot be filled with universe 1. Not also that all universes have the same coordinate system definition, with the position of the origin, directions of x, y, and z-axes, and scale of the space in any universe agreeing with those in any other universe. If a different value of `PX` is in the fourteenth and fifteenth lines, the cube will not include some of the cylinder, as shown in Fig. 5.39.

5.6.4 Lattice definition

The cell parameter `LAT` (lattice parameter) is very useful for making repeated structures. In this section, the definition of a unit lattice structure and its simple use are explained using some examples: see Sec. 5.6.5 for a more practical description.

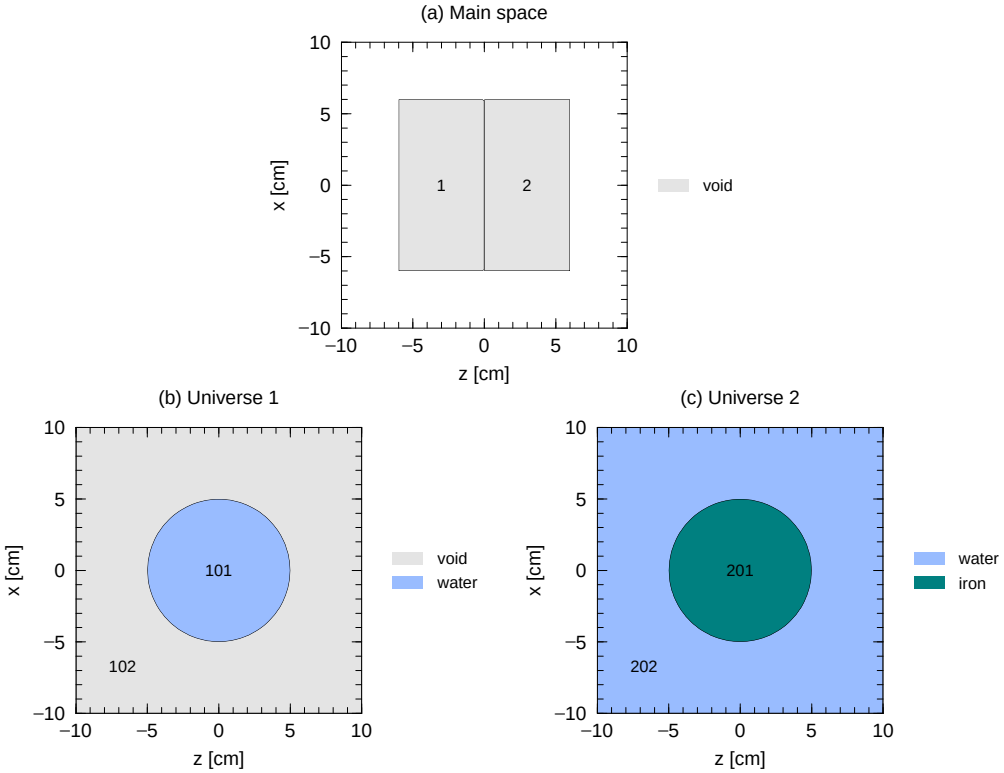


Figure 5.37: (a) Two rectangular solids. (b) Cylinder filled with water. (c) Iron cylinder in water.

The quadratic and hexangular prisms shown in Fig. 5.40 can be used as a unit structure by setting `LAT=1` or `LAT=2`, respectively. A universe can be constructed using repeated structure of such a lattice, and any region can be filled using this universe. Note that each unit must also be filled with another universe defined with any material or void. The numbering of each unit component in Fig. 5.40 corresponds to the surface number order written in the cell definition; the lattice coordinate system, which will be explained below, depends on this order.

An example using a quadratic prism (`LAT=1`) is shown below.

Example 34: `[cell]` section example (6)

```
 1:  [ Material ]
 2:    mat[1]  1H 2  16O 1
 3:  [ Cell ]
 4:      1   0        11  -12   13  -14   15  -16  FILL=1
 5:    101   0       -26   25  -24   23  -22   21  LAT=1  U=1  FILL=2
 6:    201   1  -1.0  -90  U=2
 7:      2  -1       #1
 8:  [ Surface ]
 9:    11  PX  -6
10:    12  PX   6
11:    13  PY  -6
12:    14  PY   6
13:    15  PZ  -6
14:    16  PZ   6
15:    21  PX  -2
16:    22  PX   2
17:    23  PY  -2
18:    24  PY   2
19:    25  PZ  -2
20:    26  PZ   2
21:    90  BOX  -10 -10 -10  20 0 0  0 20 0  0 0 20
```

In the fifth line, a unit cell with `LAT=1` is defined using four surface numbers. Setting `U=1` defines as the repeated structures of this unit, which is filled with universe 2 defined in the sixth line. Because the cross section of the unit in the *x*-*z* plane is a square four cm per side, the first cell defined in the fourth line as a 12 cm cube has nine blocks, as shown in Fig. 5.41. Note that the unit has an infinite length in the *y* direction of universe 1 because of only four

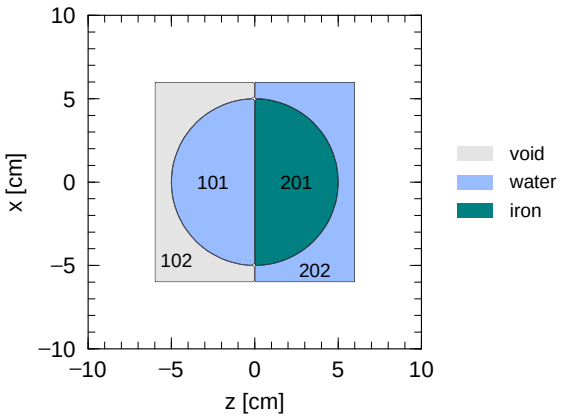


Figure 5.38: Result of `[cell]` section example (5).

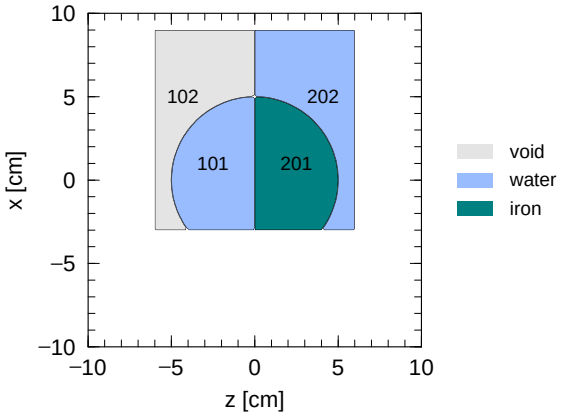


Figure 5.39: Result of `[cell]` section example (5) with the region is shifted in the *x*-direction.

surfaces are defined. To define a finite-length prism, `-24 23` must be added to the cell definition in the fifth line.

To distinguish cells in the repeated structure, each cell is placed at the lattice coordinate (i, j, k), as shown in the right panel of Fig. 5.41. Note that the ordering this coordinate notation corresponds to the general coordinate ordering (x, y, z) and are defined by the order of surface numbers written in the cell definition. To specify any cell using `mesh=reg` in a tally section, the lattice and universe styles `(201 < 101[-1 0 0])` can be used, where the lattice coordinate is represented by `[i j k]`: see Sec. 6.1.2 for more information on this format. The lattice coordinates can be viewed using the `[t-gshow]` tally with `output=7 or 8`.

The following example involves a hexangular prism (`LAT=2`).

Example 35: `[cell]` section example (7)

```
1:   [ Material ]
```

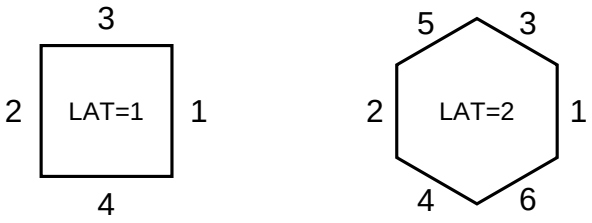


Figure 5.40: Unit structure of lattice.

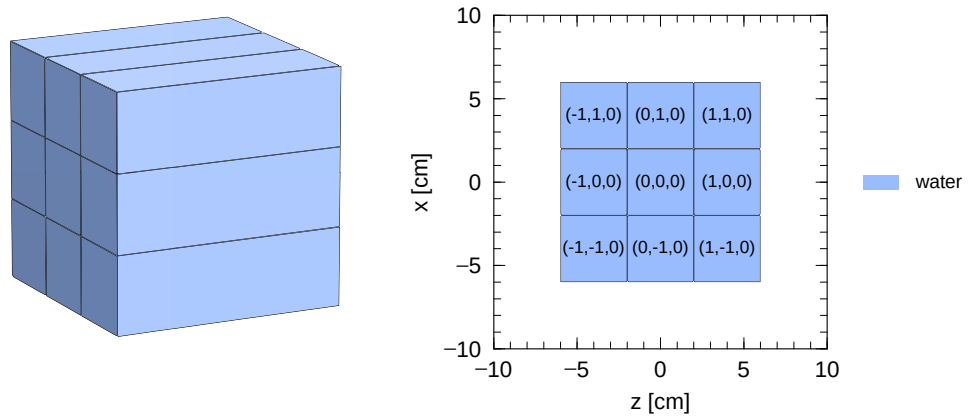


Figure 5.41: Result of [cell] section example (6) in 3D (left) and 2D (right) images.

```
 2:     mat[1]  1H 2  16O 1
 3:   [ Cell ]
 4:       1   0        11 -12  13 -14  15 -16  FILL=1
 5:     101   0       -31  32 -33  34 -35  36 -24  23  LAT=2  U=1  FILL=2
 6:     201   1  -1.0  -90  U=2
 7:       2  -1       #1
 8:   [ Surface ]
 9:     11  PX  -6
10:     12  PX   6
11:     13  PY  -6
12:     14  PY   6
13:     15  PZ  -6
14:     16  PZ   6
15:     23  PY  -2
16:     24  PY   2
17:     set: c1[2]
18:     31  PZ  [ c1*cos(pi/6)]
19:     32  PZ  [-c1*cos(pi/6)]
20:     33  P   1  0  [ 1/tan(pi/3)]  [ c1]
21:     34  P   1  0  [ 1/tan(pi/3)]  [-c1]
22:     35  P   1  0  [-1/tan(pi/3)]  [ c1]
23:     36  P   1  0  [-1/tan(pi/3)]  [-c1]
24:     90  BOX  -10 -10 -10  20 0 0  0 20 0  0 0 20
```

A hexagon with LAT=2 is defined in the fifth line using the six surfaces defined in the seventeenth through twenty third lines. The hexagonal prism is restricted in the *y*-direction by -24 23 in the cell definition and is filled with the universe 2, i.e., water, as specified in the sixth line. The first cell has the repeated structure defined in universe 1. Figure 5.42 shows the result of this example. It can be seen that some prisms near the edges of the first cell, which is defined as a 12 cm cube, are only partly used. The directions of the lattice coordinate shown in the right panel depend on the order of the surface number written in the cell definition. To specify a cell using mesh=reg in a tally section, the lattice and universe styles (201 < 101[-2 0 0]), where the lattice coordinate is represented by [i j k], can be used: see Sec. 6.1.2 for more information on this format. The lattice coordinates can be shown using the [t-gshow] tally with output=7 or 8.

5.6.5 Repeated structures

There are several simple procedures in PHITS that can be used to create repeated structures in which the same or similar units are repeated. Using the lattice parameter described in Sec. 5.6.4 is one such method; another is the LIKE *n* BUT cell parameter format.

5.6.5.1 LIKE *n* BUT cell parameter

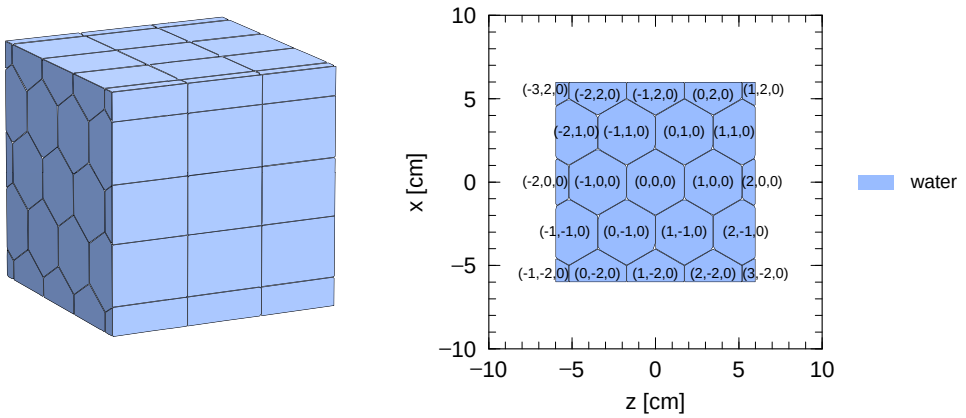


Figure 5.42: Result of `[cell]` section example (7) in 3D (left) and 2D (right) images.

Using this format, a cell that differs slightly from an original cell can be created. In this format, the cell parameters following the `BUT` statement differ from the those in the original cell (*n*). The cell parameters that can be used in this format are shown in Table 5.87. In the following example, two cell parameters, `TRCL` and `MAT`, are used.

Example 36: `[cell]` section example (8)

```
 1:  [ Material ]
 2:    mat[1]  1H 2  16O 1
 3:    mat[2]  Fe 1
 4:  [ Cell ]
 5:      1   0       -10  13 -14  #2 #3 #4
 6:      2   1  -1.0   11 -12  13 -14  15 -16
 7:      3   LIKE 2 BUT  TRCL=1
 8:      4   LIKE 2 BUT  TRCL=2  MAT=2
 9:      5  -1        #(-10  13  -14)
10:  [ Surface ]
11:    10  CY   10
12:    11  PX  -2
13:    12  PX   2
14:    13  PY  -2
15:    14  PY   2
16:    15  PZ  -2
17:    16  PZ   2
18:  [ Transform ]
19:  *tr1   3 0 -5
20:  *tr2   0 0  6  30 90 120  90 0 90  60 90 30  1
```

A 4 cm cube filled with water and placed at the origin of the coordinate system is defined in the sixth line. The interior of this cube is the second cell, which is regarded as the original cell in this example. In the seventh and eighth lines, respectively, the third and fourth cells are defined using the `LIKE` *n* `BUT` format with $n = 2$. Figure 5.43 shows the result of this example. The coordinate system of the third cell is transformed using the cell parameter `TRCL=1`, where the coordinate transform number 1 is defined in the nineteenth line of the `[transform]` section. The coordinate system of the fourth cell is transformed using `TRCL=2` and the interior material of the cell is replaced with iron defined using the material number 2 in the third line.

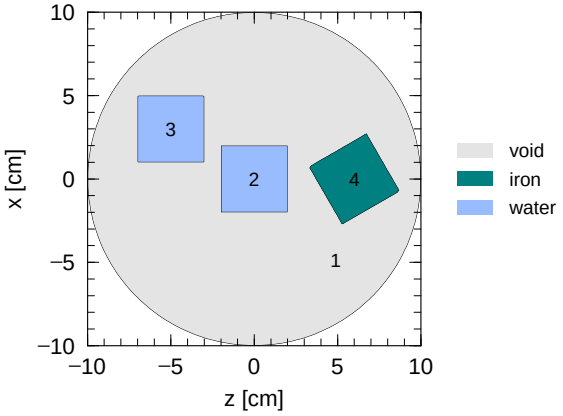


Figure 5.43: Result of `[cell]` section example (8).

5.6.5.2 Nesting structure with lattice

A nesting structure can be used based on the universe frame described in Sec. 5.6.3.. For example, universe 1 can be filled with universe 2, and universe 2 can be filled with universe 3, with this process continued following a user-defined nesting structure. The maximum number of nestings is 10, which corresponds to the parameter `mxlv` given in the file "param.inc."

In the next example, there are nine square poles defined using `LAT=1`; three of these have unique structures.

Example 37: `[cell]` section example (9)

```
 1:  [ Material ]
 2:    mat[1]  1H 2  16O 1
 3:    mat[2]  Fe 1
 4:  [ Cell ]
 5:      1   0        11  -12   13  -14  15  -16  FILL=1
 6:    101   0       -26   25  -22   21  LAT=1  U=1
 7:                  FILL=-1:1  -1:1  0:0
 8:                  2 2 3  2 3 2  3 2 2
 9:    201   1  -1.0  -90  U=2
10:    301   2 1-0.0  -10  U=3
11:    302   0        10  U=3
12:      2  -1       #1
13:  [ Surface ]
14:    10  CY   1.5
15:    11  PX  -6
16:    12  PX   6
17:    13  PY  -6
18:    14  PY   6
19:    15  PZ  -6
20:    16  PZ   6
21:    21  PX  -2
22:    22  PX   2
23:    25  PZ  -2
24:    26  PZ   2
25:    90  BOX  -10 -10 -10  20 0 0  0 20 0  0 0 20
```

The definition of the first cell in the fifth line and the lattice unit in the sixth line are the same as in the `[cell]` section example (6). However, the format of the cell parameter `FILL` in the seventh and eighth differs: in the seventh line, regions treated in this calculation are given in the lattice coordinate system. The numbers in the next line correspond to the universe number filling each lattice at (i, j, k) following the order $(-1,-1,0), (0,-1,0), (1,-1,0), (-1,0,0), \ldots, (1,1,0)$; in other words, a lattice at $(-1,-1,0)$ is filled with universe 2 and one at $(1,-1,0)$ is filled with universe 3. Universe 2 is defined in the ninth line as a space filled with water, while the universe 3 is defined in the tenth and eleventh lines has an iron cylinder centered at the origin. The result of this example is shown in Fig. 5.44, in which it can be seen that three lattices at $(1,-1,0), (0,0,0)$, and $(-1,1,0)$ have an iron cylinder. Specifying a cell using `mesh=reg` in the tally sections can be done using the lattice and universe styles as `(302 < 101[0 0 0])`, where the lattice coordinate is represented by `[i j k]`: see Sec. 6.1.2 for more information on this format.

A more complex example is shown below.

Example 38: `[cell]` section example (10)

```
 1:  [ Material ]
 2:    mat[1]  1H 2  16O 1
 3:    mat[2]  Fe 1
 4:  [ Cell ]
 5:      1   0        11  -12   13  -14  15  -16  FILL=1
 6:    101   0       -26   25  -22   21  LAT=1  U=1
 7:                  FILL=-1:1  -1:1  0:0
 8:                  2 2 3(1 0 1)  2 3(1 0 1) 2  3(1 0 1) 2 2
 9:    201   1  -1.0  -90  U=2
10:    301   0       -36   35  -32   31  LAT=1  U=3
11:                  FILL=-1:0  -1:0  0:0
12:                  4 2  2 4
13:    401   2 -10.0  -10  U=4
14:    402   0        10  U=4
```

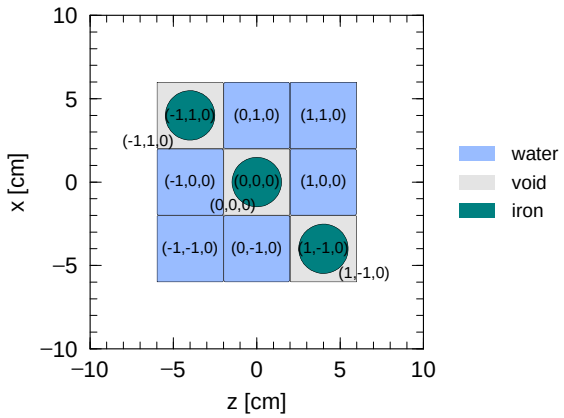


Figure 5.44: Result of `[cell]` section example (9).

```
15:       2  -1       #1
16:   [ Surface ]
17:     10  CY   0.5
18:     11  PX  -6
19:     12  PX   6
20:     13  PY  -6
21:     14  PY   6
22:     15  PZ  -6
23:     16  PZ   6
24:     21  PX  -2
25:     22  PX   2
26:     25  PZ  -2
27:     26  PZ   2
28:     31  PX  -1
29:     32  PX   1
30:     35  PZ  -1
31:     36  PZ   1
32:     90  BOX  -10 -10 -10  20 0 0  0 20 0  0 0 20
```

The virtual space formed by this input is shown in Fig. 5.45, in which there are nine square poles defined using the lattice parameter. Three of the poles comprise four units of the other lattice. In the eighth line, `(1 0 1)` denotes a transformation of the coordinate system in which the origin is shifted by 1 cm in both the *x*- and *z*-directions. The lattice and universe styles `(402 < 301[-1 -1 0] < 101[0 0 0])`, where the lattice coordinate is represented as `[i j k]`, can be used to specify any cell using `mesh=reg` in the tally sections: see Sec. 6.1.2 for more information on this format.

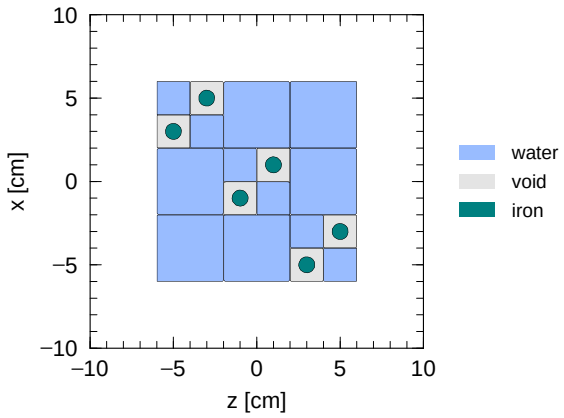


Figure 5.45: Result of `[cell]` section Example (10).

5.6.5.3 Voxel phantom

In PHITS, a virtual space can be created using voxel phantoms for calculation on complex structures such as the human body or an organism. To define a voxel phantom, a small cube must first be defined as the unit of a lattice with `LAT=1`. This unit is then repeated to define a large-size structure. Each unit can then be filled with a universe, which itself is filled with biological matter, e.g., compounds of carbon and water.

In the example below, a 10-cm cube comprising 125 ($5 \times 5 \times 5$) 2 cm cubes (voxels) is described.

Example 39: `[cell]` section example (11)

```
 1:   [ Material ]
 2:     mat[1]  1H 2  16O 1
 3:     mat[2]  Fe 1
 4:   [ Cell ]
 5:        1   0   11  -12   13  -14  15  -16  FILL=1
 6:     101   0  -20  LAT=1  U=1
 7:              FILL=-2:2  -2:2 -2:2
 8:              2 2 2 2 2  2 2 2 2 2  2 2 3 2 2  2 2 2 2 2  2 2 2 2 2
 9:              2 2 2 2 2  2 3 3 2 2  2 3 4 3 2  2 3 3 2 2  2 2 2 2 2
10:              2 2 2 2 2  2 3 3 3 2  3 4 4 4 3  2 3 3 3 2  2 2 2 2 2
11:              2 2 2 2 2  2 2 3 3 2  2 3 4 3 2  2 2 3 3 2  2 2 2 2 2
12:              2 2 2 2 2  2 2 2 2 2  2 2 3 2 2  2 2 2 2 2  2 2 2 2 2
13:     201   0        -90  U=2
14:     301   2 -10.0  -90  U=3
15:     401   1  -1.0  -90  U=4
16:       2  -1       #1
17:   [ Surface ]
18:     11  PX  -5
19:     12  PX   5
20:     13  PY  -5
21:     14  PY   5
22:     15  PZ  -5
23:     16  PZ   5
24:     20  BOX  -1 -1 -1  2 0 0  0 2 0  0 0 2
25:     90  BOX  -10 -10 -10  20 0 0  0 20 0  0 0 20
```

As the voxel unit, the 2 cm cube is defined in the twenty fourth line. The first cell, which is located within a 10-cm cube, has a repeated structure as defined in the fifth line. The region of the lattice coordinate space is determined in the seventh line. The order of voxels in the eighth through twelfth lines is as follows: $(-2,-2,-2),(-1,-2,-2),\ldots,(2,2,2)$, which represent the lattice coordinates. In the eighth through twelfth lines, `2` means universe 2, which is void, while `3` and `4` correspond to universes 3 and 4, which are of iron and water, respectively. Figure 5.46 shows the results of this example —a distorted iron box with regions of water within. To specify any cell using `mesh=reg` in tally sections, the lattice and universe styles `(401 < 101[0 0 0])`, where the lattice coordinate is represented by `[i j k]`, can be used: see Sec. 6.1.2 for more information. Note that formats such as, e.g., `(301 < 101[-2:2 -2:2 -2:2])` cannot be used because not all `101[-2:2 -2:2 -2:2]` cells have a 301st cell.

From the PHITS version 3.09, the array of universe numbers at the 8th through 12th lines of the example 39 can be expressed in a compressed format. For voxel phantoms, the universe array contains long ques of continuous same universe numbers and the compressed format is designed to reduce the size of this array. Thus the redution of the file size and computational time for read and write of the huge voxel data can be achieved. In the compressed format, the array of the continous universe numbers are given by the number of the continuous same universe with minus sign followed by the universe number. Following this rule, the array of universe number at the 8th through 12th lines of the example 39 can be rewritten as follows.

Example 40: `[cell]` section example (11')

```
 8:              2 -11 3 2 -17 3 -1 2 -2 3
 9:              4 3 2 -1 3 -1 2 -12 3 -2
10:              2 3 4 -2 3 2 3 -2 2 -12 3
11:              -1 2 -1 3 4 3 2 -2 3 -1 2
12:              -17 3 2 -11
```

Figure 5.46: Results of the [cell] section example (11) in 3D images. The structure in the right panel is equivalent to that in the left panel with part of its iron surface removed.

In the example 39 after the universe number 2 are aligned 12 times, the universe number 3 comes. On the other hand in the example 40, the second number -11, which comes after the first number 2, specifies that the universe number 2 given before was replicated 11 times afterword.

To enable save computational time, `ivoxel` can be specified in the `[parameters]` section. Performing PHITS calculation with `ivoxel` set =2 causes voxel data to be output in binary to `file(18)` and then stops the calculation. From the next calculation with `ivoxel` set =1, data output is omitted, which reduces the calculation time. If a very large amount of voxel data is used, it may be more convenient to use `infl`.

5.6.5.4 Use of tetrahedron geometry

In this section, a definition and the basic usage of tetrahedron geometry are given using an example. Tetrahedron geometry is a type of polygon mesh geometry that can express complex geometries by combining tetrahedrons of varying sizes. Note that the function of the reflection boundary does not work in tetra-mesh geometry. Instead by preparing a physically meaningless thin region in the tetra-mesh geometry and defining reflection boundary inside, the reflection boundary in tetra-mesh geometry can be realized.

PHITS adopts two tetrahedron geometry formats. One is the format for the tetrahedral mesh generator, TetGen and the other is the format for the finite element analysis program NASTRAN.

The TetGen format is represented by two files:

- the node file (a list of the xyz space coordinates of the tetrahedron nodes);
- the element file (a list of four nodes IDs for each tetrahedron element).

In PHITS, the node and the element files have a common name with different suffixes—".node" and ".ele," respectively: please refer to the ppt file or sample input file in `\phits\utility\TetraGEOM\` for a more detailed explanation.

Examples of a node file and an element file are given below.

Example 41: Example of node file

```
1:   # Simple two elements
2:   5   3  0  0
3:   #  pointID x y z
4:   1    0.0  -2.5  -4.0
5:   2   -4.0  -2.5   0.0
6:   3    4.0  -2.5   0.0
7:   4    0.0  -2.5   4.0
8:   5    0.0   2.5   0.0
```

The node file begins with the number of nodes and the number of dimensions (second line). PHITS operates only in three-dimensional geometry, and the two last zeros in the second line are not used in PHITS. The following lines (below the third line) represent a list of nodes and their xyz space coordinates in the order

```
[node no.] [x] [y] [z]
```

Example 42: Example of element file

```
1:   # Simple two elements
2:   2  4  1
3:   # elementID point(1:4) universe
4:   1     1  2  3  5    1001
5:   2     5  2  3  4    1002
```

The element file begins with the number of elements, the number of nodes comprising a tetrahedron (= 4 in PHITS), and the number of information points tagged to the element (=1 in PHITS) [second line]. The following lines (below the third line) give a list of 4-node IDs, each corresponding to a tetrahedron, in the order

```
[element no.] [node 1] [node 2] [node 3] [node4] [element universe no.]
```

The element no. represents the ID of the tetrahedron element. The node no. specifies the list of 4-node IDs in the node file. The element universe no. specifies the universe used to fill the tetrahedron element; the definition of this universe should be given in the PHITS input file (See Example 43). Arbitrary comment lines starting with # can be inserted into any line of the node or element file.

The node and element files can be created using the Tetrahedral Mesh Generator (TetGen) software, which can convert general polygon mesh data to tetrahedron geometry with some effort. TetGen can be obtained free of charge from the following source:

```
http://wias-berlin.de/software/tetgen/
```

For details on TetGen, please refer to the manual in the link above.

PHITS adopts another format, which is the bulk data format of NASTRAN. This format is a generally accepted in many software of structural analysis and computational fluid dynamics (CFD). PHITS uses the bulk data format defined using GRID and CTETRA and assumes those are written in a file with .bdf extention. The explantion of the buld data format is omitted here because the file will be deduced automatically from some software which is used to create the tetrahedron geometry. By adopting the same format for the tetrahedral geometry in PHITS as structural analysis or CFD, seamless coupled analysis between those studies and radiation transport calculation becomes possible. For the details, see a document uploaded in the folder `\phits\utility\FLUENT` which explains how to conduct such a coupled analysis.

The method for using tetrahedron geometry in a PHITS input file is explained below.

Example 43: `[cell]` section example (12)

```
 1:   [Material]
 2:   mat[1] 14N 78.1  16O 20.9  40Ar 0.93
 3:   mat[2] 1H 2  16O 1
 4:   mat[3] 56Fe 1
 5:   [Surface]
 6:   10  rpp   -5.0  5.0  -3.0  3.0  -5.0  5.0
 7:   20  rpp   -7.0  7.0  -5.0  5.0  -7.0  7.0
 8:   90  so     500.0
 9:   [Cell]
10:   101   1 -0.001205 -20  U=1  LAT=3  tfile=Tetra  TSFAC=1.0
11:     1   0           -10  FILL=1
12:     2  -1            10
13:   201   2 -1.0       -90  U=1001
14:   202   3 -7.874     -90  U=1002
```

Example 44: `[cell]` section example (13)

```
10:   101   1 -0.001205 -20  U=1  LAT=3  nfile=Tetra.bdf  TSFAC=1.0
```

To use tetrahedron geometry, a region must be defined as a rectangular shape (using the surface symbol RPP). This region should contain all of the nodes of the tetrahedrons that will be created but should not be so large that it incurs unnecessary computational costs. At the tenth line in the example code, a region with cell no. 101 is defined

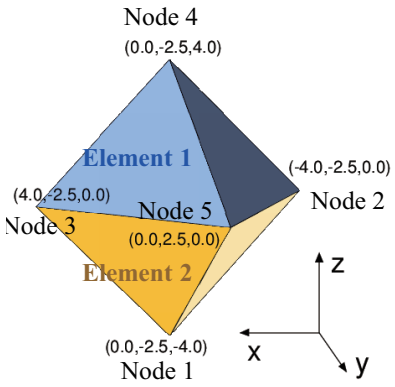


Figure 5.47: Results of the [cell] section example (12) in 3D images.

as a rectangular shape specified by surface number 20 as a 14 cm × 10 cm × 14 cm rectangular box. Setting the `LAT` option =3 declares the use of a tetrahedron geometry defined by the node and element files with names specified by `TFILE`. For a file of the NASTRAN bulk data format, the name should be specified accompanied by file `NFILE` as in the example 44. When specifying `NFILE`, if the file has a file extension (e.g., .bdf), please specify the file name including the extension. [91]. Note that the upper and lower cases of the file name differ between Mac and Linux. Using the `TSFAC` factor allows the size of the tetrahedron geometry to be scaled up or down by multiplying the coordinates of the nodes by the value of the factor. The material given in cell 101 is used to fill all of region 101 aside from the tetrahedron interiors. In the eleventh line the region with cell no. 1 is defined as a rectangular box of dimensions 10 cm × 6 cm × 10 cm. Using the `FILL` option, this region is filled by the tetrahedron geometry of cell no. 101. The Tetrahedron Geometry setting (`LAT=3`) should be used together with the universe and fill nest structures in the same manner as in lattice structure specification; the universes included in the element file should be defined in the same manner as in the lattice structure (this is done in lines 13 and 14 in this example, where element no. 1 is filled by material 2 (water), while element no. 2 is filled by material 3 (iron).

When the number of universes used in the tetrahedron geometry increases, the cost manually creating the cells done in lines 13 and 14 in the example 43 becomes heavy. By specifying the option `itetauto=1` in the parameter section, the cells corresponding to the universes for tetrahedron geometry will be automatically created. With this option, extreamly large surface no. 5000 and cells from no. 5001 to no. 5000+n are added to those given in the PHITS input file and thus use of those no. should be avoided to use this option, where n is number of universes used in the tetrahedron geometry. Density of the added cells are automatically defined by reading PSOLID and MAT tabs when a file with the NASTRAN bulk data format is used. For a TetGen format file, density information should be provided by an external file. The file name needs to be a common but with .txt suffix. The universe no. and its density should be specified for each line as 45. The addional cells will be provied with the same material no. as the cell. Error messages as shown in 46 will be displayed by executing PHITS without specifying the materials corresponding to the universes in the tetrahedron geometry. The materials from 5001 to 5000+n should be defined according to these messages. This automatic process can be verified by cheking the input echo lines in the phits.out file, where the additional surface and cells will be inserted.

Example 45: `[cell]` Example of txt file

```
1:   5001 5001 -1.0
2:   5002 5002 -7.874
```

Example 46: `[cell]` Example of error message

```
*** ERROR : undefined material
 TETRA material (MID):  1001
   should be defined as material ID number:  5001
*** ERROR : undefined material
```

[91] Before PHITS ver. 3.34, the filename should be specified without the extension such as .bdf.

```
 TETRA material (MID):  1002
   should be defined as material ID number:  5002
```

TRCL specification can not be used for the cell with LAT=3 option. To apply a coordinate transformation for the tetrahedron geometry, nest structure by universe and fill specificatgion is required and the TRCL option should be specified in the upper level above the cell containing the tetrahedron geometry with LAT=3. Please refer to the ppt file or sample input file in “/phits/utility/TetraGEOM” for a more detailed explantion.

5.7 [Transform] section

5.7.1 Formats

Coordinate transformation can be defined in this section. The transformation is available in `[source]`, `[surface]`, `[cell]` section, tallies with `r-z, xyz` mesh, and definition of magnetic and electro-magnetic fields. Only `$` and c␣ (␣ is a space) can be used as comment marks. Using `[transform]`, only rotation and translation of the coordinate can be set, and enlarge or shrink cannot be set.

Formats are shown below.

```
[ Transform ]
   TRn    X0   Y0   Z0   B1   B2   B3   B4   B5   B6   B7   B8   B9   M
```

Here, transform number is specified by n of Trn, X_0, Y_0, Z_0 are X-, Y-, and Z-components of the translation, respectively, and $B_i (i = 1, \cdots, 9)$ are input parameters to define the rotation. The last parameter M is an option which changes roles of B_i and equations of the coordinate transformation.

Table 5.88: Transformation parameter

Item	Explanation
n	Transform number. Use any number from 1 to 999,999.
X_0, Y_0, Z_0	x, y, z components of translation. The unit is cm.
$B_1 \sim B_9$	Input parameters to define rotation matrix. When $M = \pm 1$, please see Eqs. (20) and (21) in more detail. The determinant of the matrix is automatically adjusted to 1. When $M = \pm 2, \pm 3$, please see sections 5.7.2.2 and 5.7.2.3.
M	Option for coordinate transformation. When this value is positive, PHITS rotates the object in prior to translating it, if negative, PHITS translates the object in prior to rotating it. When $M = \pm 1$, components of the rotation matrix should be directly given as $B_i (i = 1, \cdots, 9)$. When $M = \pm 2$ or ± 3, only rotations around x, y, z-axes can be treated by using simple input. When $M = \pm 2$, the center of the rotation is the origin, and the order of the rotation is z, y, x. On the other hand, when $M = \pm 3$, a center coordinate of the rotation can be specified, and the order of the rotation can be changed. Rotation angles should be set in the unit of radian in the case of Trn format, while angles in degree should be set in the case of $*$TRn.

5.7.2 Mathematical definition and usage of transformation

When the coordinates before and after the transformation are defined as (x, y, z) and (x', y', z'), respectively, the mathematical definition is the following except the case of tallies:
for $M = 1$,

5.7.2.1 The case of $M = \pm 1$

When $M = \pm 1$, components of a rotation matrix can be directly given by $B_i (i = 1, \cdots, 9)$. The coordinates before and after the transformation are defined as (x, y, z) and (x', y', z'), respectively. The mathematical definition in the case of $M = 1$ is given as,

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_4 & B_7 \\ B_2 & B_5 & B_8 \\ B_3 & B_6 & B_9 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix}, \tag{20}$$

for $M = -1$,

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_4 & B_7 \\ B_2 & B_5 & B_8 \\ B_3 & B_6 & B_9 \end{pmatrix} \begin{pmatrix} x - X_0 \\ y - Y_0 \\ z - Z_0 \end{pmatrix}. \tag{21}$$

In the case of $M = 1$, PHITS rotates the object in prior to translating it. On the other hand, when $M = -1$, PHITS translates the object in prior to rotating it. The rotation is performed at the origin of the *xyz* coordinate system. Note that the direction of the translation setting $M = 1$ and -1 is opposite each other. For example, if you want to transform a cell to $+x$ direction by 10 cm, you have to set $X_0 = 10$ and -10 in the cases of $M = 1$ and -1, respectively. This rule is also applied to the cases of $M = \pm 2, \pm 3$.

Example 47: `[transform]` section example (1)

```
 1:  [ Transform ]
 2:  set: c10[90]  $ angle of around Z (degree)
 3:  set: c20[30]  $ angle of around Y (degree)
 4:  set: c30[0]   $ angle of around X (degree)
 5:
 6:  tr1   0 0 0
 7:        cos(c10/180*pi)*cos(c20/180*pi)
 8:        sin(c10/180*pi)*cos(c30/180*pi)+cos(c10/180*pi)*sin(c20/180*pi)*sin(c30/180*pi)
 9:        sin(c10/180*pi)*sin(c30/180*pi)-cos(c10/180*pi)*sin(c20/180*pi)*cos(c30/180*pi)
10:       -sin(c10/180*pi)*cos(c20/180*pi)
11:        cos(c10/180*pi)*cos(c30/180*pi)-sin(c10/180*pi)*sin(c20/180*pi)*sin(c30/180*pi)
12:        cos(c10/180*pi)*sin(c30/180*pi)+sin(c10/180*pi)*sin(c20/180*pi)*cos(c30/180*pi)
13:        sin(c20/180*pi)
14:       -cos(c20/180*pi)*sin(c30/180*pi)
15:        cos(c20/180*pi)*cos(c30/180*pi)
16:        1
```

In this example, `tr1` rotates the coordinate by `c10` degrees around *z* axis, `c20` degrees around *y* axis and finally `c30` degrees around *x* axis. By setting `c10, c20, c30`, the rotation can be defined.

5.7.2.2 The case of $M = \pm 2$

When $M = \pm 2$, only rotations around x, y, z-axes can be treated by using simple input. Input of only rotation angles is enough, and that of all componets of a rotation matirix is not needed. Note that you cannot change the order of the rotation, z, y, x.

Formats are shown below.

TR*n*	X_0	Y_0	Z_0	θ_z	θ_y	θ_x	0	0	0	0	0	0	2

X_0, Y_0, Z_0 are x, y, z components of translation, respectively. $\theta_z, \theta_y, \theta_x$ are rotation angles around z-,y-,x-axes, respectively. For B_4-B_9, any values, such as zero, should be set.

The direction of the rotation around each axis is shown in Fig. 5.48. Each rotation matrix is gives as,

$$R_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{pmatrix}, \quad R_y = \begin{pmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{pmatrix}, \quad R_z = \begin{pmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{22}$$

By using these matrixes, the mathematical definition in the case of $M = 2$ is given as,

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R_x R_y R_z \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix}, \tag{23}$$

for $M = -2$,

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R_x R_y R_z \begin{pmatrix} x - X_0 \\ y - Y_0 \\ z - Z_0 \end{pmatrix}. \tag{24}$$

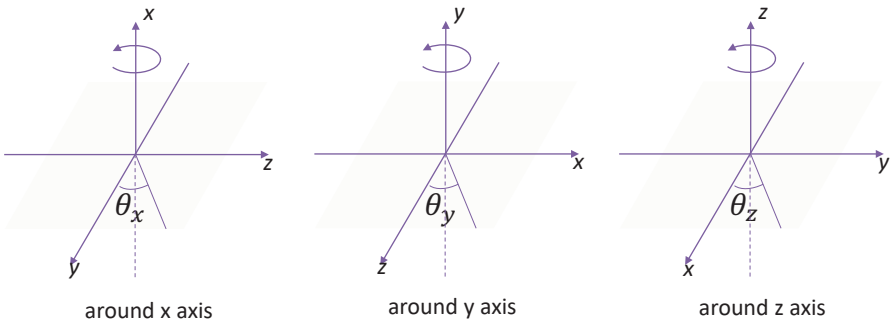


Figure 5.48: direction of rotation around x, y, z axis

Here, the coordinates before and after the transformation are (x, y, z) and (x', y', z'), respectively. Note that the direction of the translation setting $M = 2$ and -2 is opposite each other.

Example 48: `[transform]` section example (2)

```
1:   [ Transform ]
2:   *tr1   0 0 0  90   30   0  0 0 0 0 0 0  2
3:   tr2    0 0 0  pi/2 pi/6 0  0 0 0 0 0 0  2
```

This is an examples of $M = 2$, which gives the same definition of the rotation as Example 47. The rotation angles of ∗TR1 are given in units of degree, and those of TR2 are given in radian.

5.7.2.3 The case of $M = \pm 3$

By using this option, you can change a center coordinate of rotations and the order of rotation axes. Formats are shown below.

TR*n* X_0 Y_0 Z_0 X_C Y_C Z_C A_1 θ_1 A_2 θ_2 A_3 θ_3 3

Here, X_0, Y_0, Z_0 are x, y, z components of translation, respectively, and X_C, Y_C, Z_C are x, y, z components of the center coordinates for the rotation. A_i and $\theta_i (i = 1, 2, 3)$ are, respectively, rotation axis and angle of the i-th rotation. $A_i = 1, 2, 3$ mean, respectively, a rotation around x, y, z axes as the i-th rotation. On the other hand, $A_i = 0$ indicates that rotation is not necessary anymore. For example, if you want to rotate only around y-axis, you have to set $A_1 = 2$, and $A_2 = A_3 = 0$.

The coordinates before and after the transformation are defined as (x, y, z) and (x', y', z'), respectively. The mathematical definition in the case of $M = 3$ is given as,

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R_3 R_2 R_1 \begin{pmatrix} x - X_C \\ y - Y_C \\ z - Z_C \end{pmatrix} + \begin{pmatrix} X_C \\ Y_C \\ Z_C \end{pmatrix} + \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix}. \tag{25}$$

Here, R_1, R_2, R_3 are rotation matrixes around x, y, z axes specified by A_1, A_2, A_3, respectively. When $M = -3$,

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R_3 R_2 R_1 \begin{pmatrix} x - X_C - X_0 \\ y - Y_C - Y_0 \\ z - Z_C - Z_0 \end{pmatrix} + \begin{pmatrix} X_C \\ Y_C \\ Z_C \end{pmatrix}. \tag{26}$$

Note that the direction of the translation setting $M = 3$ and -3 is opposite each other.

Example 49: [transform] section example (3)

```
1:   [ Transform ]
2:   *tr1  0.0 0.0 5.0   0.0 10.0 0.0   2 45.0   3 -60.0   0 0.0   3
```

This is an examples of $M = 3$. PHITS rotates the object at the center of $(0, 10, 0)$ around y-axis by 45 deg and z-axis by -60 deg, and then translating along $+z$-axis by 5 cm.

5.7.3 The case of tally

The transformed tally displays the result in the coordinate system after the transformation, so-called body-fixed coordinate system. In this case, because the tally result is output in the coordinates system after the transformation, the relation between (x, y, z) and (x', y', z') in the equations in the case of tally is opposite to that in the other case. For example, when $M = 1$, the coordinates after the transformation (x', y', z'), which are tally coordinates, can be given as,

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_4 & B_7 \\ B_2 & B_5 & B_8 \\ B_3 & B_6 & B_9 \end{pmatrix} \begin{pmatrix} x - X_0 \\ y - Y_0 \\ z - Z_0 \end{pmatrix}. \tag{27}$$

Here, the coordinates before the transformation are (x, y, z). For example, an object at $x = 20$ cm in the original tally coordinate system is shifted to the position at $x' = 20 - X_0$ cm in the transformed tally. When X_0 is positive, the object is shifted by $-X_0$ along x-axis. On the other hand, when $M = -1$, the definition is given by,

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_2 & B_3 \\ B_4 & B_5 & B_6 \\ B_7 & B_8 & B_9 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} \tag{28}$$

The change in the cases of $M = \pm 2, \pm 3$ is the same as that in $M = \pm 1$.

5.8 [Temperature] section

Free-Gas Thermal Temperature [MeV] can be defined in this section. This section corresponds to TMP card but the time definition can be set. This value can be set in the `[cell]` section. If the temperature is double defined, temperatures defined in this `[temperature]` sections are used. If no definition, the default value 2.585×10^{-8}MeV is used. Note that in general temperature has an effect on only behavior of thermal neutrons, and has no effect on those of other particles.

```
    [ Temperature ]
        reg               tmp
          1             1.0*1.e-8
         11             5.0*1.e-8
( { 2 - 5 } 8 9 )       2.0*1.e-8
   ( 11 12 15 )         3.0*1.e-8
         16             6.0*1.e-8
        ....            ........
        ....            ........
```

The format `( { 2 - 5 } 8 9 )` can be used. However, any value that is not single numeric must be enclosed value by `(  )` .

The lattice and universe style can be used as `( 6 < 10[1 0 0] < u=3 )`.

To change the order of region number (`reg`) and temperature (`tmp`), set as `tmp reg`. The skip operator `non` can be used.

5.9 [Mat Time Change] section

By this section, you can change the material of certain cells to the other material as a function of time. This function is useful to describe a shutter of beam line, T0 chopper and the other devices for neutron optics. The unit of time is nsec.

```
[ Mat Time Change ]
             mat       time      change
               1       50.0       11
               2      100.0       12
               3     1000.0        0
             ....     ........
             ....     ........
```

In the above example, the material 1 is changed to material 11 at $t = 50.0$ nsec, 2 to 12 at 100 nsec and 3 to void at 1000 nsec. If you want to replace the order of the initial material (`mat`), time (`time`) and the final material (`change`), set as `mat change time`. You can use the skip operator `non`. These three columns are always necessary to define the mat time change function.

5.10 [Magnetic Field] section

5.10.1 Charged particle

A magnetic field can be set in the PHITS calculation. To use this function for electron and positron, set EGS5 mode.

Cell number (reg), magnetic field type (typ), half distance of magnets (gap)[cm], magnetic field intensity (mgf), transformation (trcl) and time dependence (time) should be defined in this section. Setting imagnf=1 in the [parameters] section enables simulation of particle transport in the magnetic field.

```
[ Magnetic Field ]
  reg              typ    gap            mgf           trcl     time
    1               4     10.00000      -5.956540        3     non
    2               4     10.00000       6.416140        1     non
    3               2     10.00000      -7.611980        0     0.0
    4               2     10.00000       3.516000        0     pi/2
  ( 150 < 61 )      4     13.00000       7.880140        2     non
  ( 150 < 62 )      4     13.00000      -7.440800        2     non
  ( 150 < 63 )      4     13.00000       9.441010        2     non
  ( 150 < 64 )      4     13.00000      -8.295220        2     non
  ( 150 < 65 )      4     13.00000       3.694830        2     non
  ( 150 < 66 )      4     13.00000      -2.099350        2     non
  ...              ...    ........       ........       ...     ...
  ...              ...    ........       ........       ...     ...
```

The column of trcl is omitted. The zero for trcl means no transformation. The time is a parameter of user defined time dependent magnetic field. The column of time is also omitted. The non for time means no time dependence. Two subroutines, "usrmgt1.f" and "usrmgt2.f" are included in the source as user defined subroutines for the time dependent magnetic field. The former is for Wobbler magnet, and the latter is for pulse magnet for neutron optics. These two subroutine are chosen by setting usrmgt=1, 2 in the [parameters] section. For the Wobbler magnet and pulse magnet, time means phase of the magnet and starting time, respectively.

In the above expression, reg is region number, typ can take 2 or 4 for dipole electromagnet, or quadrupole electromagnet, respectively. mgf denotes the strength of the magnetic field [kG], and trcl is the coordinate transformation number defined in the [transform] section.

The format ({ 2 - 5 } 8 9) can be used. However, any value that is not single numeric must be enclosed value by ( ) . The lattice and universe style can be used as (6 < 10[1 0 0] < u=3).

By using this format, the different magnetic field can be set for each lattice. If a cell is re-defined, the value, which is defined at first, is used.

In the case of dipole magnet, the distances gap make no sense, but set any numeric. The magnetic field is available not only in the void region, but also in the material where the normal reaction can be occurred.

z-axis is assumed to be the center of the magnetic field. The direction of the magnetic field is positive direction of *y*-axis for dipole, i.e., the positive charge particle is bent to positive direction of *x*-axis when it goes to positive direction of *z*-axis. For quadrupole, the positive particle is converged in *x*-axis, diverged in *y*-axis when it goes to positive direction of *z*-axis. The coordinate transformation by trcl is needed for different geometrical situation.

When specifying charge number of the projectile particle with izst in [source] section, the motion of the particle with the number in the magnetic field is described. Using izst, $PHITS$ can simulate the motion of the particle with charge states. The charge number defined with izst doesn't change while the particle moves. It should be noted that particles produced from nuclear reactions are not affected by the value of izst; the charge of the produced particle is given as its atomic number.

If you want to precisely calculate the trajectory of charged particles (except electrons and positrons) under a strong magnetic field that significantly bends their trajectories within a short distance (less than a few cm), set the maximum flight mesh size (deltg) in the magnetic field to a smaller value.

5.10.2 Neutron

The definition of the magnetic field for neutron is almost the same as for charged particles. Here we describe the detail of the magnetic field for neutron.

```
[ Magnetic Field ]
  reg    typ    gap           mgf          trcl     polar     time
    1     60    0.00000      35000.0          3       non      non
    2     61    0.00000      35000.0          1         1      non
    3    106    5.00000       7130.0          0         0      non
    4    104    0.00000        3.5           0       non      5.0
    5    102    0.00000        0.20          0       non      non
    6    101    3.00000       7130.0          2         1      non
    7    103    0.00000      35000.0          0        -1      non
   ...   ...    ........      ........       ...      ...      ...
   ...   ...    ........      ........       ...      ...      ...
```

We cannot take into account of the gravity nor additional dipole magnet. For `60` case, it is assumed that the spin always keeps parallel or anti-parallel to the magnet field. For `61` case, we solve the coupled equation of motion between the spin and the magnetic field. Then the spin flip can be occurred in the region with weak magnetic field. The strength of the magnetic field is specified in the unit of [T/m^2] in mgf column.

For the types above 100, we consider the coupled equations of the spin and the magnetic field. In addition, the effects of the gravity and additional dipole field can be taken into account. `106` is sextupole, `104` quadrupole, and `102` dipole, respectively. The strength of additional quadrupole magnet (z-direction) is give by the column of gap in the unit of [T].

For `101` type, the magnetic field is defined by the user program file, "usrmgf1.f." In this user program, the data measured by the neutron optics group in JAERI are read from the file and used the calculation. The strength of this field is renormalized by the value of `mgf`.

For `101` type, the magnetic field is also defined by the user program file, "usrmgf3.f." In this user program, there is a simple sextupole magnet field as same as in `106` type.

The neutron goes into the magnetic field with the initial spin if it is defined in the source section. If not, the initial spin is defined at the moment when the neutron goes into the magnetic field. The ratio of the number of parallel and anti-parallel spin to the magnetic field is determined by the polarization defined by the `polar` column. `non` in `polar` column means 0 polarization. The polarization is defined as

$$P = \frac{\phi_+ - \phi_-}{\phi_+ + \phi_-},$$

here, ϕ_+ and ϕ_- are the number of the parallel and anti-parallel particles.

5.10.3 Implementation of the magnetic field map

Since version 3.10, PHITS can read arbitrary magnetic fields given in the xyz or r-z grid data. The example of [magnetic field] for reading the magnetic field map is below:

```
[ Magnetic  Field ]
  reg   typ   gap    mgf   trcl   file
  101   -1    10.0   0.5   0      xyzlist.dat
  102   -2    100.0  3.0   0      rzlist.dat
  103   -3    10.0   10.0  0      xyzmap.dat
  104   -4    1.0    1.0   1      rzmap.dat
```

Four types of magnetic field maps can be read by PHITS, which are specified by typ=$-1, \cdots, -4$ for charged particles, and typ=$-101, \cdots, -104$ for neutrons. Only one magnetic field map can be defined for each type in an input file. The meanings of each type are shown in Table 5.89. (see Sec. 5.10.4 in more detail):

Table 5.89: Available types of magnetic field

typ	explanation
-1 or -101	xyz grid, data list type
-2 or -102	r-z grid, data list type
-3 or -103	xyz grid, data map type
-4 or -104	r-z grid, data map type

The meanings of gap and mgf are different from those for the case of the conventional magnetic fields, where gap is inversely proportional to the step size for calculating particle trajectory, i.e. setting the higher value for gap results in the smoother trajectory but longer computational time, while mgf indicates the normalization factor of the field strength for the magnetic field map cases. For example, you have to set mgf=10 when your magnetic field map is written in the unit of T because the unit of magnetic field in PHITS is kG. Note that electron and positron step sizes in the magnetic field are automatically determined irrelevant to gap. Name of the magnetic field map file is specified by file parameter. trcl indicates the transform number applied to the magnetic field map, though it does not work for electron and positron transport simulation in the same manner as the conventional magnetic field.

5.10.4 Format of magnetic field map file

The magnetic field map file should consist of the header and data parts. In the header part, comment remarks can be inserted using # and $ in similar to the PHITS input file. Examples of the magnetic field map files together with the PHITS input files to read them are included in `\phits\utility\magmap` folder.

5.10.4.1 Header format

In the header part, discrimination between lowercase and uppercase characters is not performed, and blank is ignored in the same as PHITS input. Only one parameter can be specified in one line. Table 5.90 shows the parameter list to be specified in the header part:

Table 5.90: parameter list in the header part

parameter	explanation
nx	number of x grid (only for typ=-1 & -3, not omissible)
ny	number of y grid (only for typ=-1 & -3, not omissible)
nz	number of z grid (only for typ=-1 & -3, not omissible)
nr	number of r grid (only for typ=-2 & -4, not omissible)
xmin	minimum value of x grid in cm (only for typ=-3, D=0)
xmax	maximum value of x grid in cm (only for typ=-3, not omissible)
ymin	minimum value of y grid in cm (only for typ=-3, D=0)
ymax	maximum value of y grid in cm (only for typ=-3, not omissible)
zmin	minimum value of z grid in cm (only for typ=-3 & -4, D=0)
zmax	maximum value of z grid in cm (only for typ=-3 & -4, not omissible)
rmin	minimum value of r grid in cm (only for typ=-4, D=0)
rmax	maximum value of r grid in cm (only for typ=-4, not omissible)
ibxmap	existence of the B_x map (yes=1, no=0) (only for typ=-3, D=1)
ibymap	existence of the B_y map (yes=1, no=0) (only for typ=-3, D=1)
ibzmap	existence of the B_z map (yes=1, no=0) (only for typ=-3 & -4, D=1)
ibrmap	existence of the B_r map (yes=1, no=0) (only for typ=-4, D=1)
extendx	extend the field to negative x (only for typ=-1 & -3, omissible)
extendy	extend the field to negative y (only for typ=-1 & -3, omissible)
extendz	extend the field to negative z (for all typ, omissible)
data	End of header.

If you want to flip the direction of magnetic field in the extended field, you have to write e.g. `flip=bx, bz`, where `bx` and `bz` indicate the direction of magnetic field to be flipped. For example, if you set:

```
extendx flip = bz
extendy flip = bz
```

the magnetic field for z-direction (B_z) is flipped as shown in the Fig. 5.49. Note that the magnetic fields for x and y directions are not flipped in this case.

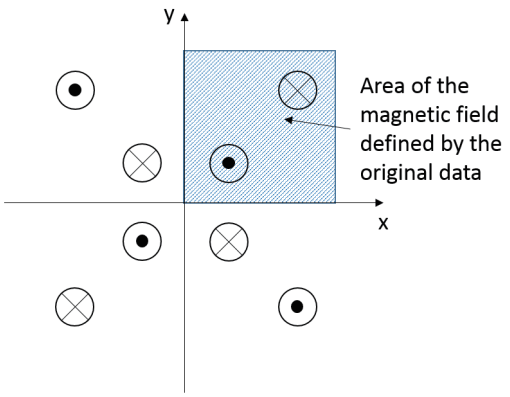


Figure 5.49: Example of the extended magnetic field with the `flip` command

5.10.4.2 Data format

In the data part, only numerical values with separation of comma, tab, or blank can be written. Comments cannot be inserted except for columns behind the last numerical data of each line. The units of the grid coordinate and the magnetic field are cm and kG, respectively. Note that the magnetic field strength at the point of each grid should be given instead of that at the center of each grid-mesh. The followings are the format of each field type:

- `typ`=-1: xyz grid, data list type

$$
\begin{array}{cccccc}
x_1 & y_1 & z_1 & B_{x_{1,1,1}} & B_{y_{1,1,1}} & B_{z_{1,1,1}} \\
x_1 & y_1 & z_2 & B_{x_{1,1,2}} & B_{y_{1,1,2}} & B_{z_{1,1,2}} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
x_1 & y_1 & z_{nz} & B_{x_{1,1,nz}} & B_{y_{1,1,nz}} & B_{z_{1,1,nz}} \\
x_1 & y_2 & z_1 & B_{x_{1,2,1}} & B_{y_{1,2,1}} & B_{z_{1,2,1}} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
x_1 & y_{ny} & z_{ny} & B_{x_{1,ny,nz}} & B_{y_{1,ny,nz}} & B_{z_{1,ny,nz}} \\
x_2 & y_1 & z_1 & B_{x_{2,1,1}} & B_{y_{2,1,1}} & B_{z_{2,1,1}} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
x_{nx} & y_{ny} & z_{ny} & B_{x_{nx,ny,nz}} & B_{y_{nx,ny,nz}} & B_{z_{nx,ny,nz}}
\end{array}
$$

- `typ`=-2: r-z grid, data list type

$$
\begin{array}{cccc}
r_1 & z_1 & B_{r_{1,1}} & B_{z_{1,1}} \\
r_1 & z_2 & B_{r_{1,2}} & B_{z_{1,2}} \\
\vdots & \vdots & \vdots & \\
r_1 & z_{nz} & B_{r_{1,nz}} & B_{z_{1,nz}} \\
r_2 & z_1 & B_{r_{2,1}} & B_{z_{2,1}} \\
\vdots & \vdots & \vdots & \\
r_{nr} & z_{nz} & B_{r_{nr,nz}} & B_{z_{nr,nz}}
\end{array}
$$

- `typ`=-3: xyz grid, data map type

$$
\begin{array}{ccccc}
B_{x_{1,1,1}} & B_{x_{1,2,1}} & B_{x_{1,3,1}} & \cdots & B_{x_{1,ny,1}} \\
B_{x_{2,1,1}} & B_{x_{2,2,1}} & B_{x_{2,3,1}} & \cdots & B_{x_{2,ny,1}} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
B_{x_{nx,1,1}} & B_{x_{nx,2,1}} & B_{x_{nx,3,1}} & \cdots & B_{x_{nx,ny,1}} \\
B_{x_{1,1,2}} & B_{x_{1,2,2}} & B_{x_{1,3,2}} & \cdots & B_{x_{1,ny,2}} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
B_{x_{nx,1,nz}} & B_{x_{nx,2,nz}} & B_{x_{nx,3,nz}} & \cdots & B_{x_{nx,ny,nz}}
\end{array}
$$

(continue the data for B_y and B_z in the same format)

- `typ`=-4: r-z grid, data map type

$$
\begin{array}{ccccc}
B_{r_{1,1}} & B_{r_{1,2}} & B_{r_{1,3}} & \cdots & B_{r_{1,nz}} \\
B_{r_{2,1}} & B_{r_{2,2}} & B_{r_{2,3}} & \cdots & B_{r_{2,nz}} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
B_{r_{nr,1}} & B_{r_{nr,2}} & B_{r_{nr,3}} & \cdots & B_{r_{nr,nz}}
\end{array}
$$

(continue the data for B_r in the same format)

The order of the data cannot be changed. However, you can omit to define the field strength for certain directions by specifying `ibxmap, ibymap, ibzmap, ibrmap` for `typ`=-3 & -4. In that case, the field strength for the omitted direction are assumed to be 0. For `typ`=-1 & -2, the grid coordinates should be given in the ascending order. The computational time for the data map types are generally shorter than that for the data list type, but they are nearly equivalent when the interval of each grid coordinate is constant.

5.11 [Electro Magnetic Field] section

Electric and magnetic fields in any region can be set. Defining parameters in this section and setting `ielctf`=1 in `[parameters]` section, PHITS can simulate a motion of a charged particle in the fields. To use this function for electron and positron, set EGS5 mode. Both the electric and magnetic fields can be set.

5.11.1 Uniform electric and magnetic fields

When use of uniform electric and magnetic fields, set the parameters as follows. It is noted that a quadrupole magnet cannot be defined, unlike the case of `[Magnetic Field]` section.

```
[ Electro Magnetic Field ]
  reg    elf    mgf    trcle    trclm
    1    100      1        1        2
    2    100      1        1        2
```

Cell number (`reg`), strength of the electric and magnetic fields (`elf` and `mgf`, respectively), direction of the two fields (`trcle` and `trclm`) should be defined. Units of `elf` and `mgf` are kV/cm and kGauss, respectively. The coordinate transformation number `trcl`, which are defined in `[transform]` section, should be set to `trcle` and `trclm`. When `trcle` is 0, the direction of the electric field is positive direction of *x*-axis. When `trclm` is 0, the direction of the magnetic field is positive direction of *y*-axis. `trcle` and `trclm` are not omissible. In the case that there is no need to When setting `elf` or `mgf` to 0, or not using the transformation, `trcle` and `trclm` should be set to 0.

When specifying charge number of the projectile particle with `izst` in `[source]` section, the motion of the particle with the number in the electro-magnetic field is described. Using `izst`, PHITS can simulate the motion of the particle with charge states. The charge number defined with `izst` doesn't change while the particle moves. It should be noted that particles produced from nuclear reactions are not affected by the value of `izst`; the charge of the produced particle is given as its atomic number.

If you want to precisely calculate the trajectory of charged particles (except electrons and positrons) under a strong electromagnetic field that significantly bends their trajectories within a short distance (less than a few cm), set the maximum flight mesh size (`deltg`) in the electromagnetic field to a smaller value.

5.11.2 Use of electric field map

Since version 3.20, PHITS can read arbitrary electro-magnetic fields given in the `xyz` or `r-z` grid data. Basic format for defining the electro-magnetic field map is the same as that for the magnetic field as described in Sec. 5.10.3, except for `filee` and `type` instead of `file` and `typ`, respectively, which specifies the file name and type of electric field. The magnetic field can co-exist with electric field by specifying its file name and type as `filem` and `typm`, respectively. The example of `[magnetic field]` for reading the magnetic field map is below:

```
[Electro Magnetic Field]
reg  type  typm  gap   elf    mgf  trcle  trclm     filee        filem
102  -3     -1   10.0  100.0  10.0   0      0    xyzmap.dat  xyzlist.dat
```

`type` and `typm` can be defined in the same manner as the type of magnetic field as given in Table 5.89. Note that type −101 to −104 cannot be used because electro-magnetic field in PHITS does not influence neutron trajectory. Only one electric or magnetic field map can be defined for each type in an input file. `elf` and `mgf` indicate the normalization factors of the field strength for the electric and magnetic field maps, respectively. When they are set to 1.0, the electric and magnetic field maps should be given in the unit of kV/cm and kG, respectively. `gap`

is inversely proportional to the step size for calculating particle trajectory, i.e. setting the higher value for gap results in the smoother trajectory but longer computational time. Note that electron and positron step sizes in the magnetic field are automatically determined irrelevant to `gap`. `trcle` and `trclm` indicate the transform number applied to the electric and magnetic field maps, respectively, though they do not work for electron and positron transport simulation in the same manner as the uniform electro-magnetic field.

The definition of the electric field map is the same as that of the magnetic field map except for the unit (kV/cm instead of kG). Please refer Sec. 5.10.4 “Format of magnetic field map file” for more detail.

5.12 [Delta Ray] section

In this section, you can set parameters used in the function to generate knocked-out electrons so-called δ-rays, which are produced along the trajectory of a charged particle in materials, as secondary particles. In the PHITS calculation, an energy transfer to the material is estimated by Linear Energy Transfer (LET; dE/dx), and is assumed to be deposited only on the particle trajectory. However, it is well known that owing to a high energy δ-ray the energy deposition is spread far away from the orbit of the primary particle. You can take the effect of δ-rays into account using this function. The production cross sections of δ-rays from those particles in liquid water were calculated using the model proposed by Butts and Katz, [92] considering the relativistic collision dynamics.

This function shouldn't be used together with `[t-sed]` tally.

You can set a threshold energy $E_{\rm th}$ (MeV) for each region except outer void to control the production of δ-rays. As the secondary particle, δ-rays with energies above $E_{\rm th}$ are explicitly generated and transported. For lower energies than $E_{\rm th}$, the deposition energies from δ-rays are included in LET. A minimum energy of $E_{\rm th}$ you can set is 0.001 MeV (= 1 keV). It is noted that in case of very low $E_{\rm th}$ or setting of a material thiner than 10 μg/cm^2, a behavior of the charged particle slightly changes. This is because the effective stopping power of the charged particles becomes smaller than its real value due to too many delta-ray productions. To calculate the delta-ray productions induced by charged particles with energies less than 1 MeV, set `emin(1)` and `emin(i=15-19)`=1 keV. A default value of $E_{\rm th}$ is $1.e+10$, i.e. δ-rays are not produced in the PHITS calculation except for setting the $E_{\rm th}$ parameter in this section. The region number and $E_{\rm th}$ are given by `reg` and `del`, respectively. Set these parameters as follows.

```
[ delta ray ]
    reg     del
      1     0.1
     11     1.0
    ....   ....
    ....   ....
```

You can use the format `( { 2 - 5 } 8 9 )`. But you need to close a value by `(  )` if it is not a single numeric value. You cannot use the lattice and universe style as `( 6 < 10[1 0 0] < u=3 )`. If you want to replace the order of region number (`reg`) and the threshold energy (`del`), set as `del reg`. You can use the skip operator `non`. Even if you use GG, use the symbol not `cell` but `reg` here.

[92] J. J. Butts and R. Katz, "Theory of RBE for Heavy Ion Bombardment of Dry Enzymes and Viruses", Radiation Research 30, 855-871 (1967).

5.13 [Track Structure] section

Performing the track-structure simulation [93] for electrons or positrons of low energy below 1 keV, those deceleration processes caused by much collision events for ionization, excitation, and molecular vibration and rotation can be calculated precisely. The track-structure simulation takes so long time that we cannot recommend to activate this mode in a conventional-scale particle transport simulation (cm orders) due to much production of low energy electrons around 10 eV.

In this section, you can specify the cells where track-structure simulation is performed. `mID` is the track-structure model specifier explained in Table 5.91.

Table 5.91: Track structure model specification by `mID`

`mID`		Electron, Positron	Proton 0-300 MeV Carbon 0-10 MeV/n	The other proton, and ion
−1	Track structure mode for arbitrary materials	chem=H20: ETS chem=Si: EST for Si Others: ETSART	ITSART	ITSART
−1	Track structure mode for arbitrary materials	ETS	ITSART	ITSART
0	(No track structure calculation)	– (EGS etc.)	– (ATIMA etc.)	– (ATIMA etc.)
1	Track structure mode for liquid water	ETS	KURBUC	ITSART

- –: No track structure mode (model used instead of track structure and is switched by `negs` or `ndedx`)
- ETS: electron-positron track structure mode dedicated to liquid water target
- KURBUC[94]: track structure mode for protons and carbon ions dedicated to liquid water target
- ITSART: generalized track structure mode for arbitrary materials and ions
- ETSART: electron-positron track structure mode for arbitrary tartgets
- ETS for Si: electron-positron track structure mode dedicated to semiconductor silicon target (PHITS-ETS for Si)[95]

When ETS or KURBUC is applied to materials other than liquid water, mean free paths of the particles in target materials are derived by scaling the cross section of liquid water by the electron density of the target material. The track structure mode for specific materials, activated by `mID=1`, is applicable to only liquid water but it simulates each reaction channel such as excitation and ionization accurately. In contrast, ITSART (the track structure model for arbitrary radiation and targets) invoked by `mID=-1` has following advantages

(1) applicable to arbitrary target materials

(2) considers atoms recoil by Rutherford scattering

[93] T. Kai et al., "Thermal equilibrium and prehydration processes of electrons injected into liquid water calculated by dynamic Monte Carlo method," Radiat. Phys. Chem., 115, 1-5 (2015).

[94] Uehara S, Nikjoo H, Goodhead DT, 1993, Cross sections for water vapour for Monte Carlo electron Track structure from 10 eV to 10 MeV region. Phys Med Biol 38: 1841-1858;

Hooshang Nikjoo, Shuzo Uehara, Dimitris Emfietzoglou, Interaction of Radiation with Matter, CRC Press, Published June 11, 2012;

Liamsuwan T, Uehara S, Emfietzoglou D and Nikjoo H 2011, Physical and biophysical properties of proton tracks of energies 1 keV to 300 MeV in water, Int. J. Radiat. Biol. 87 141-60;

Liamsuwan T and Nikjoo H 2013 A Monte Carlo track structure simulation code for the full-slowing-down carbon projectiles of energies 1 keV u^{-1} - 10 MeV u^{-1} in water, Phys. Med. Biol. 58 673-701

[95] Yuho Hirata, Takeshi Kai, Tatsuhiko Ogawa, Yusuke Matsuya and Tatsuhiko Sato, 2022, Implementation of the electron track-structure mode for silicon into PHITS for investigating the radiation effects in semiconductor devices, Japanese Journal of Applied Physics, 61, 106004.

at the cost of accuracy, such as an implicit treatment of target excitation. If the target material is included in the following list, the accuracy of the track structure mode for arbitrary materials can be improved by specifying the chemical form in the [material] section using chem parameter.
H_2O, CO_2, NH_2, NH_3, SF_6, TeF_6, CH_4, CH_3, C_2H_2, C_2H_4, C_2H_6, C_6H_6,$(CH_3)_2NH$

```
[track structure]
reg  mID
  1    1
  2    0
```

You can use the format (2 - 5 89). But you need to close a value by ( ) if it is not a single numeric value. You cannot use the lattice and universe style as (6 < 10[100] < u=3). If you want to replace the order of region number (reg) and index of the cross section database (mID), set as mID reg. You can use the skip operator non.

The followings are the important parameters for this mode. The parameters of etsmax and etsmin in [parameters] section are maximum and minimum energies of particles simulated by track-structure mode. In case of the use of the track-structure mode, the parameters of emin(12) and emin(13) should be set concurrently to 1.0e-3, and EGS5 should be activated (negs=1,2).

```
[ Parameters ]
emin(12) = 1.E-03
emin(13) = 1.E-03
negs     = 1
etsmax   = 1.E-2
etsmin   = 1.E-6
```

The parameter etsmax (D=1e-3 MeV/n) needs to be defined in [parameters] section. etsmax defines the maximum energies of proton and ions when the track structure mode is used. The energy lower bounds of track structure modes can be adjusted by emin(1) and emin(19) (D=1e-3 MeV/n) for protons and ions, respectively. If mID is 1 and etxmax is higher than the maximum energies of the track-structure cross section libraries, i.e. 300 MeV for proton and 10 MeV/n for carbon ion, PHITS performs the track structure simulation of protons and carbon ions by KURBUC up to these maximum energies. Above these energies and ions other than carbons and protons, PHITS uses ITSART.

```
[ Track structure ]
reg     mID     ebg     wvl
1       -1
2       -1      2.8     6.5
```

ETSART and Si-ETS are available from ver. 3.34. ETSART is activated by setting mID=-1. ETSART can simulate excitation reactions in semiconductor materials by entering the bandgap energy. The [track structure] section defines the bandgap energy as ebg [eV]; when ebg is undefined, ebg=0 by default, and excitation reactions are ignored. The energy required to produce one electron-hole pair in a material (w-value) can be defined as wvl [eV] in [track structure]. When the electrons reach the cutoff energy (etsmin), the number of generated electrons is calculated by dividing etsmin by the w-value and tallied in [t-interact]. This function allows us to efficiently calculate the number of electrons produced by relatively high-energy radiation by setting etsmin to a high value. In the [material] section, if chem is specified and mID=-1, electron track structure analysis can be performed for that material. Currently, ETS can be performed with chem=H2O and Si-ETS with chem=Si.

5.14 [Super Mirror] section

The reflection of low energy neutron by super mirror is defined by this section. We assume the following empirical formula to describe the reflectivity of the super mirrors.

$$R = \begin{cases} R_0 & \text{if } Q \le Q_c \\ \frac{1}{2}R_0\,(1-\tanh\left[(Q-mQ_c)/W\right])\,(1-\alpha(Q-Q_c)) & \text{if } Q > Q_c \end{cases}$$

where Q is the scattering vector (in Å^{-1}) defined by

$$Q = |\mathbf{k}_i - \mathbf{k}_f| = \frac{4\pi \sin\theta}{\lambda}.$$

The value of m is a parameter determined by the mirror material, the bilayer sequence and the number of bilayers. Q_c is the critical scattering wave vector for a single layer of the mirror material. At higher values of Q, the reflectivity starts falling linearly with a slope α until a cutoff at $Q = mQ_c$. The width of the cutoff is denoted W.

These parameters are defined as

```
[ Super Mirror ]
   r-in          r-out    mm   r0    qc       am    wm
   {2001-2020}    3001     3    0.99  0.0217   3.0   0.003
   2500           3500     3    0.99  0.0217   3.0   0.003
   2600           3600     3    0.99  0.0217   3.0   0.003
   ....            ....     ..   ...   ....     ...   ...
   ....            ....     ..   ...   ....     ...   ...
   ....            ....     ..   ...   ....     ...   ...
```

The reflection surface is defined by the surface between `r-in` and `r-out`. You can use the format `( { 2 - 5 } 8 9 )`, and you can use the lattice and universe style as `( 6 < 10[1 0 0] < u=3 )` in these definitions. The remaining parameters in above expression denote m by `mm`, R_0 by `r0`, Q_c by `qc` in Å^{-1}, α by `am` in Å, and W by `wm` in Å^{-1}.

We restrict this function only to neutrons for the case that its energy is less than 10 eV or sinθ is greater than 0.001, the latter is due to roughness of the surface.

5.15 [Elastic Option] section

In this section, you can set some parameters for user defined elastic option for low energy neutrons. By this function, you can change angle distributions of elastic collisions of data based neutron reactions. We prepare two sample routines, “usrelst1.f” and “usrelst2.f.” You can choose one of these two by usrelst=1, 2 in the parameter section. You should define the regions to which this function is applied and 4 parameters as,

```
[ Elastic Option ]
             reg       c1       c2       c3       c4
               1        5        1      3.3      0.4
               2        1        1      1.1      0.7
               3        3        1      0.3      0.8
             ....     ...      ...      ...      ...
             ....     ...      ...      ...      ...
```

If you want to replace the order of region number (reg), (c1 c2 c3 c4), set as reg c3 c2 c1 c4. You can use the skip operator non. You can use the format { 4 - 7 }, but the ({ 4 - 7 } 9 10) format cannot be used.

The sample routine of “usrelst1.f” is for Bragg scattering based on the data base, and “usrelst2.f” for any type of angular distribution described by an analytic formula.

5.16 [Data Max] section

dmax(i), which is the maximum energy of library use for i-th particle, can be defined for each nucleus in materials in this section. Maximum 6 [Data Max] sections are allowed to be used in a input file.

```
[Data Max]
  part = neutron proton
  mat   nucleus  dmax
   all   Fe      20
    5    all     50
    3    56Fe    150
```

Particle is defined in the first line as part=. From version 3.27, the upper limits of the nuclear data libraries for each nuclide are automatically determined from the address file (xsdir) coupled with dmax(i) and lib(i), where i is the particle type. Therefore, it is not necessary to specified [data max] section by users if they want to use all high-energy nuclear data libraries such as JENDL-4.0/HE defined in their xsdir file.

Three columns, mat, nucleus, and dmax, can be used. If you want to change the order of (mat) (nucleus), set as nucleus mat. You can use the skip operator non. In the mat column, material numbers can be specified and all means all materials. In the nucleus column, you can specify nucleus as 56Fe and 26056 type. You can use Fe or 26000, which specifies all isotopes of Fe. You can also use all. By the number (MeV) in the dmax column, you define the maximum energy of library use for the nucleus.

If the same nucleus is defined in the [data max] sections, the latest definition has priority in an input file.

The values of dmax(1) and dmax(2) defined by [parameters] section should be the maximum value in [data max] section.

When dmax is set to higher than the upper limit of the library energy, PHITS simply uses the cross section for the highest energy up to dmax. For example, when dmax is set to 200 in the case of using the default library (JENDL-4.0), the cross section at 20 MeV is continuously used up to 200 MeV. When you use two (or more) libraries have different upper limits from each other, you should set the limit for each nucleus in this section.

If kmout=1 is specified in [parameters] section, the values of dmax for each nucleus in the materials are shown in the output file, file(6) (D=phits.out).

5.17 [Frag Data] section

The function to read user defined cross sections can be defined in the `[frag data]` section. By using this function, you can simulate nuclear reaction events in the calculation on the basis of information given by the cross section data. It is noted that nuclear data libraries are preferentially used, even in the case of a nuclear reaction you specify in this section. You can set only proton, neutron, and nuclei as incident particles. Photons and elections cannot be used. Elastic scattering events for nucleons are separately considered by the default model implemented in PHITS.

Because weight values of the secondary particles produced by this option change from 1.0, you should decrease the value of the cutoff weight parameter `wc2(i)` in the `[parameters]` section.

Reaction cut-off energy `cmin(i)` should be set to the minimum incident energy of the defined cross sections.

Set the parameters as follows.

```
[ Frag Data ]
  opt     proj     targ     file
    0     12C      16O      DDX_12C-16O.dat
    1     proton   63Cu     DDX_p-63Cu.dat
```

The user defined cross sections are not used when `opt=0`. In the case of `opt=1`, PHITS reproduces the cross section of the reaction between an incident particle and target, which are specified by `proj` and `targ`, respectively, using the data. Options when `opt=2,3` are under construction. When `opt=4`, PHITS simply extrapolates the given data for incident energies, emission angles, and emission energies. Note that you can use this option only in the case of neo>0 and nag≠0, which will be explained below. When differential cross sections are given as pointwise data, set `opt=5`. In this case, the energy and angular dependence of the data are calculated by linear interpolation.

Format of the data of the user defined cross sections is as follows.

```
projectile
target
nei
ein(1)    ein(2)    ein(3)    ..... ein(nei+1)
totxs(1)  totxs(2)  totxs(3) ..... totxs(nei+1)
neo
eout(1)  eout(2)  eout(3) ..... eout(neo+1)
nag
angle(1)  angle(2)  angle(3) ..... angle(nag+1)
nfrg
frag(1)   frag(2)   frag(3) ..... frag(nfrg)

proxs(1,1)   proxs(1,2)   proxs(1,3) ..... proxs(1,nfrg)
ddx(1,1,1,1)   ddx(1,1,1,2)   ddx(1,1,1,3) ..... ddx(1,1,1,nag)
ddx(1,1,2,1)   ddx(1,1,2,2)   ddx(1,1,2,3) ..... ddx(1,1,2,nag)
      .........
ddx(1,1,neo,1)   ddx(1,1,neo,2)   ddx(1,1,neo,3) ..... ddx(1,1,neo,nag)

ddx(1,2,1,1)   ddx(1,2,1,2)   ddx(1,2,1,3) ..... ddx(1,2,1,nag)
ddx(1,2,2,1)   ddx(1,2,2,2)   ddx(1,2,2,3) ..... ddx(1,2,2,nag)
      .........
ddx(1,2,neo,1)   ddx(1,2,neo,2)   ddx(1,2,neo,3) ..... ddx(1,2,neo,nag)

      .........
      .........
```

(continued)

ddx(1,nfrg,1,1) ddx(1,nfrg,1,2) ddx(1,nfrg,1,3) ..... ddx(1,nfrg,1,nag)
ddx(1,nfrg,2,1) ddx(1,nfrg,2,2) ddx(1,nfrg,2,3) ..... ddx(1,nfrg,2,nag)
. . . . . .
ddx(1,nfrg,neo,1) ddx(1,nfrg,neo,2) ddx(1,nfrg,neo,3) ..... ddx(1,nfrg,neo,nag)

proxs(2,1) proxs(2,2) proxs(2,3) ..... proxs(2,nfrg)
ddx(2,1,1,1) ddx(2,1,1,2) ddx(2,1,1,3) ..... ddx(2,1,1,nag)
ddx(2,1,2,1) ddx(2,1,2,2) ddx(2,1,2,3) ..... ddx(2,1,2,nag)
. . . . . .
ddx(2,1,neo,1) ddx(2,1,neo,2) ddx(2,1,neo,3) ..... ddx(2,1,neo,nag)

. . . . . .
. . . . . .
. . . . . .
. . . . . .

proxs(nei+1,1) proxs(nei+1,2) proxs(nei+1,3) ..... proxs(nei+1,nfrg)
ddx(nei+1,1,1,1) ddx(nei+1,1,1,2) ddx(nei+1,1,1,3) ..... ddx(nei+1,1,1,nag)
ddx(nei+1,1,2,1) ddx(nei+1,1,2,2) ddx(nei+1,1,2,3) ..... ddx(nei+1,1,2,nag)
. . . . . .
ddx(nei+1,1,neo,1) ddx(nei+1,1,neo,2) ddx(nei+1,1,neo,3) ..... ddx(nei+1,1,neo,nag)

. . . . . .
. . . . . .

ddx(nei+1,nfrg,1,1) ddx(nei+1,nfrg,1,2) ddx(nei+1,nfrg,1,3) ..... ddx(nei+1,nfrg,1,nag)
ddx(nei+1,nfrg,2,1) ddx(nei+1,nfrg,2,2) ddx(nei+1,nfrg,2,3) ..... ddx(nei+1,nfrg,2,nag)
. . . . . .
ddx(nei+1,nfrg,neo,1) ddx(nei+1,nfrg,neo,2) ddx(nei+1,nfrg,neo,3) ..... ddx(nei+1,nfrg,neo,nag)

In the beginning, you specify incident particle and target in this file. nei is the number of points of the energy mesh. In the next line, you set nei+1 points of incident energies (ein in the unit of MeV/n), and then in the next next line you set neo+1 data of total reaction cross sections (totxs in the unit of mb). Note that when totxs=0, total reaction cross sections obtained by models, which are specified by `icxsni` or `icrhi`, are used. neo is the number of energy mesh points of the outgoing particles. Note that set neo data points of eout when `opt=5`. In the next line, you set neo+1 points of the energy (eout in the unit of MeV/n). nag is the number of angular mesh points for the outgoing particles. You set nag+1 data of the angles in the next line. Note that set nag data points of the angles when `opt=5`. If nag>0 the data should be given in the unit of radian, and if nag<0 those are in the unit is degree. When nag=0, isotropic is assumed. nfrg is the number of the outgoing particles. In the next line, you write nfrg names of the particles. proxs (in the unit of mb) is the production cross sections of the particles at an incident energy, and then you set neo×nag data of the double differential cross sections (ddx in the unit of mb/MeV/sr). You have to specify nei+1 groups of the proxs and ddxs per an outgoing particle.

If you write `model` in the line where you should write neo, nuclear reaction models are used for simulating nuclear reaction events. In this case, you do not have to write the data below neo.

Example 50: example of data file of user defined cross sections

```
1:   proton
2:   63Cu
3:   1
4:   10.0    100.0
5:   1000.0  500.0
6:   3
7:   1.0  10.0  50.0  100.0
```

```
 8:   -6
 9:   0.0  30.0  60.0  90.0  120.0  150.0  180.0
10:   1
11:   neutron
12:   300.0
13:   10.0  10.0  10.0  10.0  10.0  10.0
14:   15.0  13.0  12.0  11.0  10.0  10.0
15:   10.0  11.0  10.0  11.0  10.0  10.0
16:   0.0
17:    5.0   5.0   5.0   5.0   5.0   5.0
18:   10.0   8.0   7.0   6.0   5.0   5.0
19:    5.0   6.0   5.0   6.0   5.0   5.0
```

An example of the data file of the user defined cross sections is shown in Example 50. In first and second lines, proton and ^{63}Cu are defined as incident particle and target, respectively. In third line, nei=1 is set. Tow data of incident energies and total reaction cross sections are given in fourth and fifthe lines, respectively. In sixth line neo=3 is set, and then 4 values of 1.0, 10.0, 50.0, and 100.0MeV are given as energies of the outgoing particles. The number of angular mesh points nag=-6 is set in eighth line. In this case, the angles are given in the unit of degree. In tenth and eleventh lines, the number of particles and its kind (neutron) are specified, respectively. In twelfth line, a production cross section of neutron at the 10.0MeV proton is given. The double differential cross sections of the neutron are set in thirteenth, fourteenth, and fifteenth lines. Each lines correspond to the energy bins defined in seventh line, and each columns correspond to the angular bins defined in ninth line. The production cross section is used as a normalization factor. When proxs=0 (mb), the integrated values of the ddxs are used as proxs.

When neo is negative, you can set the intensity of the cross section discretely. In this case, the number of eout should be neo.

If neo is 0, the energy spectra of the outgoing particle can be given by Gaussian distribution. In the places of ddxs, you set mean value and FWHM of the Gaussian in the unit of MeV.

The following abbreviation is available when many zeros are included in ddxs.

```
0 -9
```

This description means total ten zeros.

5.18 [Importance] section

The importance for cell can be defined in this section. If the importance is not defined, it is set as `1.0`.
Maximum 6 `[importance]` sections are allowed to be defined in an input file.

```
      [ Importance ]
         part =  proton  neutron
          reg               imp
            1              1.000000
           11              5.000000
   ( { 2 - 5 } 8 9 )       2.000000
      ( 11 12 15 )         3.000000
   ( 6<10[1 0 0]<u=3 )     6.000000
          ....             ........
          ....             ........
```

Particle is defined as `part=` at the first line. If the `part` is not defined, default value is defined as `part=all`. The format to describe particles is the same as in tally definition. However, it can distinguish ityp only, each nucleus is not specified.

If you want to change the order of region number (`reg`) and (`imp`), set as `imp reg`. You can use the skip operator `non`.

You can use the format like `( { 2 - 5 } 8 9 )`, and you can use the lattice and universe style as `( 6 < 10[1 0 0] < u=3 )`. But you need to close a value by `( )` if it is not a single numeric value. The importance of bottom level is a product by each importance at each level. In PHITS, importance of a specific cell at bottom level can be defined by above format. By using the format, we can define different importance into each lattice. If the importance is double-defined, the first defined importance is valid.

If you set large importance to particles which have strong penetration through matter such as neutrino, PHITS calculation takes time too much. If you define part=all, neutrino is included. You must give attention about it.

Some rules can be used to define an importance of a cell in a repeated structures and lattices. For example, cells 5, 6, and 7 on a bottom level are included by cells 11, 12, and 13 on upper level, we can define the importance as

```
1:   [ Importance ]
2:      reg            imp
3:    ( 5 6 7 < 11 )   2.0
4:    ( 5 6 7 < 12 )   4.0
5:    ( 5 6 7 < 13 )   8.0
6:    ( 11 12 13 )     1.0
```

or

```
1:   [ Importance ]
2:      reg            imp
3:    ( 5 6 7 )        1.0
4:      11             2.0
5:      12             4.0
6:      13             8.0
```

Above two definitions give same results, but in the latter case, the importance for cells 5, 6, and 7 are displayed as 1.0 at the importance summary.

5.19 [Weight Window] section

The weight window function can be defined in this section. Maximum 6 [weight window] sections are allowed to be defined in a input file.

```
    [ Weight Window ]
      mesh =  reg
      part =  proton  neutron
       eng =  5
     ( tim =  5 )
              6.00e-7  3.98e-1  1.00e+0  7.00e+0  5.00e+4
       reg               ww1       ww2       ww3
         1             0.010000  0.100000  0.001000
        11             0.005000  0.050000  0.000300
( { 2 - 5 } 8 9 )      0.001000  0.010000  0.000100
   ( 11 12 15 )        0.000500  0.005000  0.000030
( 6<10[1 0 0]<u=3 )    0.000010  0.001000  0.000010
       ....            ........  ........  ........
                         ww4       ww5
                       0.010000  0.100000
                       0.005000  0.050000
                       0.001000  0.010000
                       0.000500  0.005000
                       0.000010  0.001000
                       ........  ........
```

Mesh type should be defined in the 1st line, i.e. mesh=reg, xyz or tet. When mesh is not defined, the default setting mesh=reg is set. For mesh=xyz, x-type, y-type, and z-type must be defined in subsequent lines (see Sec. 6 in detail). Note that the surfaces of xyz-mesh should be slightly shifted from surfaces of cells. Please be especially careful if you are using parameters to link geometry boundary and xyz-mesh ranges. After version 3.34, if the maximum and minimum xyz mesh values for [weight window] and [t-wwg] are defined as integers, the minimum value is shifted by $-1*$deltxyz and the maximum value by $+2*$deltxyz from the integer (see FAQ 2.8). The mesh optionmesh=tet can be only used for geometries containing tetrahedral-mesh geometry. The cell number specification of the rectangular box where the tetrahedral-mesh geometry is implemented needs to be followed as reg=. The weight window for each tetrahedron element can be specified. Particle type should be defined as part=. part=all means all particles. The format to describe particles part= is the same format as in tally definition. However, it can distinguish ityp only, each nucleus is not specified.

Next you define the energy mesh or time mesh. First, you define the number of mesh by eng= or tim= and, in next line, the values of each mesh ($e_1, e_2, e_3, ...$). Minimum value of weight window for each mesh should be defined in the followings. Each minimum values are like ww1, ww2, ww3, ... where wwi is a window minimum value for a mesh $e_{i-1} < E < e_i$. $e_0 = 0$ and $t_0 = -\infty$ is assumed. If there exists no eng= / tim= definitions, energy / time mesh are not prepared. In this case, you should set only ww1.

Region (reg, xyz or tet) must be written at the first column. As above example, you can make another table for wwi definitions. From second table, the region definition can be skipped as the example. You can use the skip operator non in this section.

For mesh = reg, you can use the format ({ 2 - 5 } 8 9), and you can use the lattice and universe style as (6 < 10[1 0 0] < u=3). But you need to close a value by ( ) if it is not a single numeric value. For mesh = xyz, the position of each data should be defined as (ix iy iz). For mesh = tet, minimum values of weight window for all tetrahedron elements needs to be specified following to the element number.

If you set large weight window to particles which has strong penetration through matter such as neutrino, PHITS calculation takes time too much. If you define part=all, neutrino is included. You must give attention about it.

5.20 [WW Bias] section

In the [ww bias] section, minimum values of the weight window defined in [weight window] can be biased for certain regions. This option is useful when the [weight window] parameters for a certain region are biased after automatically generating the [weight window] section by [t-wwg]. Figure 5.50 shows a flowchart how to perform the transport calculation using [weight window] and [ww bias]. First, generate the [weight window] section by [t-wwg]. If the obtained parameters of [weight window] are enough to effectively use the variance reduction technique, the [ww bias] section is not needed. However, if they are not enough, the variance reduction technique can be used more effectively by biasing the [weight window] parameters for a certain region with [ww bias]. There are two methods to set the [ww bias] section; an automatic method using [t-wwbg], which is a Weight Window Bias Generator (WWBG), and a manual method by a user, which will be explained below. For details of [t-wwbg], see Sec. 7.17. When performing a transport calculation with [ww bias] and [weight window], set icntl=0 and iwwbias=1 in [parameters].

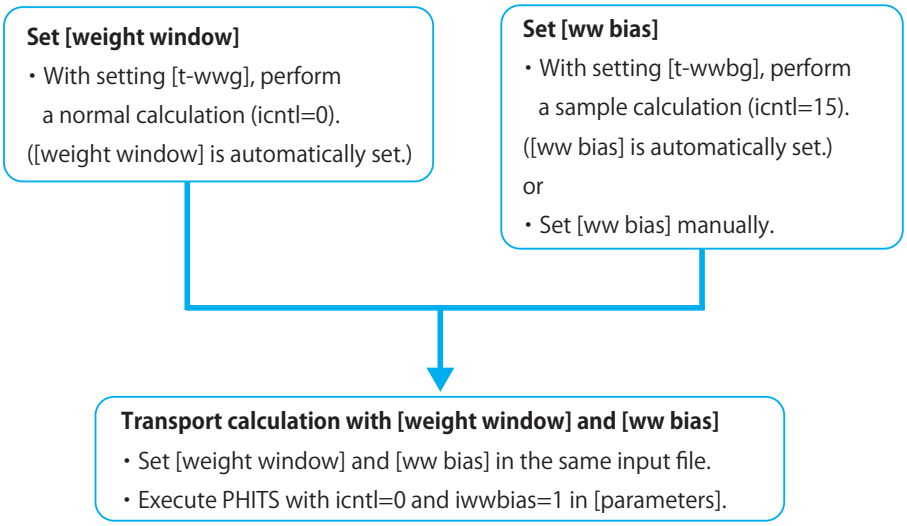


Figure 5.50: The flowchart of the connection calculation between [weight window] and [ww bias].

When more than one [weight window] section is defined, the combinations of [weight window] and [ww bias] are chosen in the order of appearance in the input file. Therefore the same number of [ww bias] sections are required as the number of [weight window].

The format of [ww bias] is as follows: (Note that set the same particle, energy-mesh, and regions as [weight window].) mesh=reg will be selected as default if the mesh specification is skipped. When mesh=xyz is specified, x-type, y-type, and z-type must be defined in subsequent lines same as [weight window]. When mesh=tet is specified, the same cell number specification by reg= is required as [weight window].

```
[ WW Bias ]
   part =  neutron
    eng =  2
           1e-3   1.0
    reg    wwb1       wwb2
      1    0.25       0.25
      2    0.50       0.50
      3    1.00       1.00
      4    2.00       2.00
    ....   ........   ........
```

In the first line, part= defines which particle is to be considered. When it is omitted, part=all is set. The

expression of `part=` is the same as that in the tally format. Note that only the expression as `ityp` can be set. Each nuclides cannot be specified. Next, the energy mesh should be defined. The line starting with `eng=` specifies the number of mesh. In the next line, energies ($e_1, e_2, e_3, ...$) are defined. Furthermore, names of columns are gives as `reg, wwb1, wwb2, ....` In the `reg` column, the cell numbers are written. The bias values are given in the columns of `wwb`*i*. The skip operator `non` can be used. Each `wwb`*i* column corresponds to energies of $e_{i-1} < E < e_i$. Here, $e_0 = 0$. The format `( { 2 - 5 } 8 9 )` can be used, as can the lattice and universe style `( 6 < 10[1 0 0] < u=3 )`. However, any value that is not single numeric must be enclosed value by `( )` .

By setting `iwwbias=1` in `[parameters]`, the `[weight window]` parameters multiplied by inverse of the defined biases in `[ww bias]` are used. In this case, the products of the multiplication are output in the input echo of `[weight window]`, and `[ww bias]` with `off` is output. If an input file without `[ww bias]` is used, all values of `[ww bias]` in the input echo are set to 1.

An example of `[ww bias]` is as follows.

Example 51: Example of `[ww bias]`

```
 1:   [ WW Bias ]
 2:      part = neutron
 3:       eng =  1
 4:            1.00000E+05
 5:
 6:      reg       wwb1
 7:      1         1/7
 8:      2         1/6
 9:      3         1/5
10:      4         1/4
11:      5         1/3
12:      6         1/2
13:      7         1
14:      8         2
15:      9         3
16:      10        4
17:      11        5
18:      12        6
```

Here, neutron is considered as `part`. One energy region of 100 GeV below is specified. Regions between 1 and 12 are gradually biased. The region of the large number is biased stronger than that of the small number.

Figure 5.51(a) shows the *xz* cross-section view of a geometry; a concrete cylinder with a central axis on the *z* axis with a radius of 100cm. Two results of the neutron fluence obtained by the transport calculation without and with `[ww bias]` were shown in Figs. (b) and (c), respectively. Source particles of 14MeV-neutrons were generated at $x = 0, y = 0, z =$ 90cm as an isotropic source. After generating a `[weight window]` section by `[t-wwg]`, the result shown in Fig. (b) was obtained by performing the calculation with only the `[weight window]` section. The result of the calculation with both the `[weight window]` section and the `[ww bias]` section of the example 51 was shown in Fig. (c). The neutron fluence in Fig. (b) was distributed in both regions of the small and large cell numbers. On the other hand, the fluence in Fig. (c) was distributed in only the large cell numbers, which were biased by `[ww bias]` of the example 51. As seen in this example, to focus on a certain region, the calculation can be efficiently performed by `[ww bias]`.

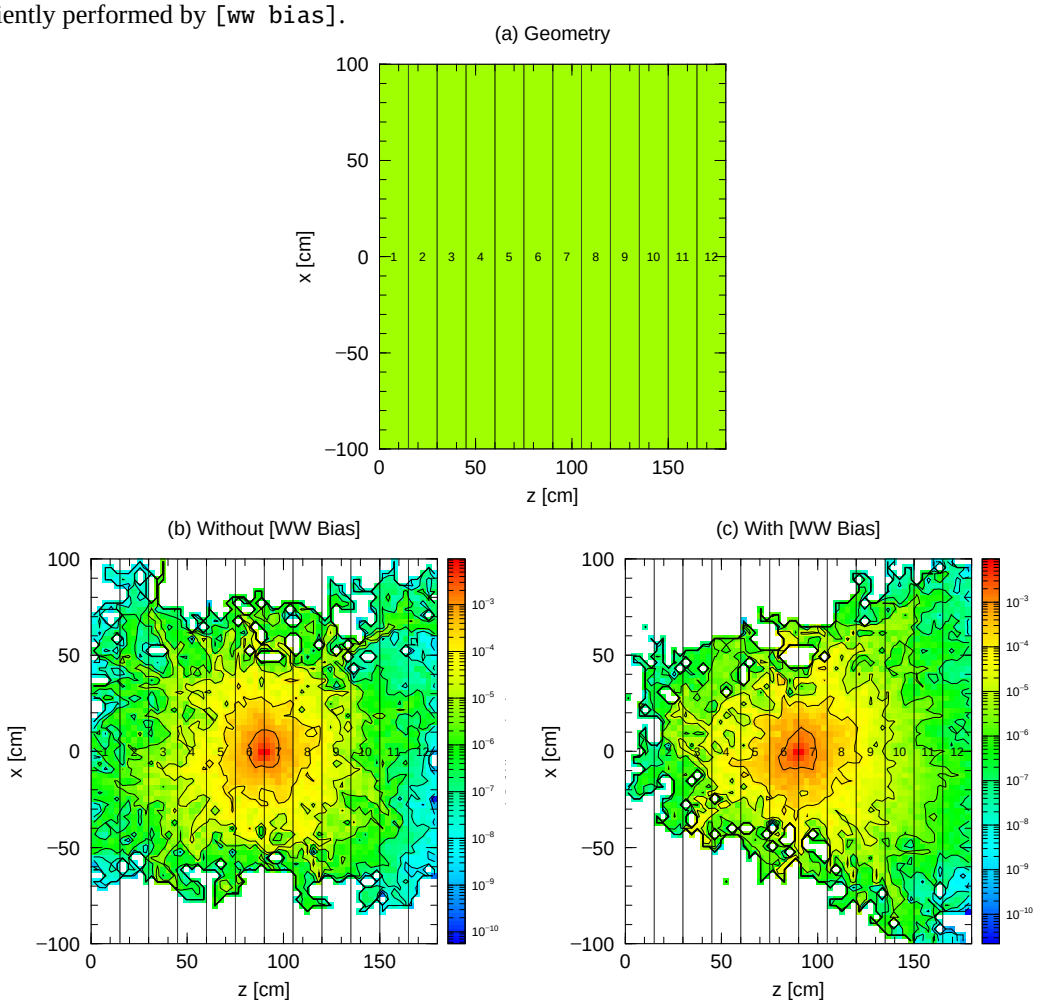


Figure 5.51: (a) *xz* cross-section view of the geometry. (b) Result without `[ww bias]`. (c) Result with `[ww bias]` of Example 51.

5.21 [Forced Collisions] section

[Forced collisions] can improve tally statistics of secondary particles generated from a certain target in which the collision probability is very low. It is especially useful in the analyses of secondary particle yields in thin target or in thick target irradiated with particles having low cross sections. When specified particle comes into a region defined as the forced collision region, the particle is divided into two particles. One has a weight by (penetration probability) × (its weight), this particle pass through to the next region. The other has a weight of (1 - penetration probability) × (its weight), and it is forced to collide with a target in the region. For neutral particles, the transmission probability is calculated according to the cross section of the incident particle, and the collision position is determined stochastically so that it is uniform within the specified region. In the case of charged particles, on the other hand, the forced collision region is divided into a number of segments, the energy loss and collision probability in each segment is determined in advance, and the collision positions are sampled according to these probabilities. The number of the segments is the same for all forced collision regions and is defined by nfcseg (D=50) in the [parameters] section. Note that the accuracy of the simulation becomes worse when [forced collisions] is used in a region in which the collision probability without [forced collisions] is greater than approximately 10%.

Regions and factors for the forced collisions can be defined in this section. Non-defined regions are set factor zero.

Maximum 6 [forced collisions] sections are allowed to be defined in an input file.

```
    [ Forced Collisions ]
        part =  proton  neutron
        reg               fcl
          1              1.000000
         11              0.500000
( { 2 - 5 } 8 9 )        0.200000
   ( 11 12 15 )          0.300000
( 6<10[1 0 0]<u=3 )     -0.500000
        ....              ........
        ....              ........
```

You set particle as part= in the first line. The default is part=all. part= is the same format as in tally definition.

If you want to replace the order of region number (reg) and (fcl), you can set as fcl reg. You can use the skip operator non.

You can use the format ({ 2 - 5 } 8 9), and you can use the lattice and universe style as (6 < 10[1 0 0] < u=3). But you need to close a value by ( ) if it is not a single numeric value. By using this format you can set different forced collision factor for each lattice. If the same cell is re-defined, the value, which is defined at first, is used.

The forced collision factor fcl means, 0: no forced collision, $|fcl| > 1$: is an error, and $|fcl| \leq 1$: multiply forced collision probability by fcl, instead the weight is reduced by $1/fcl$ times.

We have two options to control the particle transport and multiple scattering with the weight cut off in the forced collisions region. When $fcl < 0$, secondary particles produced by forced collisions are treated by the normal process. In this case, weight cut off is not performed. When $fcl > 0$, the forced collision is also applied to secondary particles. In this case, weight cut off is performed. Even if a particle is killed by this weight cut off, of course the particle is tallied before killed. There is a possibility that all particles are killed by this weight cut off, if you set the weight cut off and the forced collisions without consideration. For example, it corresponds to tallying tracks of secondary particles and information of particles at a distance from the forced collisions region. When you want to transport secondary particles produced by forced collisions, you should decrease the value of the cutoff weight parameter wc1(i) in the [parameters] section.

5.22 [Repeated Collisions] section

In this section, you can set some parameters used in this function to repeat the collisions. The collisions treated in this function are restricted to non-elastic collisions described by the reaction models excluding the reactions using the nuclear data. This function is enable to reduce the statistical fluctuation of the secondary products of the reactions, and very effective for important secondary particles produced through special reaction channels with small cross section. According to the repeated number, we change the weight of the produced particles. After the repeated collision, weight cut off is performed but secondary particles are tallied before killed by the weight cut off.

Maximum 6 [repeated collisions] sections are allowed to be defined in an input file.

```
[ Repeated Collisions ]
  part = proton neutron
mother = 2
          207Pb  208Pb
  emin = 800
  emax = 10000
reg    n-coll  n-evap
66       50       1
67       10       1
68       10      10
```

Projectile particle are set as part= in the first line. The default is part= all. Mother nuclei are also set as mother=, which is number of mother nuclei, and in the sequential lines the mother nuclei are defined. The default is all nuclei in the cell. emin and emax define the maximum and minimum energies of the projectile particles which are considered in this section. The default is all energy of the projectile. Three columns should be defined, reg is the cell number for the repeated collisions, n-coll is the number of repeated collision for the dynamical reaction processes, while n-evap for the evaporation processes, respectively.

5.23 [Volume] section

Volume for cell (cm^3) can be defined in this section. If the volume is double defined, the value defined in this section is used. The volume is utilized in the tally sections. If you do not set volume, it has 1.0 cm^3 volume.

```
    [ Volume ]
        reg              vol
          1            1.000000
         11            5.000000
( { 2 - 5 } 8 9 )      2.000000
   ( 11 12 15 )        3.000000
         16            6.000000
        ....           ........
        ....           ........
```

You can use the format `( { 2 - 5 } 8 9 )` for a group. In this case, you need to close a value by `( )` , if it is not a single numeric value.

You cannot use the lattice and universe style as `( 6 < 10[1 0 0] < u=3 )`. If you want to set cell volume in detail, use the volume definition in the tally section.

If you want to change the order of region number (`reg`) and volume (`vol`), you can set as `vol reg`. You can use the skip operator `non`.

5.24 [Multiplier] section

In the [multiplier] section, multiplier sets consisting of factors that depend on particle energies can be set as multipliers of [t-track], [t-cross], [t-point] tally results. To use this function, the multiplier subsections in the tally section must be defined. For example, this function can be used for dose estimation using any dose conversion factor.

Only one multiplier set can be defined per [multiplier] section; a maximum of 100 [multiplier] sections that can be defined in an input file. The format of this section is given as follows.

```
[ Multiplier ]
  number = -201
  interpolation = log
  part = all
  lagrange = 2
  file = multiplier201.out
  epsout  = 1
  y-txt = Conversion coefficients (arb.)
  ne = 10
       20.0       2.678
       30.0       7.020
       50.0       18.50
       100.0      24.26
       200.0      16.13
       500.0      10.51
       1000.0     10.55
       2000.0     10.98
       5000.0     12.10
       10000.0    12.45
```

The ID number of each [multiplier] section can be set by negative integer between -200 and -299. Particle type can be specified by part parameter. part = all indicates that the data are applicable to all particle. There are six particles that can be defined in a [multiplier] section. The data interpolation method is defined by interpolation parameter, which is set to log for log-log interpolation, lin for linear-linear interpolation, xlog or ylin for log-linear interpolation, xlin or ylog for linear-log interpolation. If you want to use the given data as group constants without interpolation, set interpolation parameter to glow or ghigh. The number of the energy point or group is given by ne, and the data of each energy and factor should be defined in the subsequent lines. Note that the data should be listed in ascending order of energy. The coefficient of the minimum energy point is used at energies below the minimum energy, and the coefficient of the maximum energy point is used at energies above the maximum. If you want to avoid using values outside the defined energy range, adjust the e-type for each tally.

Starting with version 3.35, the interpolation method using the 4-point (cubic) Lagrangian interpolation formula is available. This interpolation method is also recommended by the ICRP. The lagrange parameter is used to switch between the two interpolation methods. If you specify the output file name in file, the input data and the interpolation data are output. epsout can be used to make the eps file. If you want to show the name of the coefficients defined in the [multiplier] section in the output file, use y-txt to specify the text to be displayed on the y-axis.

From ver. 3.03, the conversion coefficients for several types of radiation doses and soft error rate (SER) on semiconductor devices are pre-defined. Table 5.92 shows the multiplier ID of pre-defined data. Those data are included in phits/data/multiplier directory with the file name of "m+|ID|+.inp" (e.g., m200.inp for $k = -200$). The unit of conversion coefficients for radiation dose is pSv·cm^2, and that for soft error rate is (FIT/Mbit)/(/cm^2/s). You can add your own [multiplier] section in this folder. If you specify [multiplier] section with the same ID number in your input file, the pre-defined data are overwritten by the specified data.

The conversion coefficients for SER are derived from the neutron-induced SEU cross section for a virtual semi-

conductor device calculated by PHITS and device simulator [96]. The background SER[97] is about 400 (FIT/Mbit). Here, 1 (FIT) = 1e-9 (error/hour). Generally, SEU cross sections depend on the device. Therefore, the conversion coefficients are not used to estimate the exact SER. Please utilize them to estimate SER roughly or to compare SER in the radiation field and background SER.

Table 5.92: Pre-defined conversion coefficients.

Multiplier ID	Data information
-200	$H^*(10)$[98]
-201	Effective dose based on ICRP60 (AP irradiation)[99]
-202	Effective dose based on ICRP103 (AP irradiation)[100]
-203	Effective dose based on ICRP103 (ISO irradiation)[100]
-204	New operational quantity H^* (Maximum effective dose among the all irradiation conditions[100]
-210	Sex-averaged effective dose equivalent (ISO irradiation)[101]
-211	Effective dose equivalent for male (ISO irradiation)[101]
-212	Effective dose equivalent for female (ISO irradiation)[101]
-213	Dose equivalent for male red-bone marrow (ISO irradiation)[101]
-214	Dose equivalent for female red-bone marrow (ISO irradiation)[101]
-215	Dose equivalent for male skin (ISO irradiation)[101]
-216	Dose equivalent for female skin (ISO irradiation)[101]
-299	Soft error rate on semiconductor device (ISO irradiation)[102]

5.24.1 Multiplier subsection

The multiplier option of the tally section is used to define a multiplier set following the basic format (C k), where C is a normalization factor and k is the ID number of the set (note that k should be negative). The format of the multiplier subsection is given as follows.

```
multiplier = all
      part = neutron
      emax = 1000
   mat   mset1        mset2
   all   ( 1 -201 )  ( 2 -202 )
```

In the first line, `multiplier=all` should be specified. In the second line, `part=` defines which particle is to be considered. A maximum of six particles can be entered here; `all`, which is the default, can be also used. Only those particles listed will be multiplied. In the third line, `emax=` defines the maximum energy of multiplication. If `emax` is omitted, it is automatically defined as the maximum energy given in the `[multiplier]` section. `all` should be set in the `mat` column, which is the material number considered for multiplication. The columns `mset1` and `mset2` define the multiplier sets up to a maximum of six sets. The result of each set is printed out. Although several multiplier subsections can be defined in one tally section, the number of multiplier sets should be constant across subsections.

[96] The critical charge is assumed to be 0.6 fC. The neutron energies from 1 MeV to 1 GeV are considered. For details, please refer to "S. Abe and Y. Watanabe, IEEE Trans. Nucl. Sci. 61, 3519-3526 (2014)."

[97] The background means on the ground at Tokyo. The cosmic-ray neutron flux is calculated by PARMA model. The PARMA model can be downloaded from `http://phits.jaea.go.jp/expacs`

[98] Taken from EXPACS (http://phits.jaea.go.jp/expacs/)

[99] Y. Sakamoto, O. Sato, S. Tsuda, N. Yoshizawa, S. Iwai, S. Tanaka, and Y. Yamaguchi, "Dose conversion coefficients for high-energy photons, electrons, neutrons and protons", JAERI-1345, (2003) etc.

[100] ICRU Report 95, 2020.

[101] ICRP Publication 123, Ann. ICRP 42(4), 2013.

[102] S. Abe and Y. Watanabe, IEEE Trans. Nucl. Sci. 61, 3519-3526 (2014).

PHITS has some built-in coefficients available for the [multiplier] section. If you set $k = -1$, a value of 1/weight is used as the multiplication factor to obtain tally results of a Monte Carlo particle, i.e., particles always having their weights as 1. For $k = -2$, a value of 1/velocity is used. For $k = -120$, material density is used. Therefore, you can obtain mass in the region setting icntl=5. You can also set the ID number included in /phits/data/multiplier/ directory without specifying the [multiplier] section in the input file. Currently, the databases of the conversion coefficients for several types of radiation doses are included in the directory (see Table 5.92 in detail). The unit of the dose conversion coefficients is pSv.cm^2, and thus, the doses in the unit of pSv/sec can be directly calculated when the calculated fluence is normalized to the unit of /cm^2/sec. Note that the energy unit of the dose conversion coefficients for heavy ions is MeV/n (MeV per nucleons), and you have to set iMeVperu = 1 in the [parameters] section when you use the heavy ion data in your simulation.

In addition to them, the effective doses based on ICRP60 for the AP irradiation from proton, neutron, electron, and photon can be separately calculated by specifying $k = -101, -102, -112$, and -114, respectively, though this method is not recommended anymore because the corresponding doses can be calculated by $k = -201$. Note that the unit of these dose conversion coefficients is (μSv/h)/(n/sec/cm^2), i.e. the doses in the unit of μSv/h can be directly calculated when the calculated fluence is normalized to the unit of /cm^2/sec. It should be noted that the interpolation method of conversion factor has been changed in PHITS ver. 2.00 from linear-linear to log-log.

In addition, the cross sections and kerma factors contained in nuclear data libraries can be also used as mset in the following format:

```
( C  m  MT1  MT2  ... )
```

where C is the normalization factor, m is the material ID defined in [material], and MT1 and MT2 are the MT numbers of the cross sections to be multiplied with the fluence. In the case that more than one MT numbers are specified, the multiplied factor should be the product of them except when the MT numbers are connected with :, where : indicates "sum" instead of "product" in the same as [cell] section. The complete MT number list is available from https://wwwndc.jaea.go.jp/cgi-bin/ENDFfig?help=yes or Appendix B of the manual https://www.oecd-nea.org/dbdata/data/manual-endf/endf102.pdf The major MT numbers are shown below:

MT number		explanation
1	:	(z,total) Total reaction cross section [barn]
2	:	(z,z0) Elastic scattering cross section [barn]
3	:	(z,nonelas.) non-elastic scattering cross section [barn]
4	:	(z,n) production cross section of a neutron [barn]
16	:	(z,2n) production cross section of two neutrons [barn]
17	:	(z,3n) production cross section of three neutrons [barn]
18	:	(z,fission) fission cross section [barn]
22	:	(z,nα) production cross section of a neutron and an α particle [barn]
27	:	(z,abs) Absorbed cross section [barn]
28	:	(z,np) production cross section of a neutron and a proton [barn]
102	:	(z,γ) radiative capture cross section [barn]
103	:	(z,p) production cross section of a proton [barn]
104	:	(z,d) production cross section of a deuteron [barn]
105	:	(z,t) production cross section of a triton [barn]
106	:	(z,^{3}He) production cross section of a ^{3}He particle [barn]
107	:	(z,α) production cross section of an α particle [barn]
108	:	(z,2α) production cross section of two α particles [barn]
109	:	(z,3α) production cross section of three α particles [barn]

Note that when you multiply the result with part=neutron by the cross section of a reaction that produces charged particles, such as MT=103, set e-mode> 0 in [parameters].

Negative MT number has special meanings as follows:
For part=neutron,

special number		explanation
−1	:	Total cross section without thermal adjustment [barn]
−2	:	Absorption cross section [barn]
−3	:	Elastic cross section without thermal adjustment [barn]
−4	:	Average heating number [MeV/collision]
−5	:	γ-ray production cross section [barn]
−6	:	Total fission cross section [barn]
−7	:	Fission ν [number/fission]
−8	:	Fission Q [MeV/fission]

For `part=photon`,

special number		explanation
−1	:	Compton (incoherent) scattering cross section [barn]
−2	:	Rayleigh (coherent) scattering cross section [barn]
−3	:	Photoelectric cross section [barn]
−4	:	Pair production cross section [barn]
−5	:	Total cross section [barn]
−6	:	Photon heating number [MeV/collision]

Followings are examples of `mset` using MT numbers.

- `(100.0 5 1)`: 100 x total cross section of material ID 5
- `(1.0 3 1 -4)`: Neutron kerma factor (total cross section x average heating number) of material ID 3 (`part` should be neutron)
- `(1.0 4 -5 -6)`: Photon kerma factor of material ID 4 (`part` should be photon)
- `(1.0 5 4:16:17)`: Sum of the (n,n), (n,2n), (n,3n) cross sections of material ID 5
- `(1.0 1 1 -4:-6 -8)`: Fission heat (Total fission cross section x fission Q).

If you want to know the actual multiplier value used, you have to define `mtinfo=1` just after `part` parameter in the multiplier subsection. Additionally, you have to add the `e-type` subsection after `mtinfo` for specifying the energy points for the output. The multiplier value will be output in the output file of the tally in the format of `[multiplier]` section. However, if the material number is set to 0, the coefficients will be written out when the first material number among those set in `[material]` is used in the calculation. For example, if the first written material number is not used in `[cell]` or in any other way, the coefficients for this material number are not written.

5.25 [Mat Name Color] section

Material names, size and colors for graphical output by [t-gshow] and [t-3dshow] tallies are defined in this section. By default, the name is set as material number and the color is set automatically.

```
[ Mat Name Color ]
            mat       name        size               color
              0       void           1           lightgray
              1        air         0.5          yellowgreen
              2     {mat 2}           2         orangeyellow
              3     {mat 3}           2  { 0.067 0.600 1.00 }
        { 4 - 7 }          Fe           3           mossgreen
       ....            ........
       ....            ........
```

To replace the order of material number (mat), (name), (size), and (color), set as mat color size name. The skip operator non can be used. At least one parameter in name and color must be defined. If no definition, the default values are used.

The format { 4 - 7 } can be used, but the format ({ 4 - 7 } 9 10) cannot be used. To use blanks in name definition, the name must be closed by { } as the example. To use ( ), the format \(\) should be used. In the name, { } cannot be used. Note that a superscript or subscript in the LaTeX format can be used by writing \{ \}. For example, in the case of writing ^\{ 208 \}Pb, ^{208}Pb is output. The maximum number of characters to define a name is 80.

When defining two (or more) cells of different densities from each other with the same material number in [cell] section, the cells except the first one are given other material numbers in matadd=1 (the default setting). The given numbers are written in the first part of file(6) (D=phits.out) as a warning message. Therefore, set the numbers have to be set in the mat column of this section. Even if setting matadd=0 in [parameters] section, the function unifying the material number is invalid in this section.

The color definition is based on the format in ANGEL. Set color by symbol (r bbb yy), name (red orange blue), or HSB numeric *H*(*hue*) *S* (*chroma*) *B*(*brightness*). In the case HSB numeric definition, close each numeric by { }. If only one HSB numeric is defined, chroma and brightness are set 1.

Color symbols, names, and HSB numerics are shown from next page. Note that names should be written by lowercase characters.

Table 5.93: Gray scale

Symbol	HSB	Output	Name
W	–1.0		white
O	–0.8		lightgray
K	–0.6		gray
J	–0.4		darkgray
F	–0.2		matblack
E	–0.0		black

Table 5.94: Color definition by symbols

Symbol	HSB	Output	Name
R	1.000		red
RR	0.933		orange
RRR	0.867		–
Y	0.800		yellow
YY	0.733		–
YYY	0.667		–
G	0.600		green
GG	0.533		–
GGG	0.467		–
C	0.400		cyan
CC	0.333		–
CCC	0.267		–
B	0.200		blue
BB	0.133		violet
BBB	0.067		magenta

Table 5.95: Color definition by names and HSB numerics

Name	Output	HSB		
darkred		1.000	1.000	0.600
red		1.000	1.000	1.000
pink		1.00	0.500	1.000
pastelpink		0.900	0.500	1.000
orange		0.933	1.000	1.000
brown		0.900	1.000	0.500
darkbrown		0.900	1.000	0.300
pastelbrown		0.900	0.600	0.500
orangeyellow		0.867	1.000	1.000
camel		0.800	0.700	0.700
pastelyellow		0.800	0.700	1.000
yellow		0.800	1.000	1.000
pastelgreen		0.700	0.600	1.000
yellowgreen		0.700	1.000	1.000
green		0.600	1.000	1.000
darkgreen		0.600	1.000	0.600
mossgreen		0.500	1.000	0.300
bluegreen		0.500	1.000	1.000
pastelcyan		0.400	0.400	1.000
pastelblue		0.250	0.400	1.000
cyan		0.400	1.000	1.000
cyanblue		0.400	1.000	0.500
blue		0.200	1.000	1.000
violet		0.133	1.000	1.000
purple		0.100	1.000	0.500
magenta		0.067	1.000	1.000
winered		0.002	0.800	0.700
pastelmagenta		0.067	0.600	1.000
pastelpurple		0.100	0.400	0.500
pastelviolet		0.133	0.400	1.000

5.26 [Reg Name] section

Region names and their font sizes for graphic output by [t-gshow], [t-rshow], and [t-3dshow] are defined in this section. When you set the gshow or rshow option in the other tallies, this definition is applied. By default, a region name is set to its region number.

```
[ Reg Name ]
            reg       name        size
              1       cover          1
              2        body        0.5
              3     {cell 2}          2
              4     {cell 3}          2
      { 5 - 8 }        tube          3
    ....            ........
    ....            ........
```

If you want to replace the order of region number (reg), region name (name), and font size (size), set as reg size name. You can use the skip operator non. At least one of name and size must be defined. If nothing is defined, it is assumed to be default. You can use the format { 4 - 7 }, but the ({ 4 - 7 } 9 10) format cannot be used. If you need to use blanks in the name definition, the name must be closed by { } as the example. If you want to use (and), you should write \(and \), respectively. Brackets { and } cannot be used in the name definition. Note that a superscript or subscript in the LaTeX format can be used by writing \{ \}. For example, in the case of writing ^\{ 208 \}Pb, ^{208}Pb is output. The maximum number of characters of a name that you can define is 80. You can specify a font size as a relative value to the default size.

5.27 [Counter] section

The counter function can be defined in this section. Three counters can be used in tally sections. The counter basically counts when

(1) a particle comes into specified region (in)[103]

(2) a particle goes out specified region (out)

(3) a particle takes scattering (or nuclear reaction) in specified region (coll)

(4) a particle reflects back on a certain boundary of the region (ref)

(5) a yield value is tallied in `[t-yield]` or `[t-dchain]` with `ndata=2,3` (ndata))

You can set one progress value of the counter from -9999 to 9999, or zero set (10000). Counter values are attached to particles. Secondary particles produced in collisions inherit the counter value of their parent. Capacity of the counter is from -9999 to 9999. Counter changes only this range. You can set the counter for each particle by using `part=` definition, and you can exclude some particles from the counter actions by `*part=` definition.

```
[ Counter ]
   counter = 1
      part = neutron proton
    reg     in     out    coll    ref
      1      1   10000       0      0
     11      1   10000       0      0
   counter = 2
     *part = proton deuteron triton 3he alpha nucleus
    reg                          in     out    coll
     ( { 2 - 5 } 8 9 )           -1       0       1
   counter = 3
      part = 208Pb
    reg                         coll
     ( 11 12 15 )                  5
     ( 6<10[1 0 0]<u=3 )          100
    ....            ........
    ....            ........
```

If you want to change the order of region number (`reg`), (`in`), (`out`), (`coll`), and (`ref`), set as `reg coll in out ref`. You can use the skip operator `non`. At least one must be defined in the `in out coll ref`. If nothing is defined, it is assumed no counter. Numeric gives one progress value of the counter. 10000 means zero set. The initial counter value of source particle is zero.

You can use the format `( { 2 - 5 } 8 9 )`, and you can use the lattice and universe style as `( 6 < 10[1 0 0] < u=3 )`. But you need to close a value by `(  )` if it is not a single numeric value. When you make the `[Cell]` section using a universe, specify "the cell number defined in the universe" instead of "the cell number of the frame where the universe is inserted."

In the definition of `part=`, you can specify particles up to 20 particles. For nucleus, you can use the expression like `208Pb` and `Pb`. The latter case, `Pb`, denotes all isotopes of `Pb`.

[103] The counter also counts when a source particle is generated within the region specified by in. This specification was not written in the manual, but has been included since the counter function was implemented. If you have performed a calculation that generates sources within the region specified by in, please confirm it.

From ver. 2.90, by detailed classification for coll, the opportunities shown in Table 5.96 are available as keywords so that the counter counts. nucl, atom, and dcay, which are particular interactions, belong to coll. Furthermore, nucl and atom are classified into three and ten kinds, respectively. When you want to analyze the PHITS simulation in detail, set these keywords. Note that if you set coll and nucl at the same time, the counting is duplicated when a nuclear reaction event occurs.

When (fiss) is specified, the counter is called when fission channels of the nuclear data library and the statistical decay model (GEM) are chosen.

Table 5.96: Categories of reaction events belonging coll.

Interaction	Event	Explanation
nucl		A particle takes nuclear interactions in specified region.
	elst	A particle takes elastic scattering in specified region.
	iels	A particle takes inelastic scattering in specified region.
	fiss	A particle takes fission in specified region.
atom		A particle takes atomic interactions and generate secondary particle in specified region.
	delt	A particle generates delta-rays in specified region.
	knoe	An electron or positron generates knock-on electrons in specified region.
	fluo	A particle generates atomic fluorescence x-rays in specified region.
	auge	A particle generates Auger electrons in specified region.
	brem	A particle takes bremsstrahlung in specified region.
	phel	A particle takes photoelectric effect in specified region.
	cmpt	A particle takes Compton scattering in specified region.
	pprd	A particle generates electron-positron pair in specified region.
	anih	A particle is annihilated in specified region.
	rayl	A particle takes Rayleigh scattering in specified region.
dcay		A particle decays in specified region.

When in or out is specified, the counter value will change after the particle information has been tallied in [t-cross]. Therefore, if the counter value before the change is not specified, the information at the moment of crossing the boundary plane will not be tallied. On the other hand, when coll or reaction events belonging to coll are specified, the counter value will change before the particle information has been tallied in [t-yield], [t-product], [t-interact], [t-point]. Therefore, to tally the information at the moment the reaction event occurs, specify the counter value after the change.

Specifying ndata in [counter] changes the counter value when a yield value in [t-yield] or [t-dchain] using ndata=2,3 option is replaced with cross section data in a directory specified by file(27) (D=file(1)/XS/yield/). This setting can avoid double-counting of the tally yield. When you set ndata option in [t-yield] or [t-dchain], a value based on the cross section data is tallied instead of a result of nuclear reaction models. However, for example, if a secondary particle produced by a nuclear reaction model has an energy greater than emin, it will be counted twice, with the contribution of the replacement at the time of the reaction event and the contribution at the time of the energy cutoff after its transport. In order to avoid this double-counting, setting a value of ndata to +1 in [counter] and specifying ctmax(i)=0, (i=1,2,3) in the tally section are useful, because the contribution of the secondary particles is ignored.

5.28 [Timer] section

The timer function can be defined in this section. The timer controls the time of each particle when

(1) a particle comes into specified region (in)[104]

(2) a particle goes out specified region (out)

(3) a particle takes scattering in specified region (coll)

(4) a particle reflects back on a certain boundary of the region (ref).

You can set the time to be zero(-1), stopped(1) or nothing(0).

```
[ Timer ]
    reg     in     out    coll     ref
      1      0      -1       0       0
     11      1       0       0       0
    ....   ....    ....    ....    ....
    ....   ....    ....    ....    ....
    ....   ....    ....    ....    ....
```

If you want to replace the order of region number (`reg`), (`in`), (`out`), (`coll`), and (`ref`), set as `reg coll in out ref`. You can use the skip operator `non`. At least one must be defined in the `in out coll ref`. If nothing is defined, it is assumed no action.

You can use the format `( { 2 - 5 } 8 9 )`, and you can use the lattice and universe style as `( 6 < 10[1 0 0] < u=3 )`. But you need to close a value by `(  )` if it is not a single numeric value.

[104] The counter also counts when a source particle is generated within the region specified by in. This specification was not written in the manual, but has been included since the counter function was implemented. If you have performed a calculation that generates sources within the region specified by in, please confirm it.

5.29 [User Defined Interaction], [User Defined Particle] section

This section is for utilizing user-defined interactions or particles.

- See ‘‘/phits/utility/UserDefinedModel/readme-en.pdf" for details.
- Currently, when using User Defined Interaction for electrons, positrons, and photons, it is necessary to use EGS5 (negs=1,2).
- The range of user-defined particle kf-code values that can be set is $900000 \leq$ kf ≤ 999999 or $-99999999 \leq$ kf ≤ -900000.

There are three types of files in the /phits/src/ folder:

(1) udm_int_<NAME>.f90
A source file defines a module with information on user-defined interactions, such as cross-sections. The <NAME> part of the filename can be specified by the user. The udm_int_sample.f90 file is already included in the /phits/src/ folder as a sample. In this sample source, electrons or positrons with energies over 100 MeV interact with material of atomic number Z at a cross-section of $Z \times 10^{-6}$ barn, emitting one user-defined particle.

(2) udm_part_<NAME>.f90
A source file defines a module with information on user-defined particles, such as mass, lifetime, and branching ratios. The <NAME> part of the filename can be specified by the user. The udm_part_sample.f90 file is already included in the /phits/src/ folder as a sample. This sample source defines a user-defined particle that decays into an electron neutrino. Its mass and lifetime can be specified in the input file.

(3) udm_manager.f90
A source file that specifies the module to be used.

An example input for [User Defined Interaction]:

```
[ User Defined Interaction ]
 $ Name           Bias    Parameters
 my_interaction   1.0e+5  900000
```

This input example is an input file to run udm_int_sample.f90. The first and second arguments input the Name of the module to be used and the bias value, respectively. For instance, my_interaction corresponds to the string assigned to the Name variable in the udm_int_sample module defined in udm_int_sample.f90. The occurrence frequency of processes with small cross-sections can be increased by setting the bias value greater than one. In such cases, the weight of the generated particle is automatically adjusted, so the average final results are not dependent on the bias value. For the third argument, zero or more parameter values can be input. The number and meaning of the parameters to be used depend on the source file. In the case of the udm_int_sample module, the first parameter value represents the kf-code value of the user-defined particle being generated.

An example input for [User Defined Particle]:

```
[ User Defined Particle ]
 $ Name       kfcode   Parameters
 my_particle  900000   1.0 1e-8
```

This input example is an input file to run udm_part_sample.f90. The first and second arguments input the Name of the module to be used and the kf-code value of the particle being handled in this file, respectively. For instance, my_particle corresponds to the string assigned to the Name variable in the udm_part_sample module defined in udm_part_sample.f90. For the third argument, you can input zero or more parameter values. The number and significance of the parameters to be used depend on the source file. In the case of the udm_part_sample module, the first and second parameter values represent the user-defined particle's mass (in MeV) and lifetime (in seconds). The udm_part_<NAME>.f90 file can be used not only to define user-specific particles but also to modify the properties (such as lifetime and branching ratios) of existing particles.

5.29.1 Applications Examples of User-Defined Code

Example 1: Muon Pair Production Induced by Positrons ($e^+e^-_{\text{atomic}} \rightarrow \mu^+\mu^-$)

User Code: udm_int_ee2mumu.f90

Usage:

```
[ User Defined Interaction ]
$ Name     Bias      Parameters
ee2mumu    1.0e+5
```

Example 2: Neutrino reaction
By loading neutrino databases, a wider range of neutrino reactions can be generated. An example of the database can be downloaded from the following URL:
https://rcwww.kek.jp/research/shield/sakaki/phits/temp.html

User Code: udm_int_neutrino.f90

Usage:

```
[ User Defined Interaction ]
$ Name     Bias   Parameters
neutrino   1      <Absolute or relative path to the database folder>
```

Example 3: Apply Energy Cutoff Only to Specified Regions and Particles.

User Code: udm_int_kill.f90

Usage: If a particle corresponding to kf-code <kf> in region number <reg> is not transported when its energy is less than <kf>, write these three values repeatedly in the Parameters section. The value of <Bias> is ignored.

```
[ User Defined Interaction ]
$ Name    Bias   Parameters
kill      1      <kf1> <reg1> <emin1>   <kf2> <reg2> <emin3>  ...
```

Input-1: Do not transport electrons (kf=11) in region 101 if their energy is below 500 MeV, and do not transport neutrons (kf=2112) in region 102 if their energy is below 100 MeV. (If kf=0, the condition applies to all particles.)

```
[ User Defined Interaction ]
$ Name   Bias   Parameters
kill     1      11 101 500   2112 102 100
```

Input-2: Prepare a data file in which each line contains the information in the format <kf> <reg> <emin>. Specify the relative or absolute path of that file in the Parameters section.

```
[ User Defined Interaction ]
$ Name   Bias   Parameters
kill     1      emin.txt
```

Example of emin.txt (lines that do not contain three numbers are skipped):

```
#   kf  reg  emin
    11  101   100
   -11  101   100
  2212  102    50
     0  103    30
```

6 General format of tally sections

PHITS has the following tally functions.

Table 6.1: Tally sections

name	explanation
[t-track]	Particle fluence in a certain region.
[t-cross]	Particle fluence crossing at a certain surface.
[t-point]	Particle fluence at a certain point.
[t-deposit]	Deposit energy in a certain region.
[t-deposit2]	Deposit energies in certain two regions.
[t-heat]	Heat generation in a certain region. (Not recommended[105])
[t-yield]	Residual nuclei yield in a certain region.
[t-product]	Produced particle in a certain region.
[t-dpa]	Displacement Per Atom (DPA) in a certain region.
[t-let]	LET distribution in a certain region.
[t-sed]	Microdosimetric quantity distribution in a certain region.
[t-time]	Time information of particle in a certain region.
[t-interact] (formerly named [t-star])	Number of interactions occurred in specified regions.
[t-dchain]	Residual nuclide yields (in combination with DCHAIN).
[t-wwg]	Output parameters for [weight window].
[t-wwbg]	Output parameters for [ww bias].
[t-volume]	Automatic calculation of region volume.
[t-userdefined]	Any quantities that user defined.
[t-gshow]	2D geometry visualization.
[t-rshow]	2D geometry visualization with physical quantities.
[t-3dshow]	3D geometry visualization.

Common parameters used in these tallies are described below.

6.1 Mesh type

In the tallies shown by Table 6.1, region mesh (reg), r-z scoring mesh (r-z), and xyz scoring mesh (xyz) can be used for mesh type of tallying area. Additionally tetrahedral mesh (tet) can be used for [t-track], [t-deposit], [t-yield], [t-product], [t-dpa], and [t-dchain].

One mesh should be specified like:

```
mesh = reg
```

6.1.1 Region mesh

The region mesh defined by the cell number can be written by:

[105] Before ver. 3.04, the [t-heat] tally was used to calculate deposit energy using the kerma approximation, because the [t-deposit] tally did not have the option.

```
mesh = reg
 reg = 1 2 { 3 - 5 } all
```

Each cell number should be separated by blank. In the format {n1 - n2} (n1 is smaller than n2), you can specify regions from n1 to n2. { 3 - 5 } in the above example means 3 4 5. When you set all, all defined regions are separately specified.

Some regions can be combined by using ( ).

```
mesh = reg
 reg = ( 1 2 )  ( { 3 - 5 } )  ( all )
```

Specification of (1 2) gives a sum of tally results in the cell numbers 1 and 2. In the case of ({ 3 - 5 }), a sum of tally results in the cell numbers 3, 4, and 5 can be obtained. Note that you cannot specify like (n1 - n2). When you set (all), you can obtain a sum of tally results in all defined regions. A new cell number of the combined region is given automatically starting from 1000001. The cell number is output in the echo part of the tally file, and is used when a volume of the region should be defined in a volume subsection, which will be mentioned in the next section.

When a geometry is defined using universe or lattice function, the geometry has repeated structures including several universes. A level structure is indicated by <, and the definition must be closed by ( ).

```
mesh = reg
 reg = ( 301 < 101[0 0 0] )
```

This example is referred to the geometry of the example 37 in [cell] section. The description of the above example means the cell number 301 in the cell 101 at the lattice coordinate $(0,0,0)$. (See Sec. 5.6.4 for the lattice coordinate.) You can specify a lattice coordinate by [ ] after the cell number corresponding to the lattice structure. Therefore, you can obtain a result of the cell number 301 included in a cell other than that at the lattice coordinate $(0,0,0)$. When you specify some lattice coordinates separately, you can use the following format with commas: 101[-1 -1 0, 1 -1 0, 0 0 0]. Furthermore, when you use the format like 101[-1:1 -1:1 0], you can tally nine results at lattice coordinates (i,j,k) in the region of $-1 \le i \le 1, -1 \le j \le 1, k = 0$. Note that this format gives the nine results individually. If you want to a sum of the results, please use ( ) like (301 < (101[-1:1 -1:1 0])). The style ( ) in one level can be used to combine some regions. When you specify (u=3 < 101[0 0 0]), PHITS internally expands all cells included in u=3. For example, in the case that two cell numbers 301 and 302 are included in u=3, the above expression means reg= (301 < 101[0 0 0]) (302 < 101[0 0 0]). You can use reg=all to specify all defined regions even in the case of the lattice structure, but cells other than in the bottom level are not considered.

6.1.2 Definition of the region and volume for repeated structures and lattices

See next example.

Example 52: mesh=reg example (1)

```
1:        mesh = reg
2:        reg = (all)
3:              ({ 201 - 205 })
4:              ( 161 < 160[1:2 3:6 1:1] )
```

```
5:              ( (201 202 203 204) < (161 162 163 ) )
6:              ( ( 90 100 ) 120 < 61 ( 62 63 ) )
```

This region mesh definition is echoed as

Example 53: `mesh=reg` example (2)

```
 1:        mesh =  reg            # mesh type is region-wise
 2:         reg = ( all ) ( { 201 - 205 } ) ( 161 < 160[ 1:2 3:6 1:1 ] ) ( (
 3:               { 201 - 204 } ) < ( { 161 - 163 } ) ) ( ( 90 100 ) 120 < 61
 4:               ( 62 63 ) )
 5:        volume                 # combined, lattice or level structure
 6:        non   reg      vol     # reg definition
 7:         1  10001   8.1000E+01 # ( all )
 8:         2  10002   5.0000E+00 # ( { 201 - 205 } )
 9:         3  10003   1.0000E+00 # ( 161 < 160[ 1 3 1 ] )
10:         4  10004   1.0000E+00 # ( 161 < 160[ 2 3 1 ] )
11:         5  10005   1.0000E+00 # ( 161 < 160[ 1 4 1 ] )
12:         6  10006   1.0000E+00 # ( 161 < 160[ 2 4 1 ] )
13:         7  10007   1.0000E+00 # ( 161 < 160[ 1 5 1 ] )
14:         8  10008   1.0000E+00 # ( 161 < 160[ 2 5 1 ] )
15:         9  10009   1.0000E+00 # ( 161 < 160[ 1 6 1 ] )
16:        10  10010   1.0000E+00 # ( 161 < 160[ 2 6 1 ] )
17:        11  10011   4.0000E+00 # ( ( { 201 - 204 } ) < ( { 161 - 163 } ) )
18:        12  10012   2.0000E+00 # ( ( 90 100 ) < 61 )
19:        13  10013   1.0000E+00 # ( 120 < 61 )
20:        14  10014   2.0000E+00 # ( ( 90 100 ) < ( 62 63 ) )
21:        15  10015   1.0000E+00 # ( 120 < ( 62 63 ) )
```

In the input, it looks only 5 regions defined, but in the input echo, you can see 15 regions are defined for tally. In this input echo, region numbers are defined automatically starting from 10001, and the volume of each cell is set 1 because of no `[volume]` definition.

We explain the detail of 15 regions appears in the volume description of this input echo.

First for `( all )`, 81 cells are defined in the bottom level, so the volume of `( all )` is set 81. If the volume of the cell is defined correctly in the `[volume]` section, you don't need to define the volume here again.

Next for `({ 201 - 205 })`, this combined region has volume 5 in the echo, since this combined regions have 5 cells of bottom level. This is also not required to re-define here if the volume is set in the `[volume]` section.

For `( 161 < 160[1:2 3:6 1:1] )`, the region 161 is included as a lattice in region 160. In this expression in the lattice coordinate system, 8 lattices of the region 160 from 1 to 2 in *i* direction, 3 to 6 in *j* direction, and 1 in *k* direction, are used for the tally. In the echo, the number of regions in bottom level is echoed 1. In the case, you have to specify the volume by yourself by the volume definition below.

For `( (201 202 203 204) < (161 162 163 ) )`, some regions are defined in each level, but these are all closed by `( )`, so it means one region as a whole. In this case, given volume by the echo is not correct, so set volume manually by the volume definition below.

For `( ( 90 100 ) 120 < 61 ( 62 63 ) )`, there are two independent regions in each level, so 4 regions are defined here. In this case given volume by the echo is not correct too, so set volume manually in the `[volume]` section.

You can set volume as below.

```
mesh = reg
 reg = 1 2 3 4 ( 5 < 12 ) ( {13 - 17} )
volume
 reg       vol
   1       1.0000
   2       5.0000
   3       6.0000
   4       1.0000
10001      6.0000
10002      5.0000
```

In above example, region numbers from 1 to 4 are set normally as you can see, but regions `( 5 < 12 )` and `( {13 - 17} )` have numbers 10001 and 10002. These big values are given in an input echo automatically. You can see and paste this settings from the input echo.

If you want to change the order of region number (`reg`) and volume (`vol`), set as `vol reg`. You can use the skip operator `non`.

In the input echo, numbered entry is given in `non` column. When `axis=reg`, the numbered entry is used as a value of *x*-axis.

6.1.3 r-z mesh

When you use the r-z scoring mesh, first, offsets for x and y coordinate of the central axis of cylinder can be defined as

```
mesh = r-z
  x0 = 1.0
  y0 = 2.0
```

This can be omitted. Then, define r and z mesh as

```
mesh   = r-z
r-type = [1-5]
   ..........
   ..........
z-type = [1-5]
   ..........
   ..........
```

Mesh definition is described later.

You can add θ degree of freedom in r-z mesh of `[t-track]` tally. You can use `a-type` mesh definition to specify θ of r-z mesh in the same manner as in the other angle mesh cases. However, the maximum angle of θ mesh is restricted 360 degree (2π) in this case. You can obtain the θ dependence of results of `[t-track]` tally by setting `axis=rad` or `axis=deg`. The `rad` represents the θ in radian, while `deg` in degree.

6.1.4 xyz mesh

When you use the xyz scoring mesh, set x, y, and z mesh as

```
mesh   = xyz
x-type = [1-5]
   ..........
   ..........
y-type = [1-5]
   ..........
   ..........
z-type = [1-5]
   ..........
   ..........
```

Mesh definition is described later.

6.1.5 tet mesh

This mesh is applicable only when tetrahedron geometry5.6.5.4 is used in the geometry and only active in [t-track], [t-deposit], [t-yield], [t-product], [t-dpa] tallies. When you use the tet scoring mesh, a cell no. of the rectangular box where the tetrahedron geometry is implemented needs to be specified following to the mesh definition as

```
mesh   = xyz
 reg   = 100
```

The physical quantities specfied in the tally will be scored in all the tetrahedrons separatelly. Extraction of some tetrahedrons can be achieved by specifying the region in hierarchical universe region format as

```
reg = ( 201 202 < 101 )
```

where 201 and 202 are the cells corresponding to the universes of the tetrahedron geometry specified in cell 101 (Example 43). The physical quantities for tetrahedrons belong to those universe regions are separately tallied. Additionally only the universe region specification

```
reg = 201
```

is also possible. For more than one universe regions, bracket specification as `(201 202)` is necessary. These universe region specifications are valid when only one tetrahedron geometry is included in the PHITS input. Summation of tetrahedrons is possible using the region mesh (`mesh=reg`) with selecting the universe region. .... It should be noticed also the volume of tetrahedron is simply computed from the coordinates of the nodes and thus incorrect qunatities may be obtained when the tetrahedron geometry is clipped out through the use of nest structure of “universe”and “fill”. In such a case, the region mesh (`mesh=reg`) should be used instead.

6.2 Energy mesh

Energy mesh begins as

```
e-type = [1-5]
   ..........
   ..........
```

The unit is MeV (or MeV/n when `iMeVperU=1`). `e1-type` and `e2-type` are also used in DEPOSIT2 tally. Mesh definition is described later.

6.3 LET mesh

LET mesh begins as

```
l-type = [1-5]
   ..........
   ..........
```

The unit is keV/um. Mesh definition is described later.

6.4 Time mesh

Time mesh is defined as

```
t-type = [1-5]
   ..........
   ..........
```

The unit is nsec. Mesh definition is described later.

6.5 Angle mesh

Angle mesh in cross tally is defined as

```
a-type = [1, 2, -1, -2]
   ..........
   ..........
```

If `a-type` is defined by positive number, this mesh denotes cosine mesh. If `a-type` is defined by negative number, the mesh denotes angle mesh. Mesh definition is described later.

6.6 Mesh definition

There are 8 kinds of mesh definition as `e-type, t-type, x-type, y-type, z-type, r-type, a-type` and `l-type`. The format is common for every mesh types. So only the `e-type` definition is described below. For other types, replace `e` into `t`, `x`, `y`, ... and `a`. For example, replace `ne` as `nt, nx, ny, ... , na`, `emin` as `tmin, xmin, ymin, ... , amin`, and so on.

In the defined mesh, `emin` itself is included in the minimum bin, while `emax` itself is not included in the maximum bin.

6.6.1 Mesh type

You can use 5 kinds of mesh type as shown below.

Table 6.2: Mesh type

mesh type	explanation
1	Give number of groups and mesh points by data.
2	Give number of groups, minimum and maximum values. Mesh is divided equally by linear scale.
3	Give number of groups, minimum and maximum values. Mesh is divided equally by log scale.
4	Give mesh width, minimum and maximum values. Mesh points are given by linear scale. Number of groups is set automatically as resulting maximum value becomes same with given value, or takes larger value with small excess as possible.
5	Give minimum and maximum values and log value of mesh width. Mesh points are given by log scale. Number of groups is set automatically as resulting maximum value becomes same with given value, or takes larger value with small excess as possible.

It is noted that you can use only 1, 2 (-1, -2) mesh types in `a-type` definition.

Only when `[t-cross]` and `z-type=1`, `nz=0` can be set with only one data. This setting provides only one tally surface of $z = (data)$.

Each mesh type format is shown in followings.

6.6.2 `e-type=1`

When you use `e-type=1`, set number of group, then numerical data as

```
e-type = 1
    ne =  number of group
          data(1) data(2) data(3) data(4)
          data(5) data(6) data(7) data(8)
          .........
          .........
          data(ne+1)
```

You can use multi lines without any symbols for line connection.

6.6.3 e-type=2,3

When you use e-type=2,3, set number of group, minimum value, and maximum value as

```
e-type = 2, 3
    ne = number of group
  emin = minimum value
  emax = maximum value
```

6.6.4 e-type=4

When you use e-type=4, set mesh width, minimum value, and maximum value as

```
e-type = 4
  edel = width of mesh
  emin = minimum value
  emax = maximum value
```

6.6.5 e-type=5

When you use e-type=5, set mesh width, minimum value, and maximum value as

```
e-type = 5
  edel = log( width of mesh )
  emin = minimum value
  emax = maximum value
```

In the case, mesh width is for log scale, i.e., edel= $log(M_{i+1}/M_i)$.

6.7 Other tally parameters

6.7.1 Particle parameter

You can define particles as

```
part = proton neutron pion+ 3112 208Pb
```

or

```
part = proton
part = neutron
part = pion+
part = 3112
part = 208Pb
```

See Table 4.4 for particle identification. You can also use the kf code number.

If you define all particles as

```
part = all
```

If you want to tally some particles as a group, you can use ( ) as the following. The maximum number inside the ( ) is 6. One caution to use kf code number inside ( ) is that a space before) is required after the last kf code number.

```
part = ( proton neutron ) all pion+ 3112 208Pb
```

In this case, as the first group, the sum of proton and neutron contribution is tallied, the second is the sum of all. 5 groups of the particle are printed out in this tally.

For nucleus, you can use the expression like 208Pb and Pb. The later case, Pb, denotes all isotopes of Pb.

By setting a minus sign before the particle name, you can tally the results without contribution from the specified particle. ( ) is required to use kf code number and the minus sign is needed before ( ). You can also tally the results without contributions from a group of particles by using ( ). The minis sign should be placed before ( ) after grouping the particles. The use of minus sign to subtract a contribution is not allowed inside ( ).

```
part = -proton -(proton alpha) -(2112) -(11 -11)
```

The first specification gives a result without proton contribution, the second gives a result without proton and alpha contributions, the third gives a result without neutron (2112) contribution, and the forth gives a result without electron (11) and positron (-11) contributions.

6.7.2 axis parameter

X axis value for output is described here. There are many kinds of axis shown as (depend on type of tallies and mesh type),

```
eng, reg, x, y, z, r, t, xy, yz, zx, rz,
cos, the, mass, charge, chart, dchain
let, t-eng, eng-t, t-e1, e1-t, t-e2, e2-t, e12, e21
```

```
axis = eng
```

You can set multiple `axis` per one tally by

```
axis = eng  x  y
```

or

```
axis = eng
axis = x
axis = y
```

If you define multiple axes, output results are written in different files. So you need to specify multiple output files as shown in the next subsection when multiple axes are defined.

It should be noted that you can define only one `axis` in a `[t-yield]` section from ver. 2.50. This restriction was implemented to calculate statistical uncertainties correctly. If you want to define several axes in the `[t-yield]` tally, you have to set the corresponding number of `[t-yield]` sections in a input file.

6.7.3 samepage parameter

This parameter specifies the type of data to be displayed on the same page of the image output file. The parameter must be selected from those listed in the axis parameter. The default is `part`, in which case the results for multiple particles specified in the `part` parameter are output on the same page. The values of `samepage` and `axis` should NOT be the same. Note that when the number of data bins specified in `samepage` exceeds 20, the eps file only shows the results of bins 1 through 20. Not available for `[t-yield]` tally and some others.

6.7.4 file parameter

The format to define name of output file is,

```
file = file.001  file.002  file.003
```

Note that the file name should not have the extension of '.eps' or '.vtk.' As described before, when you set multiple `axis`, set output files for each axis like following example.

```
file = file.001
file = file.002
file = file.003
```

6.7.5 resfile parameter

Specify the name of the tally file output from the previous calculation to determine the random number to be used in the restart calculation. The format is shown below.

```
resfile = file.001
```

If you specify the file in a different directory than the input file, specify it as a relative or absolute path. Even if several `resfile` parameters are set in a tally section, only the earliest one is used. `resfile` is set to `file` by default. In this case, results of the past tally are overwritten.

6.7.6 unit parameter

Set output unit as

```
unit = number
```

The unit number and its meanings are described in each tally explanation.

6.7.7 factor parameter

You can use the format below to set the normalization factor.

```
factor = number
```

If factor is positive, all the results are multiplied by this factor. If factor is negative (e.g. $-c$), all the results are scaled so that the maximum becomes c. When you use the `[t-gshow]` tally, this factor defines the line thickness instead.

6.7.8 output parameter

Set output type as

```
output = name of output
```

Details are described in each tally explanation.

6.7.9 info parameter

This option defines whether detailed information is output or not. Set 0 or 1 as

```
info = 0, 1
```

6.7.10 title parameter

This option is for title as

```
title = title of the tally
```

It is omitted, and in this case, default is used.

6.7.11 ANGEL parameter

In order to add ANGEL parameters in tally output, define as

```
angel = xmin(1.0) ymin(1.3e-8)
```

Defined parameters are converted to the ANGEL format as

```
p: xmin(1.0) ymin(1.3e-8)
```

These parameters change the minimum of the horizontal and vertical axes, respectively. The main ANGEL parameters are shown in Table 6.3. Please see the ANGEL manual in more detail.

Table 6.3: ANGEL parameter

ANGEL parameter	Explanation
xmin	Minimum of horizontal axis.
xmax	Maximum of horizontal axis.
ymin	Minimum of vertical axis.
ymax	Maximum of vertical axis.
xlin	Change horizontal axis to liner scale.
xlog	Change horizontal axis to logarithmic scale.
ylin	Change vertical axis to liner scale.
ylog	Change vertical axis to logarithmic scale.
xdec	Decimal notation for horizontal axis values.
xexp	Exponential notation for horizontal axis values.
ydec	Decimal notation for vertical axis values.
yexp	Exponential notation for vertical axis values.
cmin	Minimum of color plot.
cmax	Maximum of color plot.
cmnm	Convert cm to nm.
cmum	Convert cm to μm.
cmmm	Convert cm to mm.
cmmt	Convert cm to m.
cmkm	Convert cm to km.
nsps	Convert nsec to psec.
nsus	Convert nsec to μsec.
nsms	Convert nsec to msec.
nssc	Convert nsec to sec.
scmn	Convert sec to min.
schr	Convert sec to hour.
scdy	Convert sec to day.
scyr	Convert sec to year.

Table 6.4: ANGEL parameter (2)

ANGEL parameter	Explanation
cmap	Specification of Colormap Name Example: angel = cmap(phits2) The available colormap names are 'phits2' and those used in Matplotlib[106]. By adding '_r' to the colormap name, the color bar can be inverted. Example: angel = cmap(hot_r)
ndis	Usage of discrete color bars and specification of the number of colors. Example: angel = ndis(15)

From ver. 2.89, the default size was changed from A4 to US letter. ANGEL parameters, a4us(US letter), a3pp(A3), a4pp(A4), a5pp(A5), b3pp(B3), b4pp(B4), and b5pp(B5) can be used to change an output size of an eps file. The characters given in parentheses represent each output size.

6.7.12 Sangel parameter

This parameter can be used to insert all ANGEL parameters into tally output files unlike the usual ANGEL parameter, which can specify only ANGEL parameters defined by p:. For example, defining infl: parameter for ANGEL, the tally file includes an experimental data file, and both calculated results and the experimental data can be plotted in a figure (eps file).

The SANGEL parameter can be set using the format sangel=*number of parameter lines*, and then writing the lines to define the ANGEL parameters. An example with sangel is as follows.

```
sangel = 2
infl:{exp.dat}
w: ($\theta=$ 0 deg) / X(10) Y(100)
```

where "exp.dat" is a data file of the experimental data. The data can be shown on the figure of the tally result. By using w: parameter, the comment "θ=0 deg" can be put at the coordinate of X=10, Y=100 in the figure.

6.7.13 2d-type parameter

When you define 2 dimensional output as axis=xy, you must set this 2d-type option as

```
2d-type =  1, 2, 3, 4, 5, 6, 7
```

These 2d-types give the format of data arrange.

- 2d-type = 1, 2, 3, 6, 7
  Data are written by below format (the example is written by Fortran style.

  ```
  ( ( data(ix,iy), ix = 1, nx ), iy = ny, 1, -1 )
  ```

  10 data are written in a line. Also a header for the ANGEL input is inserted. The ANGEL header is inserted by 2d-type=1 for contour plot, 2d-type=2 for cluster plot, 2d-type=3 for color plot, 2d-type=6 for cluster and contour plot, 2d-type=7 for color and contour plot.

[106] J. D. Hunter, "Matplotlib: A 2D Graphics Environment", Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007.
Colormap: https://matplotlib.org/stable/users/explain/colors/colormaps.html
License: https://matplotlib.org/stable/project/license.html
The license is also directly stated in a-main1.f.

- 2d-type=4
  Data are written by below format

```
do iy = ny, 1, -1
do ix = 1, nx
   ( x(ix), y(iy), data(ix,iy) )
end do
end do
```

  3 data of x(ix), y(iy) and data(ix,iy) are written in a line.

- 2d-type = 5
  Data are written by below format

```
y/x  ( x(ix), ix = 1, nx )
   do iy = ny, 1, -1
     ( y(iy), data(ix,iy), ix = 1, nx )
   end do
```

  nx+1 data are written in a line, and total ny+1 lines. It is useful to use in the tabular soft like Excel.

6.7.14 gshow parameter

This option can be used in all tallies without [t-gshow] and [t-rshow]. If you set gshow option with xyz mesh, xy, yz, or xz axis, and 2d-type=1,2, or 3, ANGEL can create a graphical plot with region boundary and material name, or region name, or lattice number on the two dimensional output. You can also obtain graphical plots directory from the PHITS calculation by the epsout option.

```
gshow =  0, 1, 2, 3, 4
```

In above example, 0 means no gshow option, 1 means gshow with region boundary, 2 means gshow with region boundary and material name, 3 means gshow with region boundary and region name, 4 means gshow with region boundary and lattice numbers. When you increase the resolution of the plot by resol parameter, the indication of region name, material name and lattice number on the graph are sometimes disturbed. In this case, you should increase the mesh points instead of resol.

You can see your geometry plot on a graph without transport calculation by setting icntl=8 in the [parameters] section, and this gshow option. You should check whether regions are correct, and a xyz mesh resolution is good or not, before long time calculation.

6.7.15 rshow parameter

You can use rshow parameter in all tallies except for [t-cross] and [t-gshow] tallies. This option is available with mesh=reg or tet, and axis=xy, yz, zx. This option makes a two dimensional plot in which each region or tetrahedral element is colored with the amount of the output value in the region or tetrahedral element, respectively. And region boundaries, material name, or region name numbers are also displayed. The xyz mesh definition is required after this rshow parameter.

```
rshow =  1, 2, 3
x-type = [2,4]
   ..........
   ..........
y-type = [2,4]
   ..........
   ..........
z-type = [2,4]
   ..........
   ..........
```

rshow=0 means no rshow option, 1 means rshow with region boundary, 2 means rshow with region boundary and material name, 3 means rshow with region boundary and region name numbers. If rshow=0, xyz mesh definition is not required, and should be commented out. When you increase the resolution of the plot by resol parameter, the indication of region name, material name and lattice number on the graph are sometimes disturbed. In this case, you should increase the mesh points instead of resol. Please note that without gslat option, the boundaries between the same material (rshow=2) or same cell (rshow=3) regions will not be shown and inaccurate value may be displayed for each region or tetrahedron. Therefore, use of gslat=1 in addition is recommended.

If you use the rshow option with reg mesh, there is no output for the values of each region. In this case, you cannot re-plot the figure because of no original data. When this rshow option is used, usually axis is set as xy, yz, and zx. But you should use in addition axis=reg or tet in order to save results into another file, for re-plotting. You can re-plot figures from saved data and [t-rshow] tally function.

You can execute this option without transport calculation by using icntl=10 in the [parameters] section. For icntl=10, *PHITS* makes a two dimensional plot for the tallies with reg mesh, xy, yz, zx axis and rshow = 1, 2, 3. In the figure, different colors are used for different materials. You should check whether regions are correct and a xyz mesh resolution is good or not, before long time calculation.

6.7.16 x-txt, y-txt, z-txt parameter

If you want to change x, y, and z axis titles in the output figure, use these option. These title cannot be defined in the ANGEL parameter.

```
x-txt = x axis title
y-txt = y axis title
z-txt = z axis title
```

6.7.17 volmat parameter

The volmat parameter corrects a volume where xyz mesh crosses region boundaries. This option is effective in the case that mesh is xyz, and the material parameter is defined. This corrected volume is calculated by the Monte Carlo method for specified material. volmat denotes the number of scanning parallel to x, y, and z axis respectively for the Monte Carlo calculation. So if you set too large volmat, the calculation takes long time. You need to take care of it. If volmat is given by negative value, all xyz mesh is scanned. If positive value, the scanning is not performed when 8 apexes of the mesh are included in the same material.

6.7.18 epsout parameter

If you set epsout=1, output file is treated by ANGEL automatically and an eps file is created. This eps file name is named by replacing the extension into ".eps." With itall=1 setting, the eps file is created after every

batch calculation. You can monitor the PHITS results in real time, by displaying the eps file with the ghostview and by setting refresh function for a file updating by typing 'w' key on the ghostview.

6.7.19 counter parameter

You can make a gate to the tallying quantities by using the counter defined by [counter] section. Set minimum ctmin(i) and maximum value ctmax(i) for each counter. The i is the counter number from 1 to 3. By default, ctmin(i)=-9999, and ctmax(i)=9999. When multiple counters are specified, the common part of these terms is tallied.

In addition, you can make a gate to the tallying quantities by using the history-counter, which is the maximum counter value of all particles generated in one history, by setting chmin(i) and chmax(i) parameters. Using this function, you can tally your interested events selectively by tracing back to the generation of their sources. Please see "/phits/sample/misc/history_counter" in more detail. Note that the history-counter does not work in the batch variance mode (istdev=1). Furthermore, this function does not work well when variance reduction method such as [weight window] is used or the event generator mode is not used.

6.7.20 resolution and line thickness parameters

You can increase the resolution of the region boundaries in the gshow, rshow, and 3dshow with keeping xyz mesh by resol. Default value is 1, it is same as xyz mesh resolution. If you set resol=2, the resolution becomes 2 times for each side. It is useful to draw smooth line for xyz mesh. Also you can obtain clear graphics by set resol larger for the 3dshow. Even if you set resol larger, memory usage is not changed.

The width shows the line thickness for gshow, rshow, and 3dshow. Default value is 0.5.

6.7.21 trcl coordinate transformation

By this trcl option, you can transform the coordinate of the r-z, and xyz mesh. There are two ways to define the transformation as below.

trcl = *number*
trcl = O_1 O_2 O_3 B_1 B_2 B_3 B_4 B_5 B_6 B_7 B_8 B_9 M

The first definition is to specify the transformation number defined in [transform] section. The next one is to define the transformation directly here with 13 parameters as same as in [transform] section. If the data are not written in a line, you can write them in multiple lines without the line sequential mark. But you need to put more than 11 blanks before data on the top of the sequential lines.

In the 3dshow tally, trcl can be used to transform the box. This will be explained in the [t-3dshow] tally section.

6.7.22 dump parameter

The [t-cross], [t-time], and [t-product] tallies can be used to output the particle information as dump data. The particle information to be output and the order of output are specified by the dump parameter. The file name into which the dump data is written is specified by appending "_dmp" to the file name specified by file parameter. Note that constraints based on the history counter (chmin & chmax) do not work with the dump function because each tally outputs the particle information to the dump file at a different time than the history counter constraint.

By the parameter of dump =, the number of the dump data in one record is specified. If this number is given by positive number, the data are read as binary data. If negative, the data are read as ascii data. In next line, the data sequence of one record is described. The relation between the physical quantities and id number is the followings,

Table 6.5: id number of dump data (1)

physical quantities	kf	x	y	z	u	v	w	e	wt	time	c1	c2	c3	sx	sy	sz
id number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Table 6.6: id number of dump data (2)

physical quantities	name	nocas	nobch	no
id number	17	18	19	20

Here kf means the kf-code of the particles (see Table 4.4), x, y, z are coordinates [cm], u, v, w denote the unit vectors of the direction of the particle, e is the energy [MeV, or MeV/u for nucleus], wt is the weight, time is the initial time [ns], c1, c2, c3 are the values of counters, and sx, sy, sz are the unit vectors of the direction of spin, respectively. name is a collision number of the particle, nocas is a current history number of this batch, nobch is a current batch number, no is a cascade id in this history. These are assumed as real*8 for the binary data, *n*(1p1e24.15) data format for the ascii data.

For an example, one record has 9 data as

```
kf  e  wt  x  y  z  u  v  w
```

To read this data, we write the parameters as

```
dump = 9
  1  8  9  2  3  4  5  6  7
```

When you use this dump parameter, `axis` and `file` are restricted to one axis and one file, and `unit` is always 1. The dumped data are written on a file named "***_dmp," where "***" indicates the file name specified by `file`="***." The normal output of the tally is written on "***." From this file, you can get the information on the total normalization factor. In the parallel computing, files to the number of (PE−1) corresponding to each PE (Processor Element) are created for writing and reading dumped data. If you set `idpara=0` or `1`, a file is made in the directory named by `/wk/uname/` on each of the nodes. If you set `idpara=1` or `3`, the each IP number is put at the end of the filename. The each PE writes down its result on only the corresponding file, and reads it from the same file in the re-calculation.

6.8 Function to sum up two (or more) tally results

After version 2.74, PHITS has a function to sum up two (or more) tally results. There are two methods for sum up; one is to integrate the tally results considering the history number of each simulation, while the other is to add the tally results weighted by user defined ratios. The former can be used for the parallel calculation on computers in which MPI protocol is not installed. On the other hand, the latter is suit for the simulation with different source terms whose intensities cannot be fixed before the simulation, such as that for Intensity-Modulated Radiation Therapy (IMRT).

In order to use this function, you have to satisfy the following conditions:

- `icntl` is set to 13 in [parameters] section
- The parameters for each tally results such as `mesh, axis` and `part` are identical to one another
- "sumtally" subsection is defined in the tally section that outputs one of the summing up tallies. [107] [108]

"sumtally" subsection is ignored when `icntl` parameter is not set to 13 in [parameters] section. [109] Please see the ppt slides or sample input file in `\phits\utility\sumtally\` for details.

"sumtally" subsection should be defined between the lines of `sumtally start` and `sumtally end` written in the tally section that outputs one of the summing up tallies. The `set:` definition is ignored in the sumtally subsection.

The parameters used in "sumtally" subsection are summarized in Table 6.7.

Table 6.7: Parameters used in "sumtally" subsection

name	value	description
isumtally =	1(default), 2	Summing up procedure. 1: Integration of tally results considering history number of each simulation. 2: Sum of tally results weighted by user defined ratios.
nfile = (next line)	Number filename value	Number of summing up files. Summing up file name, weighted value. Sum of the weighted values is automatically normalized to sumfactor for `isumtally=2`.
sfile =	filename	Output file name.
sumfactor =	(optional, D=1.0)	Normalization factor.

Users can use `isumtally=1` to manually obtain the parallel calculation results. For example, if you have one tally result "result-1.out" obtained with `maxbch=10` and `maxcas=100`, and another tally result "result-2.out" obtained with `maxbch=20` and `maxcas=100`, and if you would like to sum up the results to have same statistics as of `maxbch=30` and `maxcas=100`, you have to write:

Example 54: example for `isumtally=1`

```
1:  sumtally start
2:  isumtally = 1        $(D=1) sumtally option, 1:integration, 2:weighted sum
3:  nfile = 2            $ number of tally files
4:     result-1.out  1.0
5:     result-2.out  1.0
6:  sfile = result-s.out $ file name of output by sumtally option
7:  sumfactor = 1.0      $ (D=1.0) normalization factor
8:  sumtally end
```

Using this sumtally subsection, you can obtain the results for `maxcas=100` and `maxbch=30`. Please be sure that the initial random seeds for calculating "result-1.out" and "result-2.out" should be different from each other,

[107] Before ver. 2.81, this function is available even if `file` parameter is not defined in the tally section. However, after ver. 2.82, PHITS occurs the error in the setting without the definition of `file`.

[108] Before ver. 2.85, only one "sumtally" subsection can be defined in a PHITS input file.

[109] From ver. 2.88, "sumtally" can be applied to all kinds of tallies.

otherwise you would get biased results for certain random numbers. The most recommended method for changing the initial random seed is to use `irskip` parameter. The weighted value for each tally outputs are generally set to 1 for `isumtally=1`, unless you would like to change the weight of source particles. The output file obtained from this sumtally section, "result-s.out," can be used for restart calculation by setting `istdev<0`. Note that the initial random seed in the last file of the summing up files is used.

For `isumtally=2`, the weighted summation of the tally results, $\bar{X}$, is calculated by the following equation:

$$\bar{X} = F \sum_{j=1}^{N} \frac{r_j}{r} \bar{X}_j \tag{29}$$

where F is the normalization factor defined by `sumfactor`, N is the number of summing up files defined by `nfile`, $\bar{X}_j$ is the j-th tally results, r_j is the weighted value of j-th tally, and r is the sum of r_j, i.e. $r = \sum_{j=1}^{N} r_j$. The uncertainty of the summation value, σ_X, can be calculated by

$$\sigma_X = F \sqrt{\sum_{j=1}^{N} \left(\frac{r_j}{r}\right)^2 \sigma_{X_j}^2} \tag{30}$$

where σ_{X_j} is the standard deviation of j-th tally results.

If you have two tally results "result-l.out" and "result-r.out," and you would like to sum up them weighted by factors of 2.0 and 3.0, respectively, you have to write:

Example 55: example for `isumtally=2`

```
1:  sumtally start
2:  isumtally = 2        $(D=1) sumtally option, 1:integration, 2:weighted sum
3:  nfile = 2            $ number of tally files
4:     result-l.out  2.0
5:     result-r.out  3.0
6:  sfile = result-s.out $ file name of output by sumtally option
7:  sumfactor = 5.0      $ (D=1.0) normalization factor
8:  sumtally end
```

You can obtain the same results by using multi-source function, but it is more convenient to use sumtally subsection when you would like to change the weighted values for several cases. It should be mentioned that the sum of the weighted values is automatically normalized to `sumfactor` for `isumtally=2`. The output file obtained from this sumtally section, "result-s.out," cannot be used for restart calculation.

6.9 Function to analyze two (or more) tally results

Unlike the function in the previous section, you can perform more complex calculations using two (or more) tally results through an anatally function. For example, you can calculate the biological dose for charged particle therapy taking into account the absolute values of the absorbed doses, or you can estimate the effect of a material density error on the calculated results as systematic uncertainties.

This function analyzes multiple tally results obtained from normal transport calculations. It works in two ways.

- After performing the normal transport calculation manually, specify the resulting tally file to run the anatally function. (See example 56)
- Using "autorun" script automatically run through usual transport calculations and anatally function. (See Sec. 6.9.2)

The latter method is basically used to evaluate the systematic uncertainty. Although the former method can also be used for evaluation, the latter is more convenient for evaluating uncertainty by varying a large number of input values.

The Anatally function itself works by setting `icntl=17` in the `[parameters]` section. However, the following conditions must be met.

- Multiple tally results to be analyzed must be obtained with the same conditions regarding `mesh, axis, part` and the division number (bins) of energy, space, etc.
- A `[anatally]` section should be set with specifying the file names of the tally results to be analyzed.
- An empty "anatally" subsection, consisting of only `anatally start` and `anatally end`, is defined in the tally section that outputs the tally results.

`[anatally]` section is available only when `icntl=17`. In this case, two anatally subsections are needed in the `[anatally]` and usual tally sections. The former subsection should contain the parameters of the anatally function, such as the file names of the tally results, while an empty subsection consisting only of `anatally start` and `anatally end` should be written in the latter subsection. (Note that when you use the "autorun" script shown in Sec. 6.9.2, you should set `icntl=16`, define the anatally subsection with information about the tally file names in the usual tally section, and not set `[anatally]` section.)

"anatally" subsection should be defined between the lines of `anatally start` and `anatally end`. The `set:` definition is ignored in the anatally subsection.

The parameters used in "anatally" subsection are summarized in Table 6.8. Note that parameters shown in Sec. 6.10 such as `ix,iy,iz,ipart`, which are available when `itall=3`, cannot be used when `icntl=16,17`.

Table 6.8: anatally section parameters

name	value	description
`ianataldchain =`	(optional, D=0)	Option for `[t-dchain]` tally. 0: Do not use tally result files of `[t-dchain]`. 1: Use tally result files of `[t-dchain]`. Specifying filenames given as `part` parameters in the `[t-dchain]` section automatically handles the other output files for DCHAIN (.dout, .dtrk, .dyld). Note that if you use this option, it should be written in the next line just after `anatally start`.
`manatally =`	0, 1(default)	control parameter of anatally funcition 0: user-defined anatally function. Program written in "usranatal.f" is performed (See Sec. 6.9.1) 1: ANOVA (analysis of variance) mode. (See Sec. 6.9.2)
`nfile =` (next line)	Number file name	Number of tally files. file name for analysis
`sfile =`	file name	output file name

Example 56: example for manatally=0

```
 1:  [t-track]
        .........
        .........
 2:  anatally start
 3:  anatally end
        .........
        .........
 4:  [anatally]
 5:  anatally start     1
 6:  manatally = 0        $(D=1) anatally option, 0:user-defined anatally, 1:anova
 7:  nfile = 2            $ number of tally files
 8:     result-1.out
 9:     result-2.out
10:  sfile = result-a.out $ file name of output by anatally option
11:  anatally end     1
```

In this example, calculation of user-defined anatally function is used with result-1.out and result-2.out, which are output files of [t-track] obtained by usual transport calculations (icntl=0), and then its result is outputted in result-a.out. The numbers '1' written in the lines of anatally start and anatally end are anatally subsection ID. This ID corresponds to the order of the tally section in the PHITS input file. You have to set the order of the tally section including anatally subsection. For example, if one more tally section is defined before the [t-track] in the first line, the ID should be set 2 in the [anatally] section.

6.9.1 User-defined anatally function

The user-defined anatally function is designated for analyzing multiple tally results, using user-defined program written in "usranatal.f." The default program of usranatal.f can deduce the weighted sum of each tally result in similar to the sumtally function, or the photon isoeffective dose for BNCT based on the SMK model. [110] The detailed instruction for using this function is given in "/phits/utility/usranatal" folder, together with sample input files. The sample program and input file for calculating the biological dose for charged particle therapy based on the SMK model are also included in the folder.

6.9.2 Analysis function

Script files are useful for analyzing results of transport calculations obtained by varying their conditions. For example, in the case that material density has an error, the effect of the error on the calculated results can be estimated.

PHITS with the script files for the analysis function (autorun.bat & autorun.sh) can estimate automatically the effect of the variation on the tally result.
Note that in the current version, the following restrictions remain. Only [t-track] and [t-deposit](output=dose) is available. Only one value ci can be varied. Some special parameters such as ix,iy,iz,ipart, which can be specified when itall=3, cannot be available in anatally subsetion.

The followings are the procedure of how to use the script files.

(1) Make a PHITS input file (here, "phits.inp") for transport calculations.

(2) Make an input file for the scripts (here, "autorun.inp").

(3) Execute the script (autorun.bat or autorun.sh) in "/phits/bin/autorun" folder using autorun.inp.

The detail of the input files and the calculation process of the scripts will be shown in the next subsections.

[110] T. Sato and Y. Furusawa, Radiat. Res. 178, 341-356 (2012);
T. Inaniwa and N. Kanematsu, Phys. Med. Biol. 63, 095011 (2018);
T. Sato et al., Sci Rep. 8, 988 (2018)

6.9.2.1 PHITS input file for transport calculations (phits.inp)

In phits.inp (which can be renamed), specify a value to be varied by use-defined variables `ci`($i = 1, \cdots, 99$). Furthermore, set `icntl=16` in `[parameters]` section, and set anatally subsection in a tally section as follows:

Example 57: Anatally subsection when using analysis function.

```
1:  anatally start
2:  manatally = 1
3:  sfile = track_a.out
4:  anatally end
```

"anatally" subsection should be defined between the lines of `anatally start` and `anatally end` written in the tally section. `manatally` is option for analysis. When `manatally=1`, effects of the variation of `ci` on the tally result are estimated as systematic uncertainties based on analysis variance (ANOVA). (Note that only `manatally=1` is available in ver. 3.20) `sfile` is name for output of the result of the analysis.

6.9.2.2 input file for scripts of analysis function (autorun.inp)

The input file autorun.inp (which can be renamed) for the scripts should be made using the following format:

Example 58: Input file for scripts of analysis function.

```
1:  file=phits.inp
2:  set:c1
3:  c-type=2
4:  cmin=10
5:  cmax=100
6:  nc=10
```

Specify the input file name (phits.inp) after `file=`. Set user-defined variable `ci`($i = 1, \cdots, 99$) to be varied after `set:`. Form the next line of `set:c1`, specify the information on the variation of `ci` using `c-type` which has the same format as that of the tally mesh type like `e-type`. `c-type=1,2,3` are available. In above example, the scripts execute PHITS varying the value of `c1` from 10 to 100 in 10 increments.

6.9.2.3 Procedure of scripts of analysis function

The script (autorun.bat or autorun.sh) performs the following procedures.

(1) The script reads the information on `ci` from autorun.inp, and then outputs it to a file "nc_info.inp".

(2) The script executes PHITS using phits.inp with `icntl=16` to make "phits_tmp.inp", "varfile_j.inp" ($j = 1, \cdots, nc$), and "anatally.inp". phits_tmp.inp is a temporary PHITS input file for transport calculations. In varfile_j.inp, each value of `ci` is written by `set:` command. anatally.inp is used for analysis calculation.

(3) The script makes a new folder "outfiles" to store each calculated result.

(4) The script executes PHITS with phits_tmp.inp and varfile_j.inp, and then moves the results to folder "/outfiles/j/".

(5) The script executes PHITS using anatally.inp with `icntl=17`. The result of the analysis is output in a file named by `sife` in anatally subsection of phits.inp.

6.9.2.4 Output format

In the file specified by `sfile`, in addition to mean values of tally results and their statistical errors, systematic uncertainties u_{syst} and total uncertainties u_{tot} are output. The total uncertainties are calculated by,

$$u_{\mathrm{tot}}^2 = u_{\mathrm{syst}}^2 + u_{\mathrm{stat}}^2/N. \tag{31}$$

Here, N is the number of samples, and the statistical errors are calculated as square root of the second term of the above right side. See the paper[111] for more details.

For example, when `axis=x, part=all` the output is as follows:

```
h: n           x           y(all     ),hh0l   n
#  x-lower     x-upper     all         r.err
   2.0000E-01  4.0000E-01  7.9333E-03  1.3865  1.1539  0.7686
   4.0000E-01  6.0000E-01  1.1495E-02  1.3951  1.2479  0.6237
            .........
```

In `all` column, the mean values of the tally result are output. In the right columns, the results of the analysis function are output as relative values in order of u_{tot}, u_{syst}, and σ_{stat}. Note that when the systematic uncertainty estimated by ANOVA is negative, 0 is output.

When this function is used on `[t-dchain]` results, only the total uncertainties are output to the file specified by `sfile`. By replacing the statistical errors output in the usual t-dchain tally files with the total uncertainties, the propagation of the total uncertainties, including the systematic uncertainties, can be evaluated when running DCHAIN.

[111] S. Hashimoto and T.Sato, Estimation method of systematic uncertainties in Monte Carlo particle transport simulation based on analysis of variance, J. Nucl. Sci. Technol. 56, 345-354 (2019).

6.10 Extended statistical indicators

The statistical error (standard deviation) is an important statistical indicator to confirm the reliability of the mean of tally results obtained by the Monte Carlo simulation. However, when the calculation has just started and the statistics are small, or when using options such as `[weight window]` that are prone to statistical bias, the values of the mean and statistical error can change significantly, so the statistical error alone may not be sufficient as an indicator to confirm the reliability. Therefore, the purpose of this function is to confirm that highly reliable results have been obtained by outputting various statistical indicators and checking their values.

In addition to outputting the transition of the mean and the statistical error as a function of the total history number (batch number), it also outputs the variance of the variance (VOV), which indicates the degree of variation of the statistical error, and the figure of merit (FOM), which indicates the degree of decrease of the statistical error with respect to the computation time. It also outputs a statistical check sheet, which lists whether these values are statistically reliable, and a probability density function (PDF), which is the distribution of the obtained tally results.

Note that in version 3.34, only `[t-track]` (except for `mesh=tet`) and `[t-point]` tallies are available. Also, restart calculations are not supported. (The mean and statistical error normally output can be added to the statistics by the restart calculation, but the results of VOV, FOM, PDF and the result of the statistical check sheet are reset at the restart calculation.)

6.10.1 How to use

When using this function, please set the following parameters:

- Set `itall=3` in the `[parameters]` section.
- Set `anatally` subsection in the tally section where you want to use this function.
- Set `iextstat` parameter in the tally section. (Note that this should be set outside the `anatally` subsection.)
- (In the case of `iextstat=2`,) `prodenmn, prodenmx, nbproden` are set in the tally section. (Note that these should also be set outside the `anatally` subsection.)

After making these settings, you can change the statistical indicators to be output by changing `iextstat`.

A) (If `iextstat=0`) Output the transition of the mean and the statistical error as a function of the total history number (batch number).

B) (When `iextstat=1`) In addition to the contents of A), the transition of VOV and FOM and a statistical check sheet are also output.

C) (In the case of `iextstat=2`) In addition to the contents of B), a PDF result is output. In this case, run a test calculation to check the range of tally results, then set the output range and run the main calculation. For details, please see the C) paragraph of Sec. 6.10.2.

When this function is used, a file is made with `_StD` in front of the file name extension specified in the tally section, and at the end of each batch, the respective statistical indicators for the contents of A), B), and C) are output to that file. The file size can be huge, as it will output a total of 2, 5, or 6 pages of eps files in each case. If you have a large number of x, y, z, or energy mesh bins, please limit the number of these bins to be output by setting parameters in the anatally subsection shown below. Also note that as you increase the output content from A), B), to C), the load on the computer, such as memory space used, will increase.

Example 59: Anatally subsection to specify the bin number to be output.

```
1:  anatally start
2:  ix = 89 91
3:  iz = 61 81 101
4:  iy = 1
5:  ipart = 1
6:  anatally end
```

"anatally" subsection should be defined between the lines of `anatally start` and `anatally end` written in the tally section. If only these two lines are written and no mesh bins are specified, the results for all bins will be output. (When using the output function of the extended statistical indicators, set the anatally subsection with only two lines, even if no bins are specified.) `ix,iz,iy` are parameters to specify the bins of the mesh for each variable defined in `x-type, z-type, y-type` respectively, in this example, the 89th and 91st bins for x are specified. The range is all combinations of each mesh, and in the example above, we specify 2 bins for x and 3 bins for z. Therefore, for a total of $2 * 3 = 6$ tally results, the statistical indicators will be output. The `ipart=1` means that only the first particle of the particles defined by `part=` will be output, for example, if `part = proton neutoron`, only the tally results for the proton will be output.

There are the following ten kinds of parameters to specify the mesh point: `ireg`: region (cell), `ix, iy, iz, ir, ie, it, ia: x-, y-, z-, r-, e-, t-, a-type` subsections, `ipart: part` parameter, and `imul: multiplier` parameter. When a parameter is not specified, results at all mesh points about its parameter are output. `all` can be used to output results at all points. Note that `{ n1 - n2 }` or `( )` cannot be set.

Note that the anatally subsection is also set up when using the function to analyze two (or more) tally results in Sec. 6.9, but the parameters that can be used and the concept is different from the `itall=3` case described here.

Example 60: An example using the function of the extended statistical indicators.

```
 1:  [parameters]
 2:  icntl=0
 3:  maxcas = 1000
 4:  maxbch = 100
 5:  itall =3
        .........
        .........
 6:  [t-track]
 7:  mesh = xyz
        .........
        .........
 8:  file = track.out
 9:  epsout = 2
10:  iextstat = 2
11:  prodenmn = 1E-5
12:  prodenmx = 1E-2
13:  nbproden =   50
14:  anatally start
15:     ix = 10
16:     iy = 10
17:     iz = 50 100
18:     ipart = 1
19:  anatally end
```

By setting `itall=3` in `[parametars]`, the extended statistical indicators will be output for the tally results of `[t-track]` for which the anatally subsection is set. In the example above, the "track_StD.out" file is made and figures for the results are shown in the "track_StD.eps" file. Depending on the bin numbers specified in the anatally subsection, the results are output for the 10th bin for x and y, the 50th and 100th bin for z, and the 1st particle for part. Since `iextstat=2`, there are 6 pages of output for each bin combination, with results for `iz=50` being output on pages 1 through 6 and results for `iz=100` being output on pages 7 through 12. Also, for the probability density function (PDF) of the tally results output in the case of `iextstat=2`, the lower and upper limits of the PDF range and the number of divisions are set by `prodenmn, prodenmx, nbproden` respectively. In the example above, the range from 10^{-5} to 10^{-2} is divided into 50 bins and the PDF is output.

6.10.2 Calculation method and output examples for each statistical indicator

As indicated in the previous section, this function can change the three types of output: A), B), and C). This section summarizes the calculation method for each of these statistical indicators, as well as examples of output and how to analyze the results of the indicators.

A) transition of the mean and the statistical error as a function of the total history number

In this case, PHITS outputs the mean and the statistical error. A tally result for the ith history is denoted by x_i, the mean $\bar{x}$ of the tally results by a total of N number of histories,

$$\bar{x} = \frac{1}{N} \sum x_i \tag{32}$$

where $\sum$ means summing from $i = 1$ to N. Note that the weight value is always set to 1 for simplicity. The statistical error is given as,

$$\sigma = \sqrt{\frac{\sum x_i^2 - N\bar{x}^2}{N-1}}. \tag{33}$$

This is the standard error of the mean σ and PHITS outputs the ratio $R = \sigma/\bar{x}$ to the mean $\bar{x}$ as the statistical error. If the calculation is statistically unbiased, the mean converges to a constant value as the number of histories increases, and the statistical error decreases by $1/(sqrtN$ for the total number of histories N.

In the case of A), the results of the mean (with statistical error indicated by error bars) and the statistical error are output on page 1 and 2 of the output eps file, respectively. The horizontal axis is the total number of batches, and the results for the total number of batches up to that point are shown at the end of each batch. Figure 6.1 shows an example of the output of this function for a 1 m thick concrete shielding irradiated with a 10 MeV neutron beam and tallying the neutron fluence on the opposite side of the irradiated surface. The mean and the statistical error are plotted by `maxcas`= 10^4, and we can see the transition of these results as the total number of histories (batch numbers) increase. From the the left panel, we can see that the mean changes significantly up to 40 batches, but gradually approaches a constant value after that. Next, the results in the right panel show that after 40 batches, the statistical error shows a decreasing trend, decreasing in proportion to the inverse of the square root of the number of histories.

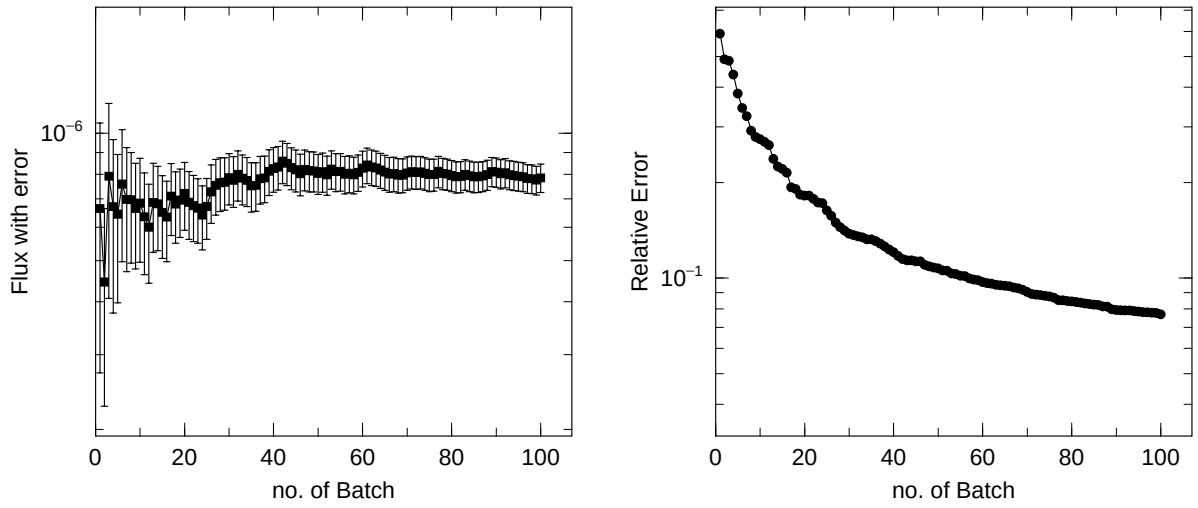


Figure 6.1: Mean and statistical error as a function of the total number of histories (total number of batches).

B) Statistical check sheet and transition of the mean, statistical error, variance of the variance (VOV), and figure of merit (FOM) as a function of the total history number

In this case, VOV and FOM are added to the contents of A) and output as well as a statistical check sheet to check the reliability of these statistical indicators.

VOV is a statistical indicator of the degree of variance of statistical errors and is given as,

$$\mathrm{VOV} = \frac{\sum (x_i - \bar{x})^4}{\{\sum (x_i - \bar{x})^2\}^2} - \frac{1}{N}. \tag{34}$$

Here, rewriting the mean $\bar{x}$ in this equation by Eq. (32), VOV is given as follows:

$$\mathrm{VOV} = \frac{\sum x_i^4 - \frac{4}{N}\sum x_i \sum x_i^3 + \frac{8}{N^2}(\sum x_i)^2 \sum x_i^2 - \frac{4}{N^3}(\sum x_i)^4 - \frac{1}{N}(\sum x_i^2)^2}{\left\{\sum x_i^2 - \frac{1}{N}(\sum x_i)^2\right\}^2}. \tag{35}$$

PHITS calculates VOV using this expression. From the first term on the right-hand side, we can see that it has a dependency of $1/N$ as well as the second term on the right-hand side, since the numerator is the sum of N and the denominator is the square of the sum of N. Thus, when the statistic is sufficient, VOV will decrease by $1/N$.

FOM is a statistical indicator of the degree of decrease in statistical error relative to computation time, using the ratio R of the standard error σ to the mean $\bar{x}$ and the computation time T [min]:

$$\mathrm{FOM} = \frac{1}{R^2 T}. \tag{36}$$

Since R is inversely proportional to the square root of N, the value of FOM will be constant if T and the total number of histories N show a proportional relationship during a stable numerical calculation. Conversely, if the FOM changes significantly before or after a certain history, the proportional relationship between T and N has disappeared, and an anomaly such as a very long calculation time being required for that history can be confirmed.

Statistical check sheet at batch 100

no. = 1, reg = 11

Quantity	Output	Value	Desired	Pass
Mean	Current value	7.8418E-07	Constant	yes
Mean	History dep. (rel. slope)	-0.0162	= 0.00	
Relative error	Current value	0.0769	< 0.05	no
Relative error	History dep. (p in N^p)	-0.5053	= -0.50	yes
Variance of the variance	Current value	0.0133	< 0.10	yes
Variance of the variance	History dep. (p in N^p)	-1.0189	= -1.00	yes
Figure of Merit	Current value	3.8531E+01	Constant	yes
Figure of Merit	History dep. (rel. slope)	-0.0158	= 0.00	

Figure 6.2: Statistical check sheet at the end of 100 batches.

In the case of B), PHITS outputs the statistical check sheet on the first page, and output the transition of the mean, statistical error, VOV, and FOM as a function of the total number of histories on the second, third, fourth, and fifth pages, respectively. Figure 6.2 shows the statistical check sheet for the same calculation as in Fig. 6.1, where the column Quantity shows the type of statistical indicator and the column Output shows the type of value to be output. The Value column shows the value at the end of the batch (current value) and the value of dependence on the total number of histories (history dep.). The pass column shows the result of determining whether or not the criteria are met. (The method of evaluating the dependence on the total number of histories and the conditions for satisfying the criteria are explained in the next section.) The results of Figure 6.2 show that all but the criterion for the statistical error (< 0.05) is satisfied, indicating that a stable calculation is being performed. However, since the statistical error is the most important indicator, it is desirable to increase the statistics to the extent that the criterion is met. Figure 6.3 shows VOV and FOM as a function of the total number of histories. From the left panel, we can see that VOV shows a decreasing trend after 40 batches, decreasing at $1/N$. On the other hand, from the results in the right panel, we can see how FOM converges to a constant value after 40 batches.

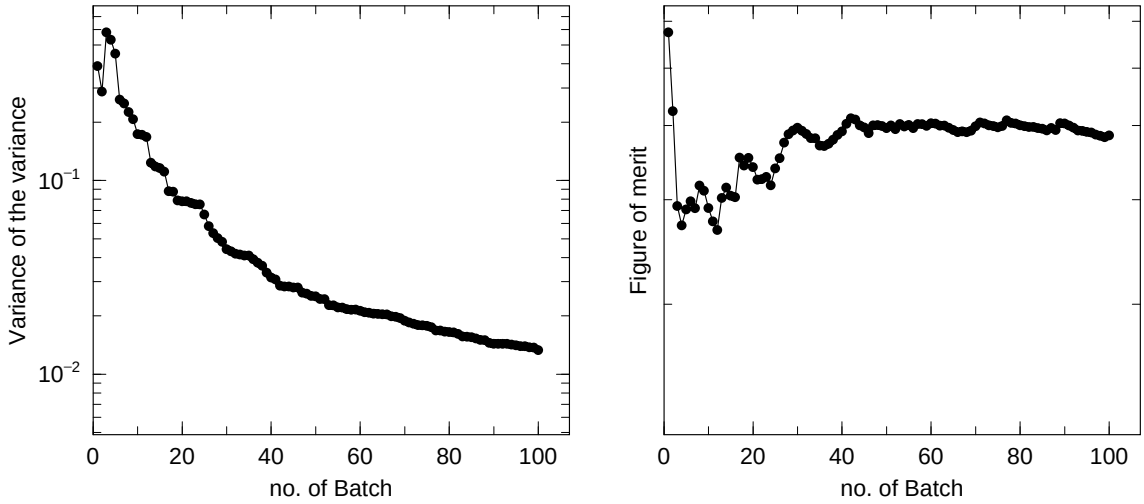


Figure 6.3: Variance of variance (VOV) and figure of merit (FOM) as a function of the total number of histories (batches).

C) Statistical check sheet, transition of mean, statistical error, VOV, and FOM as a function of the total history number, and probability density function (PDF)

In this case, in addition to the contents of B), a PDF representing the distribution of the obtained tally results is output. Please note that there are two modes of calculation: a test calculation mode for confirming the range of output PDF, and a main calculation mode for explicitly specifying that range and obtaining the PDF. Basically, first confirm the range in the test calculation mode, and then perform the main calculation by specifying the lower (`prodenmn (D=1e-2)`) and upper (`prodenmx (D=1e+2)`) limits of the PDF range and the number of divisions (`nbproden (D=100)`).

For PDF calculations, the range of the PDF output $[x_{\min}, x_{\max}]$ and the number of divisions $n_{\rm pdf}$ on a logarithmic scale are determined, and when the tally result x_i for each history is obtained, which bin number it falls into is checked and the count number for that bin is increased. Finally, the output is normalized so that the distribution integrates to 1. Here, if the tally result is 0, it is not counted, so the peak position of the distribution and the mean $\bar{x}$ may be shifted.

The test calculation mode works when none of the parameters `prodenmn, prodenmx, nbproden` for PDF output are specified. In test calculation mode, calculations are started with initial values of $x_{\min} = 10^{-2}$, $x_{\max} = 10^{2}$, and $n_{\rm pdf} = 100$, and if tally results are outside of the range $[x_{\min}, x_{\max}]$, the upper or lower limit of the range is changed according to the size of the result. For example, if a tally result of 10^3 is obtained, the upper limit is 10 times the tally result and $x_{\max} = 10^4$. On the other hand, if a smaller tally result is obtained, then 1/10 of that value is used as $x_{\min}$. In this way, by confirming the distribution of the final output PDF, we can see the range that the tally result can take. Note that if the tally result differs from the initial value $x_{\min} = 10^{-2}, x_{\max} = 10^{2}$ by several tens of digits, the range of the horizontal axis of the output PDF will expand by that number of digits. In such a case, please adjust manually while specifying the output range in the main calculation mode shown below. Note that in the test calculation mode, the stored PDF data is reset each time the range changes due to adjustment, so results for a different number of histories are output than for other statistical indicators. Parallel computation by MPI or OpenMP cannot be used for the test calculation.

Next, the main calculation mode operates if any one of `prodenmn, prodenmx, nbproden` is specified. Parameters not explicitly specified will be set to their respective default values. In this calculation mode, the range of tally results to be stored does not change during the PHITS running, and a PDF of the tally results is output. However, if a tally result is larger than the upper limit of the set range, a warning message is output; the message output stops after about 5 times (when parallel calculation is not used), but if this message appears, stop the calculation once and increase the upper limit (`prodenmx`) and run the main calculation mode again. In this calculation mode, parallel calculation by MPI or OpenMP can be used.

In the case of C), pages 1 through 5 are the same as in B). In addition, the PDF at the end of the batch is output on page 6. Figure 6.4 shows the PDF for the same calculation as in Fig. 6.1. Although the statistics are not sufficient to determine a normal distribution, we can see that the distribution has a peak around 10^{-3} and the tally

results in values within the range of approximately 10^{-4} to 10^{-2}. The mean of this tally result is about 10^{-6}, which is about 3 orders of magnitude different from the peaked value of 10^{-3}, indicating that the percentage of non-zero tally results is about 1/1000.

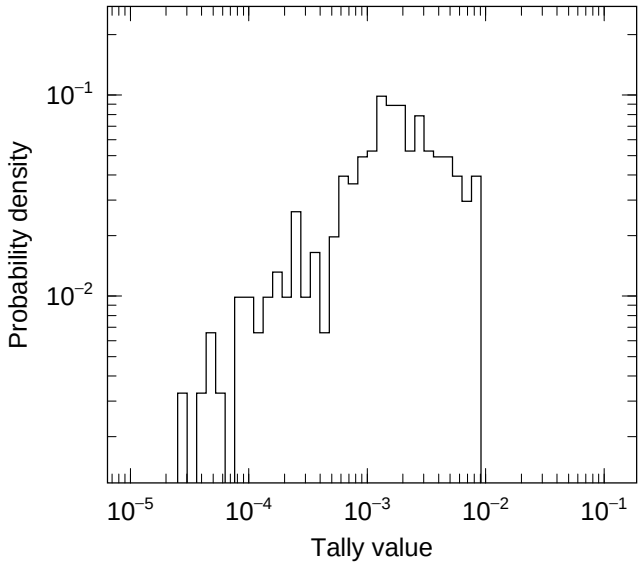


Figure 6.4: Probability density function (PDF) of tally results.

The PDF is an indicator that allows direct confirmation of the specific numerical values of the tally results, making it easier to detect anomalies, for example, having values in areas outside the distribution (especially toward larger values). Note that to confirm a clean distribution structure, a larger number of histories is required than for other statistical indicators. For this reason, PDFs are not considered in the statistical check sheet.

6.10.3 Statistical check sheet

This check sheet outputs the current value at the end of the batch for the mean, statistical error, variance of the variance (VOV), and figure of merit (FOM), and their dependence on the total number of histories (history dep.) and a total of six checkpoints are judged yes/no by comparing them to criterion values.

The dependence on the number of histories is evaluated from the results of the second half of all histories, e.g., when calculations of 100 batches are finished, the values from batch 51 to 100 are used. Since the mean and FOM are independent of the total number of histories, i.e., converge to a constant value, a function $f(N_{\mathrm{batch}}) = aN_{\mathrm{batch}} + b$ (where N_{batch} is the total number of batches) is assumed and fitted by the least squares method, and its coefficient a is close to 0 or not. Specifically, if the absolute value of this coefficient normalized by the value at the last batch is less than 0.05, it is judged as `yes`; if it is greater than 0.05, it is judged as `no`. The statistical error is proportional to the inverse of the square root of the total number of histories, so the function $f(N_{\mathrm{batch}}) = b\exp(aN_{\mathrm{batch}})$ is assumed and fitted by the least squares method, and judged if the coefficient a is close to -0.5. Specifically, if it is in the range of -0.9475 to -0.525, it is assumed to be yes. For VOV, since it is inversely proportional to the total number of histories, the least squares method is performed assuming a function similar to that for statistical errors, and it is determined if the coefficient a is close to 1.0 or not. Specifically, if it is in the range of -0.95 to -1.05, it is assumed to be yes.

Please note that the criterion values used to determine PASS in this checklist are not absolutely reliable. Depending on the calculation conditions and the desired accuracy of the calculation, the criterion values for judging the reliability of the calculation results may change; just because all six are yes does not mean that the results are always reliable. On the other hand, even if an item is no, you may be able to determine that there is no problem by checking the transition of that statistical indicator as the total number of histories increases.

7 Tally section format

7.1 [T-Track] section

This tally can be used to obtain the fluence in any specified region. In this tally, the track length is evaluated whenever particles pass through a specified region, as shown in Fig. 7.1, and the sum of the track lengths in units of [cm] is scored. Based on this, a particle fluence in units of [/cm^2/source] is determined by dividing the scored track lengths by the volume of the region and the number of source particles.

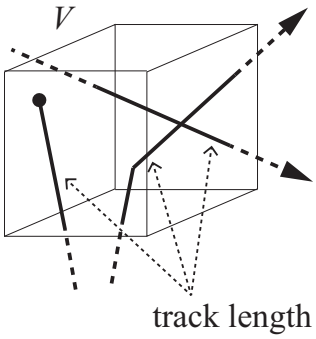


Figure 7.1: `[T-Track]` tally: track length (solid line) is calculated.

As an example, information on the detector response in a specified region can be obtained by utilizing this tally. Multiplying the fluence by the cross section (in units of cm^2) of the detector enables estimation of the number of counts in the response.

Table 7.1: `[t-track]`parameters (1)

name	value	explanation
`mesh =`	`reg, r-z, xyz, tet`	Mesh type: a mesh type subsection is required below this option.
`volume`	(optional)	This option defines the volume for each region for `reg` mesh. Volume definitions are required below this option. Values defined in `[volume]` are used in the case of no definition. If special description such as using `( )` is used to specify a region in `reg=`, its internally defined region number is output in input echo when this volume subsection is not defined.
`reg vol`		Volume definition. For details see Sec. 6.1.2.
`iechrl =`	`72` (default)	Maximum number of columns for volume input echo.
`e-type =`	`1, 2, 3, 4, 5`	Energy mesh: an energy mesh subsection is required below this option.
`t-type =`	`1, 2, 3, 4, 5` (optional)	Time mesh: a time mesh subsection is required below this option.
`a-type =`	`1, 2, -1, -2`	Angle mesh with θ degrees of freedom for `mesh=r-z`: an angle mesh subsection is required below this option.

Table 7.2: [t-track]parameters (2)

name	value	explanation
part =	all (default), particle name	Tally particle.
axis =	eng, reg, x, y, z, r tet, rad, deg, xy, yz, xz, rz t	x axis value of output data. x axis value of output data (only active with mesh=tet). x axis value of output data（variables for θ when mesh=r-z). 2-dimensional. Time axis.
samepage =	(optional, D=part)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in axis can be specified.
file =	file name	Define file names. This is required by each setting of axis.
resfile =	(optional, D=file)	Define a file name of the past tally in the restart calculation: even if several axis parameters were defined, specify only one resfile.
unit =	1, 2, 3, 4 11, 12, 13, 14	1: [1/cm^2/source] 2: [1/cm^2/MeV/source] 3: [1/cm^2/Lethargy/source] 4: [cm/source] 11: [1/cm^2/nsec/source] 12: [1/cm^2/MeV/nsec/source] 13: [1/cm^2/Lethargy/nsec/source] 14: [cm/nsec/source]
factor =	(省略可、D=1.0)	normalization factor

'Lethargy' in unit=3 or 13 is the natural logarithmic unit of energy defined by $\ln(E_{\mathrm{ref}}/E)$ using a reference energy E_{ref} and the particle energy E. Setting these units enables obtaining results in units of Lethargy, which are given as Lethargy widths, $\ln(E_{\mathrm{high}}/E_{\mathrm{low}})$, for each energy bin in the energy mesh subsection. Here E_{high} and E_{low} are the maximum and minimum values of the energy bins, respectively.

Setting unit=1,2,3,11,12 or 13 produces the mean particle fluence in a specified region calculated from the sum of the track lengths per source divided by the volume of the region. Note that, for reg mesh the volume in the [volume] section must be set; if this is not done, the particle fluence for volume=1 [cm^3], i.e., the sum of the track lengths per source, is obtained. For r-z and xyz meshes, the volume is automatically calculated. Setting unit=4 or 14 produces the sum of the track lengths per source.

θ degrees of freedom can be set in r-z mesh of this tally. The a-type mesh definition used to specify θ follows the same format as in the other angle mesh cases. The θ dependence of results can be obtained by setting axis=rad or axis=deg, where rad and deg represent θ in radians and degrees, respectively.

Table 7.3: [t-track]parameters (3)

name	value	explanation
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
2d-type =	1, 2, 3, 4, 5, 6, 7 (optional, D=3)	Options for 2-dimensional plot.
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
rshow =	0 (default), 1, 2, 3	When mesh=reg,tet and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.
gslat =	2(default), = 0 = 1 = 2 = 3	Option to draw lattice or tetrahedral geometry boundary lines when gshow or rshow is specified. Not drawing. Drawing. Not that the line of voxel phantoms may not be drawn clearly. Not drawing the lines in the same cell. Not drawing the lines in the same material. In this case, the boundary of two adjacent cells filled with the same material is not also drawn, even if not lattice or tetrahedral geometry. When gshow≥ 3, cell and lattice numbers are now shown.
x-txt = y-txt = z-txt =	(optional) (optional) (optional)	*x* axis title. *y* axis title. *z* axis title.
resol = width =	1 (default) 0.5 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness. The option defines the line thickness.
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.

Table 7.4: [t-track]parameters (4)

name	value	explanation
material =	(optional) all, number of materials	Specify materials for scoring. all : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring.
(next line)	2 5 8	Material numbers.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
multiplier =	Number of material (optional)	Multiplier for each material: a multiplier subsection is required below this option. See Sec. 5.24.1 for its detailed usage.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
bmpout =	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkout =	0(default), 1	Output the tally results in the xyz-mesh in the input format of ParaView. This file is named by replacing the extension with ".vtk." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkfmt =	0(default), 1	Format of output file for ParaView. 0: ascii, 1: binary.
foamout =	0(default), 1, 2	When foamout=1, generate a file of outputs (numerical data) with OpenFOAM field data format. This file is named by replacing the extension with ".foam." When foamout=2, generate a file of outputs (element no., center of mass x coordinate, y coordinate, z coordinate, volume, output data, relative error) with CSV format. This file is named by replacing the extension with ".csv." When mesh=tet and axis=tet, it is available.
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.

Table 7.5: [t-track]parameters (5)

name	value	explanation
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
stdcut =	(optional, D=-1)	Threshold value of STD cut off.
iextstat =	0 (default),1,2	Option for extended statistical indicators. 0: Output the transition of the mean and statistical error as a function of the total history number (batch number). 1: In addition to the contents of iextstat=0, output variance of the variance (VOV) and figure of merit (FOM) as a function of the total history number (batch number), and output a statistical check sheet. 2: In addition to the contents of iextstat=1, the probability density distribution (PDF) of the tally results is output. In this case, run a test calculation to check the range of tally results, then set the output range and run the main calculation.
prodenmn =	(optional, D=1e-2)	Lower limit of the PDF output range.
prodenmx =	(optional, D=1e+2)	The upper limit of the PDF output range.
nbproden =	(optional, D=100)	Number of divisions in the PDF output (logarithmic scale).

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

To output the extended statistical indicators, set itall=3 in [parameters], set the anatally subsection in the [t-track], and specify iextstat. The anatally subsection should be defined between the lines of anatally start and anatally end written in the tally section. If you output a PDF with iextstat=2 and you know the range of possible tally results, specify the PDF output range with the prodenmn, prodenmx, nbproden parameters and run PHITS in the main calculation. If you do not know the range, please run PHITS in a test calculation mode without specifying these three parameters and check the range before proceeding to the main calculation. Note that iextstat, prodenmn, prodenmx, nbproden should be specified outside the anatally subsection. See Sec. 6.10 for more detail.

7.2 [T-Cross] section

This tally can be used to obtain the current or flux (actually, the fluence) on any specified surface. In this tally, a particle crossing the surface simply adds 1 to the current and $1/\cos\theta$ to the flux, where θ is the angle between the direction of the particle trajectory and the normal vector to the surface. In PHITS, the current and flux are similar but distinct physical quantities; they differ in terms of the surface element used to calculate the number of the particles crossing per unit area. The current is evaluated by dividing by the area of the surface S shown in Fig. 7.2; by contrast, the flux is calculated by dividing by $S\cos\theta$. The value of S is given in the mesh type subsection as `area` for `mesh=reg` and is calculated automatically for the `r-z` and `xyz` meshes.

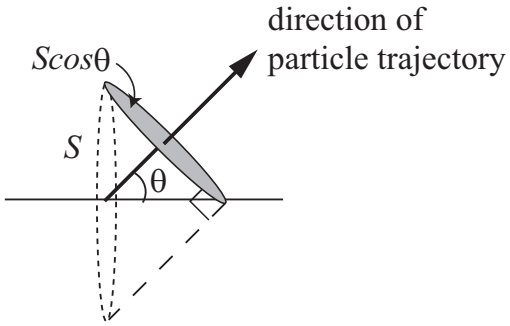


Figure 7.2: Relation between the areas S and $S\cos\theta$.

As the flux in this tally is evaluated by weighting by $1/\cos\theta$, the result is equivalent to that obtained from the `[t-track]` tally for an extremely thin region. Consequently, information on the detector response in the specified surface can be obtained from the `[t-cross]` tally. Multiplying the flux by a cross section (in the unit of cm^2) of the detector enables estimation of the number of counts in the response.

If a boundary of geometry coincides with a tally plane at cylindrical surface in `r-z` mesh or at a rotated surface in xyz mesh, the tally results might be underestimated due to the problem of significant digit. In such cases, the mesh size should be changed slightly so that it does not coincide with the boundary plane.

Table 7.6: [t-cross] parameters (1)

name	value	explanation
mesh =	reg, r-z, xyz	Mesh type: a mesh type subsection is required below this option.
part =	all (default), particle name	Tally particle.
e-type =	1, 2, 3, 4, 5 (optional)	Energy mesh: an energy mesh subsection is required below this option.
eng2let =	(optional)	0 (default): Do not convert energy defined by e-type to LET. 1: Convert energy to LET (keV/μm). Set eng2let=2 and axis=let when you want to output the results as a function of LET.
t-type =	1, 2, 3, 4, 5 (optional)	Time mesh: a time mesh subsection is required below this option.
a-type =	1, 2, -1, -2 (optional)	Angle mesh (=1,2; cosine mesh, =-1,-2; angle (in units of degree) mesh): an angle mesh subsection is required below this option for output=a-curr or =oa-curr.
iangform =	0(default),1,2,3	Select the angle in a-type mesh. 0: The angle between the normal vector to the surface and the directon of the particle trajectory (=θ). 1: The angle between the x-axis and the directon of the particle trajectory. 2: The angle between the y-axis and the directon of the particle trajectory. 3: The angle between the z-axis and the directon of the particle trajectory.

The current for a specified angle can be obtained using the angle mesh shown in Fig. 7.3. In cases where unit=4,5 6,14,15,16, the output is given as a quantity per solid angle (in steradians) calculated using the mesh size of the angle-bin defined in the angle mesh subsection.

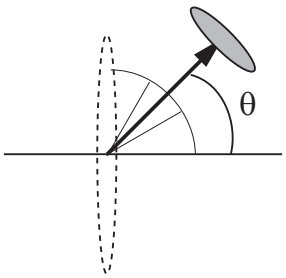


Figure 7.3: Schematic of tally using angle mesh.

'Lethargy' in unit=3,6,13,16 is the natural logarithmic unit of energy defined by $\ln(E_{\rm ref}/E)$ using a reference energy $E_{\rm ref}$ and the particle energy E. Setting these units enables obtaining results in units of Lethargy, which are given as Lethargy widths, $\ln(E_{\rm high}/E_{\rm low})$, for each energy bin in the energy mesh subsection. Here $E_{\rm high}$ and $E_{\rm low}$ are the maximum and minimum values of the energy bins, respectively.

In unit=4,5,6,14,15,16, 'sr' denotes steradians as the solid angle unit.

Table 7.7: `[t-cross]` parameters (2)

name	value	explanation
`unit =`	`1, 2, 3, 4, 5, 6`	1: [1/cm^2/source] 2: [1/cm^2/MeV/source] 3: [1/cm^2/Lethargy/source] 4: [1/cm^2/sr/source] 5: [1/cm^2/MeV/sr/source] 6: [1/cm^2/Lethargy/sr/source]
	`11,12,13,14,15,16`	11: [1/cm^2/nsec/source] 12: [1/cm^2/MeV/nsec/source] 13: [1/cm^2/Lethargy/nsec/source] 14: [1/cm^2/sr/nsec/source] 15: [1/cm^2/MeV/sr/nsec/source] 16: [1/cm^2/Lethargy/sr/nsec/source]

`axis =`	`eng, let, reg, x, y, z, r`
	`cos, the, t`
	`xy`
	`yz, xz, rz`
		x axis value of output data.
		Angle (cosine, θ[degree]) and time mesh.
		2-dimensional.
		2-dimensional, only for `enclos`=1.

`samepage =`	(optional, `D=part`)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in `axis` can be specified.
`file =`	`file name`	Define file names. This is required by each setting of `axis`.
`resfile =`	(optional, `D=file`)	Define a file name of the past tally in the restart calculation: even if several `axis` parameters were defined, specify only one `resfile`.
`factor =`	(optional, D=1.0)	Normalization factor.
`title =`	(optional)	Title.
`angel =`	(optional)	ANGEL parameters.
`sangel =`	(optional)	Special format for ANGEL parameters.
`2d-type =`	`1, 2, 3, 4, 5, 6, 7` (optional, D=2)	Options for 2-dimensional plot.

Table 7.8: [t-cross] parameters (3)

name	value	explanation
output =	flux	Flux at the surface.
	current	Current at the surface.
	f-curr	Forward current at the surface.
	b-curr	Backward current at the surface.
		The followings are the options that are not recommended to use after version 3.34.
	o-curr	Current without e-type.
	of-curr	Forward current without e-type.
	ob-curr	Backward current without e-type.
	a-curr	Current with a-type.
	oa-curr	Current with a-type but without e-type.
	a-flux	Flux with a-type.
	oa-flux	Flux with a-type but without e-type.
x-txt =	(optional)	*x* axis title.
y-txt =	(optional)	*y* axis title.
z-txt =	(optional)	*z* axis title.
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
resol =	1 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness.
width =	0.5 (default)	The option defines the line thickness.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy).
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.

The output options output=f-curr, b-curr, of-curr, ob-curr can be used in either xyz or r-z meshes. Note that in xyz meshes these options are available only for the *z*-direction.

Table 7.9: [t-cross] parameters (4)

name	value	explanation
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
dump =	Number of data	For mesh=reg, the information is dumped on the file. If dump is negative, data are written by ascii, if positive, by binary.
(next line)	Data sequence	Define the data sequence. The history information (nocas and nobch) is necessary to use idmpmode=1.
letmat =	(optional)	(Effective only when eng2let=1) Material id for LET(dE/dx). If not defined, a real material is assumed. If a material not used in the geometry is selected, its material density must be defined in [material]. To calculate LET in water, define water with 1 g/cm^3 in [material]. When letmat<0 is set, PHITS automatically calculates dE/dx for water with 1 g/cm^3 for electrons and positrons: please see "ParticleTherapy" in the recommendation setting for more details.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: no.
stdcut =	(optional, D=-1)	Threshold value of STD cut off.
multiplier =	Number of material (optional)	Multiplier for each material.[112] Multiplier subsection is required below this parameter. See Sec. 5.24.1 for its detailed usage.
enclos =	0(default)	Define crossing surfaces on the z-axis and the r-axis for mesh=r-z, and define crossing surface on the z-axis for mesh=xyz.
	1	Define crossing surfaces by enclosing surfaces of areas divided by mesh type. All surfaces enclosing an area are treated as a crossing surface.

In the [t-cross] tally, the dump option can only be used with reg meshes and only on axis=reg. If the dump option is set, the e-type, a-type and t-type meshes take on only the maximum and minimum values. The output option can be set as current, a-curr, or oa-curr. In using this dump parameter, axis and file are restricted to one axis and one file apiece and unit is always 1. The dumped data are written onto a file named "***_dmp", where "***" indicates the file name specified by file=***. The normal output of the tally is written on "***." From this file, information on the total normalization factor can be obtained; doing so requires setting one mesh each for e-type, a-type and t-type. Note that constraints based on the history counter (chmin & chmax) do not work with the dump function because the particle information is output to the dump file immediately after passing through the tally surface. The history information (nocas and nobch) is necessary to use idmpmode=1 for continuous calculation using the dump file; in addition, both the dump file with "_dmp" and the normal output file specified by file= are required to use idmpmode=1. The option dumpall is not compatible with this dump tally option when shared memory parallelization is active.

When in or out is specified in [counter], the counter value will change after the particle information has been tallied in [t-cross]. Therefore, if the counter value before the change is not specified, the information at the moment of crossing the boundary plane will not be tallied.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

From ver. 3.10, crossing surfaces are defined by enclosing surfaces of areas divided by mesh type when enclos=1 is set with mesh=xyz or mesh=r-z. All surfaces enclosing an area are treated as a crossing surface. For

[112] From ver. 2.86, multiplier option can be used in [t-cross] tally.

example, in the case of mesh=xyz, each crossing surface on the z-axis is used as a tally surface when enclos=0, and six surfaces of a rectangular solid defined by the xyz-meshes is used as a tally surface when enclos=1. The forward direction is defined as the incoming direction, and the backward direction is defined as the outgoing direction. The total area of closed surface is used to calculate per unit area.

Setting mesh=reg for the mesh type in this section requires defining the crossing surface by outgoing (r-from) and incoming (r-to) region number, and the area of the surface (area in units of cm^2), as shown in the example below [113].

```
mesh = reg
 reg =  number of crossing surfaces
  r-from  r-to     area
    2       8      10.0
    3       8       5.0
  ( 4 5 ) ( 4 5 )   2.0
  (13<5)  (14<5)    7.0
  (13<6)  (14<6)    7.0
  (13<7)  (14<7)    7.0
   ...     ...     ....
   ...     ...     ....
```

In the next line of mesh=reg, give the number of crossing surfaces to tally by reg=. Furthermore, from the next line, the values of r-in r-out, and area should be written as a matrix format. The default order for this definition is r-in r-out area. The line of these column headers can be omitted. However, to change the order, rearrange and explicitly write the column headers as r-in r-out area. The skip operator non can be also used. When specifying the region number, the format (2 -5 8 9) can be used, as can the lattice and universe style (6 < 10[1 0 0] < u=3). However, any value that is not single numeric must be enclosed by ( ) .

If mesh=reg is set, the obtained current or flux is unidirectional from r-from to r-to; a bidirectional flux can be set in the third line of the above definition.

Setting the mesh=r-z defines the numbers of two crossing surface types: the number of "nz+1" crossing surfaces for z defined by $r_i - r_{i+1}$ and the number of "nr+1" crossing surfaces for r defined by $z_i - z_{i+1}$. If an r-surface coincides with the surface of the outer void, the flux on this surface is not tallied.

If mesh=xyz is set, the number of "nz+1" crossing surfaces on z are defined by $x_i - x_{i+1}$, and $y_j - y_{j+1}$. In this case, x and y crossing surfaces are not defined.

Setting mesh=r-z or xyz causes crossing particles to be detected in both directions on the defined surface. The forward direction is defined as the positive direction on a z surface and from the center to the exterior on an r surface. From ver. 3.05, specification of z-type=1 and nz=0 is allowed to calculate fluences of particles passing through a certain surface.

[113] Before ver. 2.96, r-in and r-out were used instead of r-from and r-to, respectively. These old parameters can be used after ver. 2.97. Note that 'in' and 'out' are reversed in this definition.

7.3 [T-Point] section

It is impractical to calculate particle fluence at a specific point or line using the [t-track] tally. To address this, the point estimator tally [t-point] was introduced to estimate these quantities within a short computational time. However, to use this tally the following conditions must be satisfied in the PHITS simulation using [t-point]:

(1) particle energy should not exceed the maximum energy of the data library used, i.e., dmax;

(2) only the fluence of neutrons and photons can be calculated by [t-point];

(3) neither event generator mode nor EGS5 should be used (e-mode=0, negs=0);

(4) the material should be uniform within a certain proximity to the point detector to avoid singularity;

(5) reflection or white boundary surface should not be used.

Please see the read-me file or the sample input file in "\phits\utility\tpoint\" for more details.

Table 7.10: [t-point] parameters (1)

name	value	explanation
point =	number of data	Option for point detectors: a subsection is required below this option.
ring =	number of data	Option for ring detectors: a subsection is required below this option.
part =	particle name (optional)	Tally particle. neutron photon are set when part is not specified.
e-type =	1, 2, 3, 4, 5	Energy mesh: an energy mesh subsection is required below this option.
t-type =	1, 2, 3, 4, 5 (optional)	Time mesh: a time mesh subsection is required below this option.
unit =	1, 2, 3 11, 12, 13	1: [1/cm^2/source] 2: [1/cm^2/MeV/source] 3: [1/cm^2/Lethargy/source] 11:[1/cm^2/nsec/source] 12:[1/cm^2/nsec/MeV/source] 13:[1/cm^2/Lethargy/nsec/source]
axis =	eng, t	x axis value of output data.
samepage =	(optional, D=part)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in axis can be specified. point and ring can be also specified.
file =	file name	Define file names. This is required by each setting of axis.
resfile =	(optional, D=file)	Define a file name of the past tally in the restart calculation: even if several axis parameters were defined, specify only one resfile.
factor =	(optional, D=1.0)	Normalization factor.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.

'Lethargy' in unit=3 or 13 is the natural logarithmic unit of energy defined by $\ln(E_{\rm ref}/E)$ using a reference energy $E_{\rm ref}$ and the particle energy E. Setting these units enables obtaining results in units of Lethargy, which are given as Lethargy widths, $\ln(E_{\rm high}/E_{\rm low})$, for each energy bin in the energy mesh subsection. Here $E_{\rm high}$ and $E_{\rm low}$ are the maximum and minimum values of the energy bins, respectively.

In [t-point], the number of points or rings (instead of the mesh, as in other tallies) must be defined. For example, point=3 should be specified to define 3 point detectors. The maximum number of points or rings per [t-point] is 20; to set more detectors, another [t-point] tally must be defined. Point and ring detectors cannot be combined in one [t-point] tally. The information on a point or ring must be defined in t successive lines following the definition of the point or ring parameter. The point detector definition is described as follows.

```
[ T-point ]
point =    1     # number of point detectors
non     x        y       z        r0
  1     10.0     0.0     50.0     1.0
```

where x, y, and z indicate the coordinates of the point detector and r0 is the radius of the fictitious sphere (for more information on the fictitious sphere, see the read-me file in "\phits\utility\tpoint\"). These parameters are given in units of cm.

The ring detector is defined as follows:

```
[ T-point ]
ring =    1     # number of ring detectors
non   axis    ar       rr       r0
  1    z      50.0     10.0     1.0
```

where axis indicates the direction of the ring axis specified as x, y, or z, ar is the distance from the origin to the center of the ring, rr is the ring radius, and r0 is the radius of the fictitious sphere. The order of these parameters can be changed by changing the order of notation, e.g., x y z r0 can be changed to z y x r0. Aside from these factors, the parameters defined in [t-point] are the same as those in [t-track], including the multiplier option. Thus, the radiation dose at a specific point can be estimated using [t-point]. However, material, two-dimensional plot options, and transforms cannot be specified in [t-point].

Table 7.11: [t-point] parameters (2)

name	value	explanation
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
stdcut =	(optional, D=-1)	Threshold value of STD cut off.
multiplier =	Number of material (optional)	Multiplier for each material: a multiplier subsection is required below this option. See Sec. 5.24.1 for its detailed usage.
iextstat =	0 (default),1,2	Option for extended statistical indicators. 0: Output the transition of the mean and statistical error as a function of the total history number (batch number). 1: In addition to the contents of iextstat=0, output variance of the variance (VOV) and figure of merit (FOM) as a function of the total history number (batch number), and output a statistical check sheet. 2: In addition to the contents of iextstat=1, the probability density distribution (PDF) of the tally results is output. In this case, run a test calculation to check the range of tally results, then set the output range and run the main calculation.
prodenmn =	(optional, D=1e-2)	Lower limit of the PDF output range.
prodenmx =	(optional, D=1e+2)	The upper limit of the PDF output range.
nbproden =	(optional, D=100)	Number of divisions in the PDF output (logarithmic scale).

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

To output the extended statistical indicators, set itall=3 in [parameters], set the anatally subsection in the [t-point], and specify iextstat. The anatally subsection should be defined between the lines of anatally start and anatally end written in the tally section. If you output a PDF with iextstat=2 and you know the range of possible tally results, specify the PDF output range with the prodenmn, prodenmx, nbproden parameters and run PHITS in the main calculation. If you do not know the range, please run PHITS in a test calculation mode without specifying these three parameters and check the range before proceeding to the main calculation. Note that iextstat, prodenmn, prodenmx, nbproden should be specified outside the anatally subsection. See Sec. 6.10 for more detail.

7.4 [T-Adjoint] section

This is the tally used in adjoint mode (`iadjoint=1`) calculations. The results from this tally are equivalent to the results from a `[T-Track]` tally in the corresponding forward mode calculation. See Sec. 5.2.9 and the sample input file in `\phits\utility\adjoint\` for more details on how to set up an adjoint simulation, and on how to set the `e-type` (energy mesh), `e-smin`, `e-smax` and `factor` parameters in this tally.

Table 7.12: `[t-adjoint]` parameters (1)

name	value	explanation
`mesh =`	`reg, r-z, xyz`	Mesh type: a mesh type subsection is required below this option.
`material =`	(optional) `all`, number of materials	Specify materials for scoring. `all`: default (same as no definition). Set the number of materials, then define the material numbers on the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring.
(next line)	`2 5 8`	Material numbers.
`e-type =`	`1, 2, 3, 4, 5`	Energy mesh: an energy mesh subsection is required below this option.
`m-source =`	(D=1)	Specify the number of multi-sources defined in the `[source]` section (maximum of 6 multi-sources supported). Columns of results (flux & r.err) will be written to the output file for each multi-source.
`e-smin =`	(D=`0.58`)	Minimum energy of the real photon source.
`e-smax =`	(D=`0.62`)	Maximum energy of the real photon source.
`t-type =`	`1, 2, 3, 4, 5` (optional)	Time mesh: a time mesh subsection is required below this option.
`unit =`	`1` (only)	[1/cm^2/source]

`axis =`	`eng, reg, x, y, z, r`
	`xy, yz, xz, rz`
	`t`
		x axis value of output data.
		2-dimensional.
		Time axis.

`file =`	`file name`	Define output file names. This is required by each setting of `axis`.
`resfile =`	(optional, D=`file`)	Define a file name of the past tally in the restart calculation: even if several `axis` parameters were defined, specify only one `resfile`.
`factor =`	(optional, D=`1.0`)	Normalization factor. Specify the emission rate of the real photon source here.
`title =`	(optional)	Title.
`angel =`	(optional)	ANGEL parameters.
`sangel =`	(optional)	Special format for ANGEL parameters.
`2d-type =`	`1, 2, 3, 4, 5, 6, 7` (optional, D=3)	Options for 2-dimensional plot.

`x-txt =`
`y-txt =`
`z-txt =`
	(optional)
	(optional)
	(optional)
		x axis title.
		y axis title.
		z axis title.

Table 7.13: [t-adjoint] parameters (2)

name	value	explanation
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy, yz, xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
rshow =	0 (default), 1, 2, 3	When mesh=xyz and axis=xy, yz, xz, region border (1), material name (2), region name (3) are plotted using this option. A xyz mesh section must be added below this option.
resol =	1 (default),	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness.
width =	0.5 (default)	This option defines the line thickness.
volume =	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined.
reg vol		Volume definition. For details see Sec. 6.1.2.
iechrl	72 (default)	Maximum number of columns for volume input echo.
volmat	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps". When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D=9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
gslat =	1 (default), 0	1: show lattice boundary in gshow, 0: do not show

Table 7.14: [t-adjoint] parameters (3)

name	value	explanation
stdcut =	(optional, D=-1)	Threshold value of STD cut off.
multiplier =	Number of material (optional)	Multiplier for each material: a multiplier subsection is required below this option. See 5.24.1 for its detailed usage.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

When using function to display transition of tally results and errors, mesh points to output tally results and these statistical errors can be specified by setting "anatally" subsection. This subsection should be defined between the lines of anatally start and anatally end written in the tally section. See Sec. 6.10 for more detail.

7.5 [T-Deposit] section

This tally can be used to obtain deposition energies (heat) in certain regions. After version 3.05, it tallies the deposition energies not only by ionization of charged particles but also kerma of neutral particles. The necessity of the use of kerma approximation is automatically judged based on [parameters] section: neutron kerma approximation is not used when e-mode≥1, while photon kerma approximation is not used when negs=1,2. Please see Sec. 5.2.8 in more detail. Note that the neutron kerma approximation is available only for energies below dmax(2) (usually 20 MeV is set).

In the calculation of deposition energies by ionization of charged particle, any factor can be multiplied using the user-defined subroutines "usrdfn1.f" and "usrdfn2.f." As examples, the default program of "usrdfn1.f" returns the dose equivalent calculated from the product of the deposit energy with the $Q(L)$ relationship defined in the ICRP60, while that of "usrdfn2.f" estimates biological dose on the basis of Microdosimetric Kinetic Model [114]. Any factor can be changed and added using this routine. Note that letmat should be set to water when these default programs are used (see "\ParticleTherapy\" in the recommendation setting in more detail). It should be also noted that this function cannot be applied to the deposition energy calculation by kerma approximation and track structure mode.

To tally the energy losses for particles entering the tally region, the counter should be defined by using part in the [counter] section as well as ctmin and ctmax in this tally section.

Table 7.15: [t-deposit] parameters (1)

name	value	explanation
mesh =	reg, r-z, xyz, tet	Mesh type: a mesh type subsection is required below this option.
part =	all (default), particle name	Tally particle. When output=deposit and deposit=0, all is necessary and only maximum 5 particles can be set. Deposition energies calculated by kerma approximation for neutrons and photons are scored in part=neutron and photon, respectively, as well as part=all.
material = (next line)	(optional) all, number of materials 2 5 8	Specify materials for scoring. all : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring. Material numbers.
letmat =	(optional)	Material id for LET(dE/dx). If not defined, a real material is assumed. If a material not used in the geometry is selected, its material density must be defined in [material]. To calculate LET in water, define water with 1 g/cm^3 in [material]. When letmat<0 is set, PHITS automatically calculates dE/dx for water with 1 g/cm^3 for electrons and positrons: please see "ParticleTherapy" in the recommendation setting for more details.
dedxfnc =	(optional, D=0)	0: without; 1: use "usrdfn1.f;" 2: use "usrdfn2.f." As examples, the default program of "usrdfn1.f" returns the dose equivalent calculated from deposit energy multiplied by the $Q(L)$ relationship defined in the ICRP60, while that of "usrdfn2.f" estimates biological dose on the basis of Microdosimetric Kinetic Model.

[114] T.Sato *et al.*, "Biological dose estimation for charged-particle therapy using an improved PHITS code coupled with a microdosimetric kinetic model," Radiat. Res. 171, 107-117 (2009).

Note that deposition energies calculated by kerma approximation are not included in this mode. It should be also noted that this mode cannot be used when weights of particles depositing energy change within a history, such as the cases that [weight window] or [importance] section is defined.

When unit is set =0 with output=dose, results can be obtained in units of [Gy/source]. When mesh=reg, the volumes of each cell should be defined in the [volume] section or set as volume parameters of the [t-deposit] section. Because absorbed dose is an intensive variable, PHITS does not output a 'sum over' in output files for unit=0. Note that, in a region including more than two materials the dose in the region does not equal the average value of the region. For example, when there are two materials with masses M_1 and M_2, and absorption energies E_1 and E_2, respectively, PHITS gives $\frac{E_1}{M_1}\frac{V_1}{V_1+V_2}+\frac{E_2}{M_2}\frac{V_2}{V_1+V_2}$ in this tally, even though its average dose is $\frac{E_1+E_2}{M_1+M_2}$. Here, V_1 and V_2 are volumes of the two materials.

By setting output=deposit, this tally can be also used for calculating the probability density of deposition energies by each history. This mode is useful for estimating the detector responses and the soft-error rates of semi-conductor devices.

When output=deposit and deposit=0, the contribution of each type of particle set in part can be obtained from the result of all. In this case, if all is not set in part, part=all is automatically set.

By setting output=deposit and deposit=1, the distribution of energies deposited by each type of particle set in part can be obtained. Thus, the sum of results obtained by setting part=*individual particle* does not equal to the distribution of energies deposited by all particles.

In the case of output=deposit and deposit=0, the statistical uncertainties of all results except for the values of part=all calculated independent of istdev as standard deviations using history variance mode.

Table 7.16: [t-deposit] parameters (2)

name	value	explanation
e-type =	1, 2, 3, 4, 5	Energy mesh: an energy mesh subsection is required below this option.
t-type =	1, 2, 3, 4, 5 (optional)	Time mesh: a time mesh subsection is required below this option.
output =	dose deposit	Score the energy loss of charged particles and nuclei. Score the variance of deposition energies by individual history (an e-type subsection is required).
unit =	0, 1, 2, 3, 4, 5	0: Dose [Gy/source]; only for output=dose. 1: Dose [MeV/cm^3/source]; only for output=dose. 2: Dose [MeV/source]; only for output=dose. 3: Number [1/source]; only for output=deposit 4: Number [1/nsec/source]; only for output=deposit 5: Dose [J/m^3/source]; only for output=dose
deposit =	0(default), 1	Option for deposit energy plot when output=deposit. 0: distribution of total deposit energy with contribution of each particle (please use this setting normally). 1: distribution of deposit energy by each particle (default setting until ver. 2.81).
mother =	(optional) all, number of mother nuclei	Specify mother nuclei for which kerma factors are calculated. all : default (same as no definition). When setting the number of mother nuclei, define the mothers in the next line. The number of mother can be set as a negative value, in which case the specified mothers are not included for scoring. Note that this parameter is effective only for dose calculations using the kerma approximation for neutrons and photons because energy deposition from charged particles cannot be distinguished to each mother nucleus.
(next line)	208Pb Pb	Nucleus if specified using mass: without mass, all nuclei are isotopes of Pb. To specify multiple mother groups, use multiple [t-deposit] tallies.
axis =	eng, reg, x, y, z, r, t tet xy, yz, xz, rz, t-eng, eng-t	x axis value of output data. x axis value of output data (only active with mesh=tet). 2-dimensional.
samepage =	(optional, D=part)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in axis can be specified.
file =	file name	Define file names. This is required by each setting of axis.
resfile =	(optional, D=file)	Define a file name of the past tally in the restart calculation: even if several axis parameters were defined, specify only one resfile.
factor =	(optional, D=1.0)	Normalization factor.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
2d-type =	1, 2, 3, 4, 5, 6, 7 (optional, D=3)	Options for 2-dimensional plot.
x-txt = y-txt = z-txt =	(optional) (optional) (optional)	*x* axis title. *y* axis title. *z* axis title.

Table 7.17: [t-deposit] parameters (3)

name	value	explanation
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
rshow =	0 (default), 1, 2, 3	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.
resol =	1 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness.
width =	0.5 (default)	The option defines the line thickness.
volume	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined.
reg vol		Volume definition. For details see Sec. 6.1.2.
iechrl =	72 (default)	Maximum number of columns for volume input echo.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz, t-eng, eng-t).
bmpout =	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkout =	0(default), 1	Output the tally results in the xyz-mesh in the input format of ParaView. This file is named by replacing the extension with ".vtk." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkfmt =	0(default), 1	Format of output file for ParaView. 0: ascii, 1: binary.
foamout =	0(default), 1, 2	When foamout=1, generate a file of outputs (numerical data) with OpenFOAM field data format. This file is named by replacing the extension with ".foam." When foamout=2, generate a file of outputs (element no., center of mass *x* coordinate, *y* coordinate, *z* coordinate, volume, output data, relative error) with CSV format. This file is named by replacing the extension with ".csv." When mesh=tet and axis=tet, it is available.

Table 7.18: [t-deposit] parameters (4)

name	value	explanation
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: do not show.
dresol =	(optional, D=0.0)	Parameter for representing detector resolution. This is valid only for output=deposit. When dresol=σ_r and dfano=F, the deposition energy E of each event fluctuates following a Gaussian distribution with standard deviation $\sqrt{\sigma_r^2 + FE}$. If dresol<0, the energy resolutions (energy dependencies, spectral peak shapes, etc) are overwritten by those defined in "usrresol.f."
dfano =	(optional, D=0.0)	Parameter for representing detector resolution. Valid only for output=deposit. When dresol=σ_r and dfano=F, the deposition energy E of each event fluctuated following a Gaussian distribution with standard deviation $\sqrt{\sigma_r^2 + FE}$.
stdcut =	(optional, D=-1)	Threshold value of STD cut off.
nlatcel	(optional)	By setting cell number which has LAT parameter, deposit energies in cells, which have same cell number but placed in different lattice coordinates, are processed as different cells. It works only for output=deposit.
nlatmem	(optional, D=1000)	The maximum number of cells deal with separately in one history.

Although the fano factor is generally defined as a dimensionless quantity, the dfano parameter is defined as a quantity with the dimension of energy. For example, dfano should be set to 3.62E-07 = 3.62E-06 * 0.1, when the fano factor is 0.1 and the average energy to create an electron-hole pair is 3.62 eV for silicon detector.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

As an extension of sum up the deposit energies such as reg = (100 200 300 $\cdots$), weighted summation option is added. Using this option in this tally, energies deposited in specified *i*-th regions for each history, $E_{\mathrm{history},i}$ are multiplied by specified coefficients, $\alpha_i(N_{\mathrm{list}})$, and then summed up as written by:

$$E_{\mathrm{history}} = \sum_i \alpha_i(N_{\mathrm{list}}) \times E_{\mathrm{history},i}. \tag{37}$$

This option weights depending on deposited energies in each history, and it can be applied to simulation, for example soft errors in semiconductor devices. If you want to weight depending on the condition for each particle, please use user defined subroutine ("usrdfn1.f" or "usrdfn2.f") and set dedxfnc=1 or 2. This option can be used only mesh=reg, and it can be executed when you specify reg convolution. Formats and examples are shown below;

```
mesh = reg
  reg = weightsum
ncond = 4
   no   cell   operator   ethres   list
    1    100       ge        0.1      1
    2    100       gt        0.2      1
    3    100       eq        0.0      2
  and    200       lt        0.4      2
    4    100       le        0.5      3
ncell = 5
 cell   list01   list02   list03   list00
  100      0.0      0.5      1.0      1.0
  200      0.1      0.6      2.0      1.0
  300      0.2      0.7      3.0      1.0
  400      0.3      0.8      4.0      1.0
  500      0.4      0.9      5.0      1.0
```

First, you define the number of condition. The next line defines the order of data, i.e., condition number (`no`), cell number (`cell`), operator (`operator`), threshold energy (`ethres`), and list number (`list`). In next lines, conditions are described as many as `ncond` value. These lines means that 'list number is used when the energy deposited in the specified cell satisfies large/small relation with threshold energy.' The condition written in first is preferred when the history satisfies some conditions. If the history does not satisfy all the condition, coefficients of `list00` are used. You can set several conditions by writing `and` in no column. In the cell column, you can use the format `( 2 - 5 8 9 )` and the lattice and universe style as `( 6 < 10[1 0 0] < u=3 )`. But you need to close a value by `( )` if it is not a single numeric value. In the operator column, you can use greater than `gt`, greater equal `ge`, equal `eq`, less equal `le`, and less than `lt`. The unit of `ethres` is MeV.

Next, you define the number of convoluted cell. The next column defines the order of data, i.e., cell number (`cell`), and list number (`list`*xxxx*). In next lines, convoluted cell number and coefficients are described. In the cell column, you can use the format `( 2 - 5 8 9 )` and the lattice and universe style as `( 6 < 10[1 0 0] < u=3 )`. But you need to close a value by `( )` if it is not a single numeric value. It should be noted that same deposited energy is added up many times when you set same cell number many times. If you do not define `list00`, coefficients of `list00` are set to zero.

7.6 [T-Deposit2] section

This tally scores deposit energy distribution in two regions and plot the correlation between two deposit energies. By this, one can simulate, for an example, dE, E counters and plot the correlations in 2-dimensional graph. In the calculation of deposition energies by ionization of charged particle, In the same as `[t-deposit]`, any factor can be multiplied using the user-defined subroutines "usrdfn1.f" and "usrdfn2.f."

Table 7.19: `[t-deposit2]` parameters (1)

name	value	explanation
`mesh =`	`reg`	Mesh type: a `reg` mesh type subsection is required below this option.
`reg =`	`r1 r2`	Specify two cell numbers.
`part =`	`all` (default), particle name	Tally particle.
`letmat1 =`	(optional)	Material id for LET(dE/dx) of region `r1`. If omitted, real material is assumed.
`letmat2 =`	(optional)	Material id for LET(dE/dx) of region `r2`. If omitted, real material is assumed. If a material not used in the geometry is selected, its material density must be defined in `[material]`. To calculate LET in water, define water with 1 g/cm^3 in `[material]`. When `letmat1, or 2`<0 is set, PHITS automatically calculates dE/dx for water with 1 g/cm^3 for electrons and positrons: please see "ParticleTherapy" in the recommendation setting for more detail.
`dedxfnc1 =`	(optional, D=0)	For region `r1`; 0: without; 1: use "usrdfn1.f;" 2: use "usrdfn2.f."
`dedxfnc2 =`	(optional, D=0)	For region `r2`; 0: without; 1: use "usrdfn1.f;" 2: use "usrdfn2.f." As examples, the default program of "usrdfn1.f" returns the dose equivalent calculated from deposit energy multiplied by the $Q(L)$ relationship defined in the ICRP60, while that of "usrdfn2.f" estimates biological dose on the basis of Microdosimetric Kinetic Model.
`e1-type =`	`1, 2, 3, 4, 5`	Energy mesh: for region `r1`, an energy mesh subsection is required below this option.
`e2-type =`	`1, 2, 3, 4, 5`	Energy mesh: for region `r2`, an energy mesh subsection is required below this option.
`t-type =`	`1, 2, 3, 4, 5` (optional)	Time mesh: a time mesh subsection is required below this option.
`unit =`	`1, 2`	1: Number [1/source] 2: Number [1/nsec/source]

`axis =`	`eng1, eng2, t`
	`e12, e21, t-e1, t-e2,`
	`e1-t, e2-t`
		x axis value of output data.
		2-dimensional.

`samepage =`	(optional, `D=part`)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in `axis` can be specified.

Table 7.20: [t-deposit2] parameters (2)

name	value	explanation
file =	file name	Define file names. This is required by each setting of axis.
resfile =	(optional, D=file)	Define a file name of the past tally in the restart calculation: even if several axis parameters were defined, specify only one resfile.
factor =	(optional, D=1.0)	Normalization factor.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
2d-type =	1, 2, 3, 4, 5, 6, 7 (optional, D=3)	Options for 2-dimensional plot.
x-txt =	(optional)	*x* axis title.
y-txt =	(optional)	*y* axis title.
z-txt =	(optional)	*z* axis title.
volume	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined.
reg vol		Volume definition. For details see Sec. 6.1.2.
iechrl =	72 (default)	Maximum number of columns for volume input echo.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=e12, e21, t-e1, t-e2, e1-t, e2-t).
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.

Table 7.21: `[t-deposit2]` parameters (3)

name	value	explanation
`ctmin(i) =`	(optional, D=-9999)	Minimum value for *i*-th counter.
`ctmax(i) =`	(optional, D= 9999)	Maximum value for *i*-th counter.
`chmin(i) =`	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (`istdev=1`).
`chmax(i) =`	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (`istdev=1`).
`dresol1 =`	`(optional, D=0.0)`	Parameter for representing detector resolution in tally region 1. When `dresol1`=σ_r and `dfano1`=F, the deposition energy E of each event fluctuates following a Gaussian distribution with standard deviation $\sqrt{\sigma_r^2 + FE}$. If `dresol1<0`, the energy resolutions (energy dependencies, spectral peak shapes, etc) are overwritten by those defined in "usrresol.f."
`dresol2 =`	`(optional, D=0.0)`	Parameter for representing detector resolution in tally region 2. See the column of `dresol1` in more detail.
`dfano1 =`	`(optional, D=0.0)`	Parameter for representing detector resolution in tally region 1. See the column of `dresol1` in more detail.
`dfano2 =`	`(optional, D=0.0)`	Parameter for representing detector resolution in tally region 2. See the column of `dresol1` in more detail.
`stdcut =`	(optional, `D=-1`)	Threshold value of STD cut off.

To tally the energy loss for each projectile particle entering the tally region, the counter should be defined using `part` in the `[counter]` section and `ctmin`, `ctmax` in this tally section.

When specifying `stdcut`, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when `stdcut` is positive and `itall=0,1` is set in `[parameters]` section. When "all" relative values of STD of the tally result are larger than 0 and smaller than `stdcut` at the last of one batch. If `stdcut` in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

7.7 [T-Heat] section

This tally can be used to obtain deposition energies (heat) in certain regions using kerma approximation. However, the function to calculate deposition energy using kerma approximation was also implemented in [t-deposit] after version 3.05, and thus, we do not recommend to use [t-heat] after that version.

Although the deposition energy from neutrons is zero for e-mode≥1, it can be calculated instead from the energy loss of all charged particles and nuclei. The deposition energy from photons is also usually obtained from kerma factors. For electron=1 with electron transport, photon kerma factors are not used; instead, deposition energies are obtained from the energy loss of electrons.

Table 7.22: [t-heat] parameters (1)

name	value	explanation
mesh =	reg, r-z, xyz	Mesh type: a mesh type subsection is required below this option.
axis =	reg, x, y, z, r xy, yz, xz, rz,	x axis value of output data. 2-dimensional.
samepage =	(optional, D=part)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in axis can be specified.
file =	file name	Define file names. This is required by each setting of axis.
resfile =	(optional, D=file)	Define a file name of the past tally in the restart calculation: even if several axis parameters were defined, specify only one resfile.
material =	(optional) all, number of materials	Specify materials for scoring. all : default (same as no definition). When setting more than one material number, the material numbers should be specified in the next line. By setting a material number to a negative value the specified material is excluded from scoring.
(next line)	2 5 8	Material numbers.
output =	heat simple	total: Total deposit energy. total: Total deposit energy. leakage: Kinetic energy of particles entering the outer void. total: Total deposit energy. recoil: Kinetic energy of residual nuclei when cut-off energy emin(15-19) is set. ionization: Energy deposited by energy-loss of charged particles. low neutron: Energy deposit calculated by neutron kerma factors. photon: Energy deposit by kerma factors: (If electron=1, contributions of electrons below cut-off energy.) others: Excitation energy of residual nuclei. Setting igamma=1 makes this value 0 owing to photon emission.

Table 7.23: [t-heat] parameters (2)

name	value	explanation
output = (continued)	all	In addition to the above parameters; Contributions of $d, t, ^3$He, α, and residual nuclei to recoil. Contributions of p, π^+, π^-, and others to ionization (contributions of particles specified by part are output. but not plotted in the eps file). stopped particle: Kinetic energy of stopped proton, neutron, π^+, π^-, and other particles in materials (contributions of particles specified by part are output but not plotted in the eps file). others: Remaining excitation energy and fission components. When axis is 2-dimensional, all is the same as simple. Only total, recoil, ionization, low neutron, electron, and others are output.
part =	particle name(optional)	When output set =all, deposit energies for ionization and stopped particles specified here are output. Results are not plotted in the eps file. Photon and neutron contributions calculated with kerma factor cannot be distinguished by specifying part. For this purpose, please see the output obtained by output=simple.
unit =	0, 1, 2	0: [Gy/source] 1: [MeV/cm^3/source] 2: [MeV/source]
factor =	(optional, D=1.0)	Normalization factor.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
2d-type =	1, 2, 3, 4, 5, 6, 7 (optional, D=3)	Options for 2-dimensional plot.
x-txt = y-txt = z-txt =	(optional) (optional) (optional)	x axis title. y axis title. z axis title.
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
rshow =	0 (default), 1, 2, 3	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.
resol = width =	1 (default) 0.5 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness. The option defines the line thickness.

Generally speaking, heat is energy produced by the ionization of charged particles. However, in the transport simulation, transport is stopped below the set particle cut-off energy. In this case, additional components of heat, including recoil, stopped particle, and others, will be output to the heat tally. These components may change as the parameters of the transport are changed.

Table 7.24: [t-heat] parameters (3)

name	value	explanation
volume	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined.
reg vol		Volume definition. For details see Sec. 6.1.2.
iechrl =	72 (default)	Maximum number of columns for volume input echo.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz, t-eng, eng-t).
bmpout =	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkout =	0(default), 1	Output the tally results in the xyz-mesh in the input format of ParaView. This file is named by replacing the extension with ".vtk." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkfmt =	0(default), 1	Format of output file for ParaView. 0: ascii, 1: binary.
electron =	0 (default), 1	Eelectron contribution options. 0: using photon kerma factors (electron and positron should NOT be transported, otherwise their deposition energies are double-counted) 1: calculating by ionization loss (electron and positron transport are required)
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: do not show.

When unit is set =0 with output=dose, results can be obtained in units of [Gy/source]. When mesh=reg, the volumes of each cell should be defined in the [volume] section or set as volume parameters of the [t-deposit] section. Because absorbed dose is an intensive variable, PHITS does not output a 'sum over' in output files for unit=0. Note that, in a region including more than two materials the dose in the region does not equal the average value of the region. For example, when there are two materials with masses of M_1 and M_2, and absorption energies of E_1 and E_2, respectively, PHITS gives $\frac{E_1}{M_1}\frac{V_1}{V_1+V_2} + \frac{E_2}{M_2}\frac{V_1}{V_1+V_2}$ in this tally, even though its average dose is $\frac{E_1+E_2}{M_1+M_2}$. Here, V_1 and V_2 are volumes of the two materials.

Table 7.25: `[t-heat]` parameters (4)

name	value	explanation
`stdcut =`	(optional, `D=-1`)	Threshold value of STD cut off.

When specifying `stdcut`, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when `stdcut` is positive and `itall=0,1` is set in `[parameters]` section. When "all" relative values of STD of the tally result are larger than 0 and smaller than `stdcut` at the last of one batch. If `stdcut` in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

7.8 [T-Yield] section

This tally gives information on produced nuclei. Nuclei with both the number of protons and neutrons greater than zero are tallied. No residual nuclei calculated by nuclear data libraries are scored. Note that the nuclei produced by neutrons in the energy below `dmax(2)` are scored by setting `e-mode`≥1.

Table 7.26: `[t-yield]`parameters (1)

name	value	explanation
`mesh =`	`reg, r-z, xyz, tet`	Mesh type: a mesh type subsection is required below this option.
`volume`	(optional)	This option defines the volume for each region for `reg` mesh. Volume definitions are required below this option. Values defined in `[volume]` are used in the case of no definition. If special description such as using `( )` is used to specify a region in `reg=`, its internally defined region number is output in input echo when this volume subsection is not defined.
`reg vol`		Volume definition. For details see Sec. 6.1.2.
`iechrl =`	`72` (default)	Maximum number of columns for volume input echo.
`part =`	`all` (default), particle name	Reaction particle.
`axis =`	`reg, x, y, z, r`	x axis value of output data.
	`tet`	x axis value of output data (only active with `mesh=tet`).
	`xy, yz, xz, rz`	2-dimensional.
	`mass`	Mass number distribution: if the nucleus is specified, isotope distribution is used.
	`charge`	Atomic number distribution: nucleus cannot be specified.
	`chart`	Nucleus chart (x:N, y:Z). Nucleus cannot be specified.
	`dchain`	for DCHAIN output. This option is not available for `mesh=r-z` only.
		For this tally, only one axis parameter is defined in the `[t-yield]` section.
`info =`	`0, 1`	With stable nuclei and magic number for chart.
`file =`	`file name`	Define file names. This is required by each setting of `axis`.
`resfile =`	(optional, `D=file`)	Define a file name of the past tally in the restart calculation: even if several `axis` parameters were defined, specify only one `resfile`.

In the `[t-yield]` section, projectiles to tally can be specified by `part`, but the output is the sum of their contributions. To obtain each contribution separately, set multiple `[t-yield]` sections.

When `igamma=3` in the `[parameters]` section, information on the isomer production based on the EBITEM model can be obtained by setting `axis=chart` or `=dchain` in this tally.

Table 7.27: [t-yield]parameters (2)

name	value	explanation
unit =	1, 2	1: [1/source] 2: [1/cm^3/source]
factor =	(optional, D=1.0)	Normalization factor.
output =	(optional) product (default) cutoff	Change the timing of the score. Nuclei produced by nuclear reaction are tallied. Nuclei stopped by energy cutoff are tallied. If nuclei are not transported, this is the same as product.
mother =	(optional) all, number of mother nuclei (next line) 208Pb Pb	Specify mother nuclei. all : default (same as no definition). When setting the number of mother nuclei, define the mothers in the next line. The number of mother can be set as a negative value, in which case the specified mothers are not included for scoring. Nucleus if specified using mass: without mass, all nuclei are isotopes of Pb. To specify multiple mother groups, use multiple [t-yield] tallies.
nucleus =	(optional) all, number of nuclei (next line) 208Pb Pb	Specify output nuclei. all : default (same as no definition). When setting number of nuclei, define the nuclei in the next line. Nucleus if specified with mass. Without mass, all nuclei are isotopes of Pb.
elastic =	1(default),0,-1	Option for treating secondary particles produced by nuclear reaction. In the case of output=product 1: tally all secondary particles including those generated from elastic scattering. 0: tally secondary particles generated only from inelastic scattering. -1: Do not tally any secondary particles if one of the secondary particles is the same as the target nucleus, such as elastic scattering case. If not, tally all secondary particles but subtract the target nucleus. In the case of output=cutoff 0,1: Do not tally any secondary particles. -1: Subtract the target neucleus. When elastic=-1, the yields of target nuclei could be a negative value, but those data are ignored in the eps file (regarded as 0 in the figure).
mxnuclei =	3000(default)	Maximum number of nuclides scored in the case of axis=chart or dchain. 0: Score the yield of all nuclides on the nuclear chart (136 charge numbers * 236 mass numbers * 3 meta stables = 96,288). >0: Score the yield of nuclides up to this number, and ignore the rest by outputting a warning message. A smaller mxnuclei results in smaller size of allocated memory and shorter computational time, but larger chance to miss the yields of some important nuclides.

If output=cutoff is specified, the parameters of part and mother are neglected.

Table 7.28: [t-yield]parameters (3)

name	value	explanation
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
2d-type =	1, 2, 3, 4, 5, 6, 7 (optional, D=3)	Options for 2-dimensional plot.
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
rshow =	0 (default), 1, 2, 3	When mesh=reg,tet and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.
gslat =	2(default), = 0 = 1 = 2 = 3	Option to draw lattice or tetrahedral geometry boundary lines when gshow or rshow is specified. Not drawing. Drawing. Not that the line of voxel phantoms may not be drawn clearly. Not drawing the lines in the same cell. Not drawing the lines in the same material. In this case, the boundary of two adjacent cells filled with the same material is not also drawn, even if not lattice or tetrahedral geometry. When gshow≥ 3, cell and lattice numbers are now shown.
x-txt = y-txt = z-txt =	(optional) (optional) (optional)	*x* axis title. *y* axis title. *z* axis title.
resol = width =	1 (default) 0.5 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness. The option defines the line thickness.
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.

Table 7.29: [t-yield]parameters (4)

name	value	explanation
material =	(optional) all, number of materials	Specify materials for scoring. all : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring.
(next line)	2 5 8	Material numbers.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
bmpout =	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkout =	0(default), 1	Output the tally results in the xyz-mesh in the input format of ParaView. This file is named by replacing the extension with ".vtk." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkfmt =	0(default), 1	Format of output file for ParaView. 0: ascii, 1: binary.
foamout =	0(default), 1, 2	When foamout=1, generate a file of outputs (numerical data) with OpenFOAM field data format. This file is named by replacing the extension with ".foam." When foamout=2, generate a file of outputs (element no., center of mass *x* coordinate, *y* coordinate, *z* coordinate, volume, output data, relative error) with CSV format. This file is named by replacing the extension with ".csv." When mesh=tet and axis=tet, it is available.
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.

Table 7.30: [t-yield]parameters (5)

name	value	explanation
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
stdcut =	(optional, D=-1)	Threshold value of STD cut off.
special =	D=0 (optional)	When special> 0, nuclear reactions are repeated more than once to increase statistics.
ndata =	0, 1, 2(default), 3	Option to replace the results of nuclear reaction events with particle production cross section (activation cross section) data. 0: This option is not available. 1: Replace with cross section data in nucleon-induced reactions on α, ^{14}N, ^{16}O targets. 2: Replace with cross section data in a directory specified by file(27) (D=file(1)/XS/yield/) in proton, neutron, deuteron, alpha-induced, and photo-nuclear reactions when using the evaluated nuclear data library without e-mode. 2: Replace with cross section data in a directory specified by file(27) (D=file(1)/XS/yield/) in proton, neutron, deuteron, alpha-induced, and photo-nuclear reactions when using the evaluated nuclear data library with or without e-mode or nuclear reaction models.

When coll or reaction events belonging to coll are specified in [counter], the counter value will change before the particle information has been tallied. Therefore, to tally the information at the moment the reaction event occurs, specify the counter value after the change.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

When specifying ndata option, PHITS replaces the results of nuclear reaction events with special reaction cross section data (ndata=1) or cross section data in a directory specified by file(27)[115](ndata=2,3). This option helps you obtain more reliable production yields when the yield given by the nuclear reaction models does not reproduce the experimental data well. When a nuclear reaction event occurs in a transport calculation, PHITS replaces a tally yield obtained from the evaluated data or model with the cross section data of the ndata option. This replacement occurs only in the tally that specifies ndata and is independent of the other tallies or the transport calculation. Note that double counting may occur if output=cutoff. For example, if a secondary particle produced by a nuclear reaction model has an energy greater than emin, it will be counted twice, with the contribution of the replacement at the time of the reaction event and the contribution at the time of the energy cutoff after its transport. If you want to avoid the double-counting of tally yields, please set ctmax(i)=0, (i=1,2,3) and ndata in [counter] section. See also Sec. 5.27 for this method.

When ndata=1, special reaction cross section data are used to particle yields of the tally in reaction events of neutron- or proton-induced reactions on ^{4}He,^{14}N,^{16}O targets. The following nuclear reactions are considered. The tally value is evaluated by excitation functions of these nuclear reactions.

[115] From ver. 3.35, activation cross section data of JENDL-5 are stored. See the following paper for more details: O. Iwamoto, N. Iwamoto, S. Kunieda, F. Minato, S. Nakayama, Y. Abe, et al., "Japanese evaluated nuclear data library version 5: JENDL-5", J. Nucl. Sci. Technol., 60(1), 1-60 (2023). DOI: 10.1080/00223131.2022.2141903 (Open Access).

^{4}He$(n,x)^{3}$H	^{14}N$(n,x)^{3}$H	^{14}N$(n,x)^{7}$Be	^{14}N$(n,x)^{11}$Be	^{14}N$(n,x)^{10}$C	^{14}N$(n,x)^{11}$C
^{14}N$(n,x)^{14}$C	^{14}N$(n,x)^{13}$N	^{16}O$(n,x)^{3}$H	^{16}O$(n,x)^{7}$Be	^{16}O$(n,x)^{11}$Be	^{16}O$(n,x)^{10}$C
^{16}O$(n,x)^{11}$C	^{16}O$(n,x)^{14}$C	^{16}O$(n,x)^{15}$C	^{16}O$(n,x)^{13}$N	^{16}O$(n,x)^{16}$N	^{16}O$(n,x)^{14}$O
^{16}O$(n,x)^{15}$O	^{4}He$(p,x)^{3}$H	^{14}N$(p,x)^{7}$Be	^{14}N$(p,x)^{11}$Be	^{14}N$(p,x)^{10}$C	^{14}N$(p,x)^{11}$C
^{14}N$(p,x)^{13}$N	^{14}N$(p,x)^{14}$O	^{16}O$(p,x)^{3}$H	^{16}O$(p,x)^{7}$Be	^{16}O$(p,x)^{11}$Be	^{16}O$(p,x)^{10}$C
^{16}O$(p,x)^{11}$C	^{16}O$(p,x)^{14}$C	^{16}O$(p,x)^{13}$N	^{16}O$(p,x)^{14}$O	^{16}O$(p,x)^{15}$O	

When `ndata=2,3`, this option uses cross section data in a directory specified by `file(27)`. From ver. 3.35, activation cross section data of JENDL-5 are stored. You can obtain particle production yields based on JENDL-5 for proton, neutron, deuteron, alpha-induced, and photo-nuclear reactions with energies up to 200, 200, 100, 50, 200 MeV/n, respectively. If `ndata=2`, replacement is only performed when using the evaluated nuclear data library without `e-mode`. If `ndata=3`, there is no such limitation as above, and PHITS basically replaces the calculated results of all nuclear reaction events. No replacement are made for reactions other than the target nuclides and incident energies given in the `file(27)`, and the results of the evaluated nuclear data and nuclear reaction models are tallied as is.

For `ndata=2,3` the production of all particles is evaluated by replacing the cross section data for `output=product`, but for `output=cutoff` the production of light ions such as d,t,^{3}He,α are not replaced. This is because these particles are likely to be produced with energies greater than `emin` in the calculation of the evaluated nuclear data and nuclear reaction models, and will be tallied at the energy cutoff after transport. However, even with `output=cutoff`, if the calculation is performed by evaluated nuclear data without `e-mode`, the production of light ions will not occur (with `lionprd=-1` (default value)), thus replacing all particle production. When `ndata=2,3` and `axis=dchain`, no cross section data replacement is performed for neutron-induced reactions below 20 MeV, because this `axis` assumes that the production yield from the reactions is calculated using DCHAIN.

For `ndata=2,3`, you can use original data by changing the data file in `file(27)` (`D=file(1)/dchain-sp/data/`). Its file name should be given as "[element symbol]+[mass number(three digits)]+[-y-]+[incident particle ID]+[.dat]," e.g., "O_016-y-n.dat" or "Pd107-y-n.dat." Here, the incident particle IDs of proton, neutron, deuteron, α, and photon (photon-nucleus) are p, n, d, a, and g, respectively. Note that if the element has only one character in its symbol, an underline "_" must be added after the element symbol. An example of the production cross section data is as follows.

```
# ZAP = 48113  LIP = 1  INT = 2
# Elab (MeV)     sigma (b)
1.000000e-11  0.000000e+00
2.530000e-08  0.000000e+00
2.000000e+01  0.000000e+00
2.100000e+01  0.000000e+00
 .......
```

where `ZAP`, `LIP`, and `INT` are produced nuclei, its isomer state, and interpolation method, respectively. The produced nuclei are specified as $1000Z + A$, where Z is atomic number and A(three digits) is mass number. The isomer state is specified by `LIP`: =0; the ground state, =1; the first metastable state, =2; the second metastable state. The interpolation method `INT = (1,2,3,4,5)` are as follows.

```
INT = 1  : Histogram
      2  : Linear(x) - Linear(y)
      3  : Log(x) - Linear(y)
      4  : Linear(x) - Log(y)
      5  : Log(x) - Log(y)
```

The second line should be written as,

```
# Elab (MeV)    sigma (b)
```

The incident energy [MeV] and cross section [barn] should be written in the first and second columns from the third line. The number of the energy bin is not limited. Note that when the incident energy falls outside the energy range specified by the data, the cross section data at the minimum or maximum energies are used, respectively for the lower and higher energies. Larger nuclei than He are tallied from the data.

7.9 [T-Product] section

This tally gives particles and nuclei produced by nuclear reactions, decays, and fission, and also tallies source particles. It differs from [t-yield] in that the energy distribution and time distribution of produced particles and nuclei can be obtained in [t-product]. No residual nuclei calculated by nuclear data libraries are scored. Note that the nuclei produced by neutrons in the energy below dmax(2) are scored by setting e-mode≥1.

Table 7.31: [t-product] parameters (1)

name	value	explanation
mesh =	reg, r-z, xyz, tet	Mesh type: a mesh type subsection is required below this option.
part =	all (default), particle name	Produced particle.
material =	(optional) all, number of materials	Specify materials for scoring. all : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring.
(next line)	2 5 8	Material numbers.
mother =	(optional) all, number of mother nuclei	Specify mother nuclei. all : default (same as no definition). When setting the number of mother nuclei, define the mothers in the next line. The number of mother can be set as a negative value, in which case the specified mothers are not included for scoring. When output=atomic, mother cannot be specified.
(next line)	208Pb Pb	Nucleus if specified using mass: without mass, all nuclei are isotopes of Pb. To specify multiple mother groups, use multiple [t-product] tallies.
e-type =	1, 2, 3, 4, 5	Energy mesh: an energy mesh subsection is required below this option.
eng2let =	(optional)	0 (default): Do not convert energy defined by e-type to LET. 1: Convert energy to LET (keV/μm). Set eng2let=2 and axis=let when you want to output the results as a function of LET.
t-type =	1, 2, 3, 4, 5 (optional)	Time mesh: a time mesh subsection is required below this option.
a-type =	1, 2, -1, -2	Angle mesh with θ degrees of freedom for mesh=r-z: an angle mesh subsection is required below this option.

Isomeric nuclei are supported as valid inputs for the part parameter. Isomers can be specified by placing m, n, or g at the end of isotope names such as Na-24m (first isomer), In-116m (first isomer), In-116n(second isomer) or In-116g(ground state). To score isomers, do not forget to set |igamma|=3 in [Parameters] section. Even if |igamma|=3, part=Na-24 means to score summation of Na-24 ground state and Na-24m isomer.

Table 7.32: [t-product] parameters (2)

name	value	explanation
unit =	1, 2, 3, 4, 5, 6	1: [1/source] 2: [1/cm^3/source] 3: [1/MeV/source] 4: [1/cm^3/MeV/source] 5: [1/Lethargy/source] 6: [1/cm^3/Lethargy/source]
	11, 12, 13, 14, 15, 16	11:[1/nsec/source] 12:[1/cm^3/nsec/source] 13:[1/MeV/nsec/source] 14:[1/cm^3/MeV/nsec/source] 15:[1/Lethargy/nsec/source] 16:[1/cm^3/Lethargy/nsec/source]
	21, 22, 23, 24, 25, 26	21: [1/sr/source] 22: [1/cm^3/sr/source] 23: [1/MeV/sr/source] 24: [1/cm^3/MeV/sr/source] 25: [1/Lethargy/sr/source] 26: [1/cm^3/Lethargy/sr/source]
	31, 32, 33, 34, 35, 36	31:[1/nsec/sr/source] 32:[1/cm^3/nsec/sr/source] 33:[1/MeV/nsec/sr/source] 34:[1/cm^3/MeV/nsec/sr/source] 35:[1/Lethargy/nsec/sr/source] 36:[1/cm^3/Lethargy/nsec/sr/source]
axis =	eng, let, reg, x, y, z, r tet xy, yz, xz, rz t	x axis value of output data. x axis value of output data (only active with mesh=tet). 2-dimensional. Time axis.
samepage =	(optional, D=part)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in axis can be specified.
file =	file name	Define file names. This is required by each setting of axis.
resfile =	(optional, D=file)	Define a file name of the past tally in the restart calculation: even if several axis parameters were defined, specify only one resfile.

'Lethargy' in unit=5, 6, 15, 16, 25, 26, 35 or 36 is the natural logarithmic unit of energy defined by $\ln(E_{\mathrm{ref}}/E)$ using a reference energy E_{ref} and the particle energy E. Setting these units enables obtaining results in units of Lethargy, which are given as Lethargy widths, $\ln(E_{\mathrm{high}}/E_{\mathrm{low}})$, for each energy bin in the energy mesh subsection. Here E_{high} and E_{low} are the maximum and minimum values of the energy bins, respectively.

In unit=21 - 26 or 31 - 36, 'sr' denotes steradians as the solid angle unit.

Table 7.33: [t-product] parameters (3)

name	value	explanation
output =	source	Source particles defined in [source] section.
	nuclear (default)	Particles from nuclear reaction, including elastic.
	nonela	Particles from nonelastic collision.
	elastic	Particles from elastic collision.
	decay	Particles from decay. （Particles generated from RI source are not included.)
	fission	Particles from fission. (Particles generated from fission neutron source are not included.)
	atomic	Particles from atomic interaction.
primary =	1(optional), 0	1: Both primary and secondary particles at the end of reactions are tallied. 0: Only secondary particles are tallied. This parameter is valid for output=nuclear, elastic, atomic. Otherwise, it is treated as 0.
factor =	(optional, D=1.0)	Normalization factor.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
2d-type =	1, 2, 3, 4, 5, 6, 7 (optional, D=3)	Options for 2-dimensional plot.
x-txt =	(optional)	*x* axis title.
y-txt =	(optional)	*y* axis title.
z-txt =	(optional)	*z* axis title.
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
rshow =	0 (default), 1, 2, 3	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.
resol =	1 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness.
width =	0.5 (default)	The option defines the line thickness.

In case output= decay or source, primary parameter is invalid because primary particle does not exist. No particles are excluded from tally by primary. In case output= nonela, or fission, primary particle can survive but primary parameter is invalid because primary and secondary particles are indistinguishable. No particles are excluded from tally by primary. In case output= atomic, track structure mode is used, and primary=1, primary particle is scored at every atomic reaction. Please ensure that this operation aligns with your intent. output= nuclear is sum of nonela and elastic. primary only affects elastic whereas nonela is not affected.

Table 7.34: [t-product] parameters (4)

name	value	explanation
volume reg vol	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined. Volume definition. For details see Sec. 6.1.2.
iechrl =	72 (default)	Maximum number of columns for volume input echo.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
bmpout =	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkout =	0(default), 1	Output the tally results in the xyz-mesh in the input format of ParaView. This file is named by replacing the extension with ".vtk." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkfmt =	0(default), 1	Format of output file for ParaView. 0: ascii, 1: binary.
foamout =	0(default), 1, 2	When foamout=1, generate a file of outputs (numerical data) with OpenFOAM field data format. This file is named by replacing the extension with ".foam." When foamout=2, generate a file of outputs (element no., center of mass *x* coordinate, *y* coordinate, *z* coordinate, volume, output data, relative error) with CSV format. This file is named by replacing the extension with ".csv." When mesh=tet and axis=tet, it is available.
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.

Table 7.35: [t-product] parameters (5)

name	value	explanation
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
dump =	Number of data	For mesh=reg, the information is dumped on the file. If dump is negative, data are written by ascii, if positive, by binary.
(next line)	Data sequence	Define the data sequence. The history information (nocas and nobch) is necessary to use idmpmode=1.
letmat =	(optional)	(Effective only when eng2let=1) Material id for LET(dE/dx). If not defined, a real material is assumed. If a material not used in the geometry is selected, its material density must be defined in [material]. To calculate LET in water, define water with 1 g/cm^3 in [material]. When letmat<0 is set, PHITS automatically calculates dE/dx for water with 1 g/cm^3 for electrons and positrons: please see "ParticleTherapy" in the recommendation setting for more details.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: do not show.
stdcut =	(optional, D=-1)	Threshold value of STD cut off.

When coll or reaction events belonging to coll are specified in [counter], the counter value will change before the particle information has been tallied. Therefore, to tally the information at the moment the reaction event occurs, specify the counter value after the change.

In the [t-product] tally, the dump option can only be used. If the dump option is set, the meshes of e-type and t-type have only the meaning of the maximum and minimum values. In using this dump parameter, axis and file are restricted to one axis and one file apiece and unit is always 1. The dumped data are written onto a file named "***_dmp", where "***" indicates the file name specified by file=***. The normal output of the tally is written on "***." From this file, information on the total normalization factor can be obtained; doing so requires setting one mesh each for e-type, a-type and t-type. Note that constraints based on the history counter (chmin & chmax) do not work with the dump function because the particle information is output to the dump file immediately after particle generation. The history information (nocas and nobch) is necessary to use idmpmode=1 for continuous calculation using the dump file; in addition, both the dump file with "_dmp" and the normal output file specified by file= are required to use idmpmode=1. The option dumpall is not compatible with this dump tally option when shared memory parallelization is active.

[t-product] can tally source particles, and the function can be used to modify the dump file. Dump files can be read and information written to a new dump file through modifications made by setting the dump parameter and output=source in this tally section and icntl=6 in the [parameters] section.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

7.10 [T-DPA] section

This tally gives the displacement per atom (DPA) value, or number of displaced atoms per target atom, which represents radiation damage in materials irradiated by energetic particles. The result produced by this tally includes the contribution of the Coulomb scattering cross section for charged particle transportation. The DPA of low energy neutron incident reactions can be also obtained using `e-mode`≥1. Note that the sum-over value is not output from `[t-dpa]`.

Table 7.36: `[t-dpa]` parameters (1)

name	value	explanation
`mesh =`	`reg, r-z, xyz, tet`	Mesh type: a mesh type subsection is required below this option.
`part =`	`all` (default), particle name	Tally particle.
`material =`	(optional) `all`, number of materials (next line)	Specify materials for scoring. `all` : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring. Material numbers.
`unit =`	`1, 2`	1: DPA [x1.E-24/source] 2: DPA [1/source]
`idpa =`	(optional, D=0)	0: dpa without athermal recombination correction (NRT-dpa). 1: dpa with athermal recombination correction (arc-dpa).

`axis =`	`eng, reg, x, y, z, r`
	`tet`
	`xy, yz, xz, rz`
		x axis value of output data.
		x axis value of output data (only active with `mesh=tet`).
		2-dimensional.

`samepage =`	(optional, `D=part`)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in `axis` can be specified.
`file =`	`file name`	Define file names. This is required by each setting of `axis`.
`resfile =`	(optional, `D=file`)	Define a file name of the past tally in the restart calculation: even if several `axis` parameters were defined, specify only one `resfile`.

Table 7.37: [t-dpa] parameters (2)

name	value	explanation
output =	dpa	Total DPA value.
mother =	(optional) all, number of mother nuclei	Specify mother nuclei for contributing to DPA. all : default (same as no definition). When setting the number of mother nuclei, define the mothers in the next line. The number of mother can be set as a negative value, in which case the specified mothers are not included for scoring. Note that this parameter is effective only for dose calculations using the kerma approximation for neutrons and photons because energy deposition from charged particles cannot be distinguished to each mother nucleus.
(next line)	208Pb Pb	Nucleus if specified using mass: without mass, all nuclei are isotopes of Pb. To specify multiple mother groups, use multiple [t-dpa] tallies.
factor =	(optional, D=1.0)	Normalization factor.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
2d-type =	1, 2, 3, 4, 5, 6, 7 (optional, D=3)	Options for 2-dimensional plot.
x-txt = y-txt = z-txt =	(optional) (optional) (optional)	*x* axis title. *y* axis title. *z* axis title.
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
rshow =	0 (default), 1, 2, 3	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.
resol =	1 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness.
width =	0.5 (default)	The option defines the line thickness.
volume	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined.
reg vol		Volume definition. For details see Sec. 6.1.2.
iechrl =	72 (default)	Maximum number of columns for volume input echo.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.

Table 7.38: [t-dpa] parameter (3)

name	value	explanation
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
bmpout =	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkout =	0(default), 1	Output the tally results in the xyz-mesh in the input format of ParaView. This file is named by replacing the extension with ".vtk." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkfmt =	0(default), 1	Format of output file for ParaView. 0: ascii, 1: binary.
foamout =	0(default), 1, 2	When foamout=1, generate a file of outputs (numerical data) with OpenFOAM field data format. This file is named by replacing the extension with ".foam." When foamout=2, generate a file of outputs (element no., center of mass *x* coordinate, *y* coordinate, *z* coordinate, volume, output data, relative error) with CSV format. This file is named by replacing the extension with ".csv." When mesh=tet and axis=tet, it is available.
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.
ctmin(i) = ctmax(i) =	(optional, D=-9999) (optional, D= 9999)	Minimum value for *i*-th counter. Maximum value for *i*-th counter.
chmin(i) = chmax(i) =	(optional, D=-9999) (optional, D= 9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1). Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: do not show.
stdcut =	(optional, D=-1)	Threshold value of STD cut off.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

7.11 [T-LET] section

This tally can be used to obtain information on track length and dose as a function of the LET(dE/dx) of a given material. This tally counts the energy losses of charged particles and nuclei; accordingly, the Event Generator mode (`e-mode`≥1) must be used to transport low-energy neutrons. Note that this tally does not consider the contributions of electrons or positrons below their cutoff energies ((`emin`)) in the result.

Table 7.39: `[t-let]` parameters (1)

name	value	explanation
`mesh =`	`reg, r-z, xyz`	Mesh type: a mesh type subsection is required below this option.
`part =`	`all` (default), particle name	Tally particle.
`material =`	(optional) `all`, number of materials	Specify materials for scoring. `all` : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring.
(next line)	`2 5 8`	Material numbers.
`letmat =`	(optional)	Material id for LET(dE/dx). If not defined, a real material is assumed. If a material not used in the geometry is selected, its material density must be defined in `[material]`. To calculate LET in water, define water with 1 g/cm^3 in `[material]`. When `letmat`<0 is set, PHITS automatically calculates dE/dx for water with 1 g/cm^3 for electrons and positrons: please see "ParticleTherapy" in the recommendation setting for more details.
`l-type =`	`1, 2, 3, 4, 5`	LET mesh: a LET mesh subsection must be added below this option. Note that the LET spectrum may have unnatural peaks when a very fine mesh, e.g., 20 meshes per order of magnitude, is set.

Table 7.40: [t-let] parameters (2)

name	value	explanation
unit =	1, 2, 3, 4, 5, 6	1: Track [cm/(keV/μm)/source] 2: Dose [MeV/(keV/μm)/source] 3: Track [cm/ln(keV/μm)/source] 4: Dose [MeV/ln(keV/μm)/source] 5: Track [cm/source] 6: Dose [MeV/source]
	7, 8, 9, 10, 11, 12	7: Track [1/cm^2/(keV/μm)/source] 8: Dose [MeV/cm^3/(keV/μm)/source] 9: Track [1/cm^2/ln(keV/μm)/source] 10:Dose [MeV/cm^3/ln(keV/μm)/source] 11:Track [1/cm^2/source] 12:Dose [MeV/cm^3/source]
	13, 14	13: $L * f(L)$ [dimensionless], where $\int f(L)dL = 1$. 14: $L * d(L)$ [keV/μm], where $\int d(L)dL = 1$.
axis =	let, reg, x, y, z, r xy, yz, xz, rz	x axis value of output data. 2-dimensional.
samepage =	(optional, D=part)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in axis can be specified.
file =	file name	Define file names. This is required by each setting of axis.
resfile =	(optional, D=file)	Define a file name of the past tally in the restart calculation: even if several axis parameters were defined, specify only one resfile.
factor =	(optional, D=1.0)	Normalization factor.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
2d-type =	1, 2, 3, 4, 5, 6, 7 (optional, D=3)	Options for 2-dimensional plot.
x-txt = y-txt = z-txt =	(optional) (optional) (optional)	*x* axis title. *y* axis title. *z* axis title.
gshow =	0 (default), 1, 2, 3, 4	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option.
rshow =	0 (default), 1, 2, 3	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.

From version 3.02, new options (unit=13 and 14) were implemented. Using them, the frequency and dose probability densities of LET L, $f(L)$ and $d(L)$, respectively, can be easily calculated. The results obtained from unit=13 and 14 are proportional to those obtained from unit=2 and 4, respectively, but their absolute values are different because the integral of the probability densities are normalized to 1 for the new options. In place of sum over, the frequency mean is output when unit=13, and the dose mean is output when unit=14, though their statistical errors are not calculated (always shown as 0.000). Note that unit=13 and 14 can be set only when axis=let.

Table 7.41: [t-let] parameters (3)

name	value	explanation
resol =	1 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness.
width =	0.5 (default)	The option defines the line thickness.
volume	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined.
reg vol		Volume definition. For details see Sec. 6.1.2.
iechrl =	72 (default)	Maximum number of columns for volume input echo.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: do not show.
stdcut =	(optional, D=-1)	Threshold value of STD cut off.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

7.12 [T-SED] section

Calculation of the probability density of deposition energies in microscopic sites, called the lineal energy y or specific energy z, is of great importance in the estimation of the relative biological effectiveness (RBE) of charged particles. However, such microscopic probability densities cannot be directly calculated by PHITS simulation using the `[t-deposit]` tally, as PHITS is designed to simulate particle motion on a macroscopic scale and employs a continuous-slowing-down approximation (CSDA) for calculating the energy loss of charged particles. We therefore introduced a special tally named `[t-sed]` for calculating the microscopic probability densities using a mathematical function that can instantaneously calculate quantities around trajectories of charged particles. The function was developed on the basis of track structure simulations considering productions of δ-rays and Auger electrons. Note that the name 'SED' derives from 'Specific Energy Distribution.' Details of the calculation procedure are given elsewhere. [116] [117] The function of `[delta ray]` section shouldn't be used together with this tally. When you use `[t-sed]`, you have to activate the EGS mode (`negs`=1 or 2) and set the cut-off energy of electrons and positrons to 10 keV or 100 keV (`emin(12-13)`=0.01 or 0.1).

Using this tally, information on the probability densities of y and z in water can be obtained. It is also possible to calculate probability densities in different materials, although the accuracy of doing so has no yet been checked. Similar to `[t-let]`, the dose is only counted from the energy loss of charged particles and nuclei, and therefore the event generator mode (`e-mode`≥1) must be used to transport low-energy neutrons. The deposition energy in microscopic sites can be expressed as the deposit energy ε in MeV, the lineal energy y in keV/μm, or the specific energy z in Gy. The definitions of these quantities are given in ICRU Report 36. Usage of `[t-sed]` is similar to that of `[t-let]`.

Table 7.42: `[t-sed]` parameters (1)

name	value	explanation
`mesh =`	`reg, r-z, xyz`	Mesh type: a mesh type subsection is required below this option.

`volume`
 	
`reg vol`
	(optional)	This option defines the volume for each region for `reg` mesh. Volume definitions are required below this option. Default values are given in input echo in the case of no definition.
		Volume definition. For details see Sec. 6.1.2.

`iechrl =`	72 (default)	Maximum number of columns for volume input echo.
`model =`	0,1 (D=0)	0: old model[116],[117] 1: new model (paper under preparation)
`se-unit =`	`0, 1, 2, 3`	Unit of deposition energy in microscopic sites. 0: Number of ionizations and electronic excitations. Selectable only when `model`=1. 1: deposit energy ε in MeV 2: lineal energy y in keV/μm 3: specific energy z in Gy
`cdiam =`	(optional, D=1.0)	Diameter of the target site in μm. A value from 0.001 to 2.0 can be specified. If `cdiam` is set to a negative value, the peak due to the production of Auger electron is ignored.
`se-type =`	`1, 2, 3, 4, 5`	ε, y or z mesh (unit is defined by `se-unit`): an energy mesh subsection (specified in `ne, emin, emax`, etc.) is required below this option. If a 'Warning: Z bin is not enough!' message occurs, the `emin, emax`, and `ne` parameters must be changed. This warning indicates that the microdosimetric function cannot calculate the y (or z) distribution because the range is too small or the mesh resolution is too poor. For example, set `se-type=3, emin=0.01, emax=10000.0, ne=60` or higher for calculating the y distribution for cite diameter = 1 μm (`cdiam=1.0`, `se-unit=2`).

[116] T. Sato, R. Watanabe and K. Niita, "Development of a calculation method for estimating the specific energy distribution in complex radiation fields," Radiat. Prot. Dosim. 122, 41-45 (2006).

[117] 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," Radiat. Res. 171, 107-117 (2009).

Table 7.43: [t-sed] parameters (2)

name	value	explanation
part =	all (default), particle name	Tally particle.
axis =	sed, reg, x, y, z, r xy, yz, xz, rz	x axis value of output data. 2-dimensional.
samepage =	(optional, D=part)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in axis can be specified.
file =	file name	Define file names. This is required by each setting of axis.
resfile =	(optional, D=file)	Define a file name of the past tally in the restart calculation: even if several axis parameters were defined, specify only one resfile.
unit =	1, 2, 3, 4, 5, 6 7,8	1: Track [cm/(keV/μm)/source] 2: Dose [MeV/(keV/μm)/source], proportional to $y * f(y)$. 3: Track [cm/ln(keV/μm)/source] 4: Dose [MeV/ln(keV/μm)/source], proportional to $y * d(y)$. 5: Track [cm/source] 6: Dose [MeV/source] 7: $y * f(y)$ [dimensionless], where $\int f(y)dy = 1$. 8: $y * d(y)$ [keV/μm], where $\int d(y)dy = 1$. These units are for the case of se-unit=2. For se-unit=1 and 3, [keV/μm] is replaced by [MeV] and [Gy], respectively.
factor =	(optional, D=1.0)	Normalization factor.

From version 3.02, new options (unit=7 and 8) were implemented. Using them, the frequency and dose probability densities of y, $f(y)$ and $d(y)$, respectively, can be easily calculated. The results obtained from unit=7 and 8 are proportional to those obtained from unit=2 and 4, respectively, but their absolute values are different because the integral of the probability densities are normalized to 1 in the cases of the new options. In place of sum over, the frequency mean is output when unit=7, and the dose mean is output when unit=8, though their statistical errors are not calculated (always shown as 0.000). Note that unit=7 and 8 can be set only when axis=sed.

Table 7.44: `[t-sed]` parameters (3)

name	value	explanation
`title =`	(optional)	Title.
`angel =`	(optional)	ANGEL parameters.
`sangel =`	(optional)	Special format for ANGEL parameters.
`2d-type =`	`1, 2, 3, 4, 5, 6, 7` (optional, D=3)	Options for 2-dimensional plot.
`gshow =`	`0` (default), `1, 2, 3, 4, 5`	When `mesh=xyz` and `axis=xy,yz,xz`, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. `gshow=5` outputs only material colors in pixel style when `icntl=8`.
`rshow =`	`0` (default), `1, 2, 3`	When `mesh=xyz` and `axis=xy,yz,xz`, region border (1), material name (2), and region name (3) are plotted using this option. A `xyz` mesh section must be added below this option.
`gslat =`	1(default), 0	1: show lattice boundary in `gshow`, 0: do not show.

`x-txt =`
`y-txt =`
`z-txt =`
	(optional)
	(optional)
	(optional)
		x axis title.
		y axis title.
		z axis title.

`resol =`
 	
`width =`
	`1` (default)
	
	`0.5` (default)
		This option multiplies the region line resolution by a factor of `resol` with the `gshow` or `rshow` option set to define the line thickness.
		The option defines the line thickness.

`trcl =`	(optional)	Coordinate transformation number or definition for `r-z` or `xyz` mesh.

Table 7.45: [t-sed] parameters (4)

name	value	explanation
material =	(optional) all, number of materials	Specify materials for scoring. all : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring.
(next line)	2 5 8	Material numbers.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
letmat =	(optional)	Material id for LET(dE/dx). If not defined, a real material is assumed. If a material not used in the geometry is selected, its material density must be defined in [material]. To calculate LET in water, define water with 1 g/cm^3 in [material]. When letmat<0 is set, PHITS automatically calculates dE/dx for water with 1 g/cm^3 for electrons and positrons: please see "ParticleTherapy" in the recommendation setting for more details.
rhomat =	(Optional, D=1.0)	Density of the material (g/cm^3) specified by letmat. The definition of rhomat is required when the material specified by letmat is other than water, although [t-sed] was originally developed for water.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.
ctmin(i) = ctmax(i) =	(optional, D=-9999) (optional, D= 9999)	Minimum value for i-th counter. Maximum value for i-th counter.
chmin(i) = chmax(i) =	(optional, D=-9999) (optional, D= 9999)	Minimum value for i-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1). Maximum value for i-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
stdcut =	(optional, D=-1)	Threshold value of STD cut off.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

7.13 [T-Time] section

This tally gives information on the number of particles of energy cut-off, escape, and decay in the time mesh [nsec]. Energy spectra of the particles can be also obtained. Especially, this is the unique tally can obtain energy spectra of only the particles of energy cut-off.

Table 7.46: `[t-time]` parameters (1)

name	value	explanation
`mesh =`	`reg, r-z, xyz`	Mesh type: a mesh type subsection is required below this option.
`part =`	`all` (default), particle name	Tally particle.
`material =`	(optional) `all`, number of materials	Specify materials for scoring. `all` : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring.
(next line)	`2 5 8`	Material numbers.
`t-type =`	`1, 2, 3, 4, 5`	Time mesh: a time mesh subsection is required below this option.
`e-type =`	`1, 2, 3, 4, 5`	Energy mesh: an energy mesh subsection is required below this option.
`unit =`	`1, 2, 3, 4`	1: [1/source] 2: [1/nsec/source] 3: [1/nsec/cm^3/source] 4: [1/nsec/cm^3/MeV/source]
`axis =`	`t, eng, reg, x, y, z, r` `xy, yz, xz, rz`	x axis value of output data. 2-dimensional.
`samepage =`	(optional, `D=part`)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in `axis` can be specified.
`file =`	`file name`	Define file names. This is required by each setting of `axis`.
`resfile =`	(optional, `D=file`)	Define a file name of the past tally in the restart calculation: even if several `axis` parameters were defined, specify only one `resfile`.

`output =`	`all`
	`cutoff`
	`escape`
	`decay`
		Energy cut off, escape, and decay particles.
		Energy cut off particles.
		Escape particles.
		Eecay particles.

`factor =`	(optional, D=1.0)	Normalization factor.
`title =`	(optional)	Title.
`angel =`	(optional)	ANGEL parameters.
`sangel =`	(optional)	Special format for ANGEL parameters.
`2d-type =`	`1, 2, 3, 4, 5, 6, 7` (optional, D=3)	Options for 2-dimensional plot.

`x-txt =`
`y-txt =`
`z-txt =`
	(optional)
	(optional)
	(optional)
		x axis title.
		y axis title.
		z axis title.

Table 7.47: [t-time] parameters (2)

name	value	explanation
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
rshow =	0 (default), 1, 2, 3	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.
resol =	1 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness.
width =	0.5 (default)	The option defines the line thickness.
volume	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined.
reg vol		Volume definition. For details see Sec. 6.1.2.
iechrl =	72 (default)	Maximum number of columns for volume input echo.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
bmpout =	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkout =	0(default), 1	Output the tally results in the xyz-mesh in the input format of ParaView. This file is named by replacing the extension with ".vtk." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkfmt =	0(default), 1	Format of output file for ParaView. 0: ascii, 1: binary.
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.

Table 7.48: [t-time] parameters (3)

name	value	explanation
dump =	Number of data	For mesh=reg, the information is dumped on the file. If dump is negative, data are written by ascii, if positive, by binary.
(next line)	Data sequence	Define the data sequence. The history information (nocas and nobch) is necessary to use idmpmode=1.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: no.
stdcut =	(optional, D=-1)	Threshold value of STD cut off.

In the [t-time] tally, the dump option can only be used only with output=cutoff. If the dump option is set, the e-type and t-type meshes take on only the maximum and minimum values. In using this dump parameter, axis and file are restricted to one axis and one file apiece and unit is always 1. The dumped data are written onto a file named "***_dmp," where "***" indicates the file name specified by file=***. The normal output of the tally is written on "***." From this file, information on the total normalization factor can be obtained; doing so requires setting one mesh each for e-type and t-type. Note that constraints based on the history counter (chmin & chmax) do not work with the dump function because the particle information is output to the dump file immediately after the condition specified by output parameter is met. The history information (nocas and nobch) is necessary to use idmpmode=1 for continuous calculation using the dump file; in addition, both the dump file with "_dmp" and the normal output file specified by file= are required to use idmpmode=1. The option dumpall is not compatible with this dump tally option when shared memory parallelization is active.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

7.14 [T-Interact] section (formerly named [T-Star])

This tally gives the number of interactions occurred in the specified regions. It was formerly named `[t-star]` because it can calculate the star density, but we changed the name from version 3.04 in order to explicitly state that it is the tally for counting the number of interactions. Note that the former name, `[t-star]`, can still be accepted even after version 3.04. To calculate the number of atomic interactions using this tally, use the EGS5 mode (`negs=1,2`). To calculate the number of nuclear interactions using nuclear data library, use the event-generator mode (`e-mode`≥1).

Table 7.49: `[t-interact]` parameters (1)

name	value	explanation
`mesh =`	`reg, r-z, xyz`	Mesh type: a mesh type subsection is required below this option.
`MorP =`	`mean, prob`(omissible)	Output the mean number (`mean`) or probability density (`prob`) of interactions. Only when `mesh=reg`, select `prob`.
`part =`	`all` (default), particle name	Reaction particle.
`material =` (next line)	(optional) `all`, number of materials `2 5 8`	Specify materials for scoring. `all` : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value, in which case the specified materials are not included for scoring. Material numbers.
`mother =` (next line)	(optional) `all`, number of mother nuclei 208Pb Pb	Specify mother nuclei. `all` : default (same as no definition). When setting the number of mother nuclei, define the mothers in the next line. The number of `mother` can be set as a negative value, in which case the specified mothers are not included for scoring. When `output=atomic`, `mother` cannot be specified. Nucleus if specified using mass: without mass, all nuclei are isotopes of Pb. To specify multiple mother groups, use multiple `[t-interact]` tallies.
`e-type =`	`1, 2, 3, 4, 5`	Energy mesh: an energy mesh subsection is required below this option.
`t-type =`	`1, 2, 3, 4, 5` (optional)	Time mesh: a time mesh subsection is required below this option.
`unit =`	`1, 2` 11, 12	1: [1/cm^3/source] 2: [1/cm^3/MeV/source] 11:[1/cm^3/nsec/source] 12:[1/cm^3/nsec/MeV/source]

`axis =`	`eng, reg, x, y, z, r`
	`xy, yz, xz, rz`
	`t`
	`act`
		x axis value of output data.
		2-dimensional.
		Time axis.
		Number of interaction (for `MorP=prob`).

`maxact =`	(optional, `D=100`)	Maximum number of interactions per history for `axis=act`. If the number of interactions exceeds this value, the history is counted as 'overflow,' whose probability is outputted just above 'sum over' line.

Table 7.50: `[t-interact]` parameters (2)

name	value	explanation
`samepage =`	(optional, `D=part`)	The type of data to be displayed on the same page of the image output file. Parameters that can be defined in `axis` can be specified.
`file =`	`file name`	Define file names. This is required by each setting of `axis`.
`resfile =`	(optional, `D=file`)	Define a file name of the past tally in the restart calculation: even if several `axis` parameters were defined, specify only one `resfile`.
`factor =`	(optional, D=1.0)	Normalization factor.
`title =`	(optional)	Title.
`angel =`	(optional)	ANGEL parameters.
`sangel =`	(optional)	Special format for ANGEL parameters.
`2d-type =`	`1, 2, 3, 4, 5, 6, 7` (optional, D=3)	Options for 2-dimensional plot.
`x-txt =`	(optional)	*x* axis title.
`y-txt =`	(optional)	*y* axis title.
`z-txt =`	(optional)	*z* axis title.
`output =`		Reaction type. Only one can be set.
	`all`	All reactions.
	`decay`	Decay reaction.
	`elastic`	Elastic scattering.
	`nuclear`	Nuclear reaction excluding elastic scattering. This corresponds to `nonela` in `[t-product]`.
	`fission`	Fission.
	`absorption`	Absorption.
	`heavyion`	Heavy Ion reaction.
	`transmut`	Reactions inducing transmutation of target nuclei.
	`atomic`	Atomic interaction excluding multiple scattering.
	`deltaray`	Delta-ray production.
	`knockelec`	Production of knock-on electrons by electrons and positrons.
	`atmflu`	Atomic fluorescence x-ray emission.
	`auger`	Auger electron emission.
	`brems`	Bremsstrahlung.
	`photoelec`	Photoelectric effect.
	`compton`	Compton scattering.
	`pairprod`	Electron-positron pair production.
	`annih`	Positron annihilation.
	`ts_hit`	All events in track-structure mode.
	`ts_ioniz`	Ionization in track-structure mode.
	`ts_bio`	Sum of the events that can cause DNA damage (ionization, electronic-excitation, electron capture, and dissociative electron attachment) in track-structure mode.
	`ts_w`	Sum of the events that generate electrons (ionization and electron stripping) in track-structure mode. The w-value can be deduced from the quotient of deposition energy and this value.

[116] PHITS-etsmode is track-structure mode in PHITS.
[117] PHITS-KURBUC is proton & carbon track-structure mode in PHITS.

Table 7.51: [t-interact] parameters (3)

		(continued in output)
		The followings are useful when you need to distinguish the contributions from each track-structure modality.
	ets_elast	Elastic scattering in PHITS-etsmode[118].
	ets_ioniz	Ionization in PHITS-etsmode.
	ets_e-exc	Electronic excitation in PHITS-etsmode.
	ets_dea	Dissociative electron attachment in PHITS-etsmode.
	ets_v-exc	Vibration excitation in PHITS-etsmode.
	ets_p-exc	Photon excitation in PHITS-etsmode.
	ets_r-exc	Rotation excitation in PHITS-etsmode.
	ets_plasmon	Plasmon excitation in PHITS-etsmode.
	ets_ioniz_e-exc	Sum of the ionization and electronic excitation in track-structure mode.
	ets_hit	All reactions in PHITS-etsmode.
	kurbuc_elast	Elastic scattering in PHITS-KURBUC[119].
	kurbuc_ioniz	Ionization in PHITS-KURBUC.
	kurbuc_e-exc	Electronic excitation in PHITS-KURBUC.
	kurbuc_e-cap	Electron capture in PHITS-KURBUC.
	kurbuc_ioniz_e-exc_e-cap	Ionization, electronic excitation and electron capture in PHITS-KURBUC.
	kurbuc_e-stp	Electron stripping in PHITS-KURBUC.
	kurbuc_hit	All reactions in PHITS-KURBUC.
	its_elas	Elastic scattering in ITSART[120].
	its_ioniz	Ionization in ITSART.
	its_e-exc	Electronic excitation in ITSART.
	its_ioniz_e-exc	Ionization and electronic excitation in ITSART.
	its_hit	All reactions in ITSART.
gshow =	0 (default), 1, 2, 3, 4, 5	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), region name (3), and LAT number (4) are plotted using this option. gshow=5 outputs only material colors in pixel style when icntl=8.
rshow =	0 (default), 1, 2, 3	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.
volume	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined.
reg vol		Volume definition. For details see Sec. 6.1.2.
iechrl =	72 (default)	Maximum number of columns for volume input echo.

[120] ITSART is generalized track-structure mode in PHITS.

Table 7.52: [t-interact] parameters (4)

name	value	explanation
resol =	1 (default)	This option multiplies the region line resolution by a factor of resol with the gshow or rshow option set to define the line thickness.
width =	0.5 (default)	The option defines the line thickness.
volmat =	(optional, D=9)	This option corrects the volume value for each mesh when material is defined by the xyz mesh. (0 means no correction). Value of volmat is the number of scans per one xyz mesh side.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, rz).
bmpout =	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkout =	0(default), 1	Output the tally results in the xyz-mesh in the input format of ParaView. This file is named by replacing the extension with ".vtk." When mesh=xyz and axis=xy, yz, xz, it is available.
vtkfmt =	0(default), 1	Format of output file for ParaView. 0: ascii, 1: binary.
maxangel =	Number of part(default)	Specify the number of part shown in the eps file. This parameter limits the number of particles shown in a tally eps file, while it does not limit in a numerical data file.
ctmin(i) =	(optional, D=-9999)	Minimum value for *i*-th counter.
ctmax(i) =	(optional, D= 9999)	Maximum value for *i*-th counter.
chmin(i) =	(optional, D=-9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chmax(i) =	(optional, D= 9999)	Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: do not show.
stdcut =	(optional, D=-1)	Threshold value of STD cut off.
nlatcel =	(optional)	By setting cell number which has LAT parameter, the number of interactions in cells, which have same cell number but placed in different lattice coordinates, are processed as different cells. It works only for MorP=prob.
nlatmem =	(optional, D=1000)	The maximum number of cells deal with separately in one history.

When coll or reaction events belonging to coll are specified in [counter], the counter value will change before the particle information has been tallied. Therefore, to tally the information at the moment the reaction event occurs, specify the counter value after the change.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

7.15 [T-Dchain] section

This tally is used for generating input files for DCHAIN. Figure 7.4 illustrates the flowchart of the connection calculation between PHITS and DCHAIN. Please see the read-me file in "\phits\recommendation\dchain\" folder for more details of the procedure for the connection. When publishing results obtained using this tally, please provide a reference to the document [121] in footnote below.

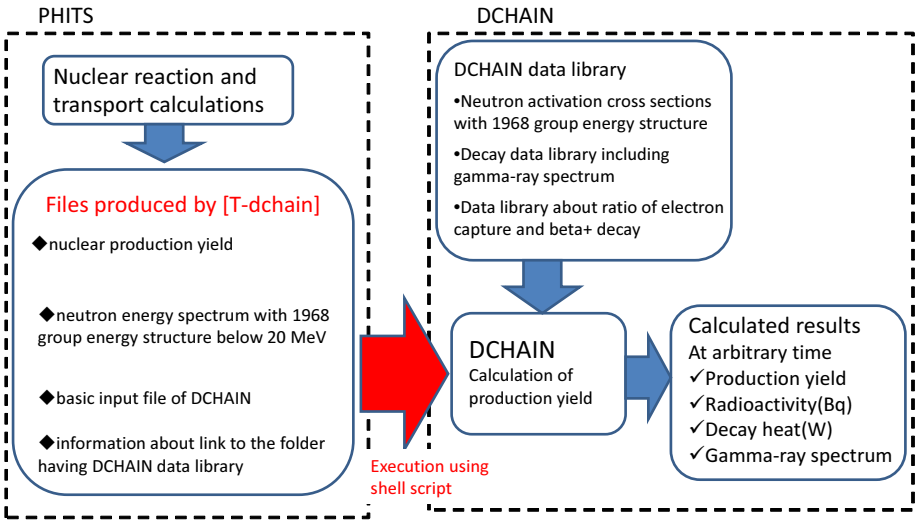


Figure 7.4: Concept of the connection calculation between PHITS and DCHAIN.

In the PHITS calculation, `[t-dchain]` automatically creates `[t-track]` and `[t-yield]` as well as the input file of DCHAIN. The `[t-track]` tally calculates the neutron energy spectra below 20 MeV with a 1968-energy-group structure. The `[t-yield]` tally calculates the nuclear production yields by protons, heavy-ions, mesons, and neutrons with energies above 20 MeV.

In the DCHAIN calculation, the neutron energy spectra are multiplied by the activation cross section contained in the DCHAIN data library. Then, the total activations are estimated by adding these results and those directly calculated by PHITS using the `[t-yield]` tally. After that, DCHAIN evaluates radioactivity, nuclide, decay heat, and the gamma energy spectrum at irradiation and cooling time.

Note that the time variation in particle transport simulation or that given by time distribution defined in the `[source]` section is independent of the time variation in the DCHAIN calculation.

Setting `e-mode`≥1 in the `[parameters]` section enables calculation of the yields of radioactive nuclides produced by low-energy neutron reactions below 20 MeV using PHITS instead of the activation cross sections contained in the DCHAIN data library. However, the accuracy of the event generator mode relative to that of the DCHAIN data library in terms of calculating the residual-nuclide yields has not been verified, and it is therefore recommended that the use set `e-mode=0` (default) in the PHITS calculation using `[t-dchain]`. Note that activations from the originally activated target are not included in the DCHAIN calculation.

From ver. 3.00, the natural isotope expansion defined in `[material]` was effective in input files of DCHAIN generated by `[t-dchain]`. Note that in the case that one nucleus is defined two (or more) times in a material, only the latter one is effective. For example, if a material is defined as follows:

```
MAT[1] Fe 1  56Fe 1
```

a contribution of `56Fe` included in `Fe` is ignored in the DCHAIN calculation.

[121] Tetsuya Kai, et al., "DCHAIN-SP 2001: High Energy Particle Induced Radioactivity Calculation Code", JAERI-Data/Code-2001-016 (2001) in Japanese

Table 7.53: [t-dchain] parameters (1)

name	value	explanation
mesh=	reg, xyz, tet	Mesh type (Currently ONLY reg, xyz, tet mesh can be specified): a mesh type subsection must be added below this option. (reg = cell number).
file=	file name	Input file name of DCHAIN. Any extension except .dtrk, .dyld, or .dout can be used.
title=	(optional)	Title.
stdcut=	(optional, D=-1)	Threshold value of STD cut off.
ndata =	0, 1, 2(default), 3	Option to replace the results of nuclear reaction events with particle production cross section (activation cross section) data. 0: This option is not available. 1: Replace with cross section data in nucleon-induced reactions on α, ^{14}N, ^{16}O targets. 2: Replace with cross section data in a directory specified by file(27) (D=file(1)/XS/yield/) in proton, neutron, deuteron, alpha-induced, and photo-nuclear reactions when using the evaluated nuclear data library without e-mode. 2: Replace with cross section data in a directory specified by file(27) (D=file(1)/XS/yield/) in proton, neutron, deuteron, alpha-induced, and photo-nuclear reactions when using the evaluated nuclear data library with or without e-mode or nuclear reaction models.
iredufmt =	0,1(default)	Option to toggle between the old default format and the new much more space-efficient format for the [T-Track] and [T-Yield] tally outputs produced by [T-Dchain] When iredufmt=0, use the old format (same as the normal format for those tallies). When iredufmt=1, use the new more space efficient format specific to [T-Dchain]. This is very strongly encouraged for large xyz and tet mesh problems.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

See Sec. 7.8 for ndata.

Table 7.54: [t-dchain] parameters (2)

name	value	explanation
timeevo= (next line)	number time and factor	Number of irradiation and cooling steps in DCHAIN calculation. time: time step of irradiation and/or cooling. factor: normalized factor for beam intensity. Time should be calculated from the end of the last step and not from the start of the first irradiation. The allowable units are seconds (s), minutes (m), hours (h), days (d), and years (y). One (or more) blank character must be placed between the number indicating the time and the unit. There must also be a space between the unit and the factor. *See example of input for [t-dchain] tally in Example 61.
outtime= (next line)	number time	Number of output timings in the DCHAIN calculation. Output timing. If a positive value is given as the output timing, it is calculated from the start of the first irradiation step. If a negative value is given, it is calculated from the end of last irradiation step. Except for this positive and negative rule, the format for specifying the timing is the same as that for timeevo. The timing cannot be specified until after all steps defined in timeevo are finished. To output the timing when no radioactive nuclide exists, e.g., 0.0 m, set iprtb2=0 in the [t-dchain] section.
amp=	(optional,D=1.0)	Power of source (source/second). If totfact is specified in the [source] section, please adjust amp so that amp×totfact is actual source intensity, which is in the unit of source/second. When you use the RI source with norm=0 (default setting), you should set amp=1 because the totfact factor automatically includes a factor in the unit of source/sec.
target =	0, 1 (optional,D=0)	0: OFF; Not necessary to write target information: the information is automatically determined from the [material], [cell], and [volume] sections. 1: ON; Necessary to write target information. *To add nuclides that are not defined in the [material] in the DCHAIN calculation, and/or the volume is not set in the [volume], write target=1 and provide the related information as in Example 62. *See example of target subsection for target=1 in Example 62.
iertdcho =	1 (default), 0	Toggles propagation of PHITS's [t-yield] and [t-track] statistical uncertainties (if using iredufmt=0, yield uncertainties are separately found in the "***_err.dyld" output file) through the DCHAIN decay and buildup calculations. When iertdcho=1, the uncertainties are propagated through the DCHAIN calculation and reported in extra columns in the primary activation output file ("spd-act.out" / "***.act"). When iertdcho=0, the uncertainties are not propagated and the extra columns for the uncertainties are omitted, leaving the file with the same formatting from before the implementation of this functionality.
itdecs =	(D=1) 0 1 (default)	Option for the decrease of target nuclides due to spallation nuclear reaction. Ignore the decrease of target nuclides due to spallation nuclear reaction. Consider.
itdecn =	(D=1) 0 1 (default)	Option for the decrease of target nuclides due to nuclear reaction induced by neutrons below 20 MeV. Ignore the decrease of target nuclides due to nuclear reaction induced by neutrons below 20 MeV. Consider.

Table 7.55: `[t-dchain]` parameters (3)

name	value	explanation
`inxslib =`	(D=100)	Options for selection of neutron reaction cross section library to be used by DCHAIN.
	0,40	JEFF-3.1A (DCHAIN-SP's default from 2009-2019)
	1	FENDL/A-2.0 (DCHAIN-SP's default from 2001-2009). (only works with legacy simulations using the old 175-group T-Track neutron flux tallies)
	2,20	JENDL/AD-2017
	21	JENDL-4.0[122]
	30	ENDF/B-VII.1[123]
	31	ENDF/B-VIII.0[124]
	41	JEFF-3.3[125]
	50	FENDL/A-3.0[126]
	51	EAF-2010[127]
	60	BROND-3.1[128]
	70	CENDL-3.1[129]Note that missing isotopic data for S, Cl, K, Ca, V, Zn, W, Hg, and Tl.
	90	TENDL-2017[130]Note that due to its large size, this library is available separately by request to PHITS office.
	100	hybrid library composed of, in order: JENDL-5 + JENDL-4.0 + ENDF/B-VIII.0 + JEFF-3.3 + FENDL/A-3.0
	101	hybrid library of: ENDF/B-VIII.0 + JENDL-5 + JENDL-4.0 + JEFF-3.3 + FENDL/A-3.0
	102	hybrid library of: JEFF-3.3 + FENDL/A-3.0 + ENDF/B-VIII.0 + JENDL-5 + JENDL-4.0
	120	hybrid library of: JENDL/AD-2017 + JENDL-4.0 + ENDF/BVIII.0 + JEFF-3.3 + FENDL/A-3.0 (The default library before version 3.34)
	-1	Custom library whose name is specified with the `hnxslib` parameter (see DCHAIN manual for more details)

Hybrid libraries consist of all reactions in the first listed library plus the reactions in each subsequently listed library not present in the earlier ones.

[122] K. Shibata et al., "JENDL-4.0: A New Library for Nuclear Science and Engineering," Journal of Nuclear Science and Technology, 48, 1, 1-30 (2011).
[123] M. B. Chadwick et al., "ENDF/B-VII.1 Nuclear Data for Science and Technology: Cross Sections, Covariances, Fission Product Yields and Decay Data," Nuclear Data Sheets, 112, 12, 2887 - 2996 (2011), special Issue on ENDF/B-VII.1 Library.
[124] D. A. Brown, et al., "ENDF/B-VIII.0: The 8th Major Release of the Nuclear Reaction Data Library with CIELO-project Cross Sections, New Standards and Thermal Scattering Data," Nuclear Data Sheets, 148, 1 - 142 (2018), special Issue on Nuclear Reaction Data.
[125] A. Plompen et al., "The Joint Evaluated Fission and Fusion Nuclear Data Library, JEFF-3.3," Submitted to The European Physical Journal (2019).
[126] R. A. Forrest, et al., "FENDL-3 Library-Summary Documentation," Tech. Rep. INDC(NDS)-0628, International Atomic Energy Agency (Dec. 2012).
[127] J.-C. Sublet, et al., "The European Activation File: EAF-2010 neutron-induced cross section library," Tech. Rep. CCFE-R (10) 05 (2010).
[128] A. I. Blokhin, et al., "New version of neutron evaluated data library BROND-3.1," Yad. Reak. Konst, 2, 2, 62 (2016).
[129] Z. G. Ge, et al., "The updated version of Chinese evaluated nuclear data library (CENDL-3.1)," J. Korean Phys. Soc, 59, 2, 1052-1056 (2011).
[130] D. Rochman et al., "The TENDL library: hope, reality and future," EPJ Web Conf., 146, 02006 (Sep. 2017).

Table 7.56: [t-dchain] parameters (4)

name	value	explanation
idcylib =	(D=5)	Options for selection of decay data library to be used by DCHAIN.
	0	A hybrid library composed of evaluations and revisions from EAF-3.1, FENDL/D-1, ENSDF (1997), Table of Isotopes 8th ed. (1996), and Chart of the Nuclides 1996. (DCHAIN-SP's only decay library from 2001-2019)
	2	JENDL/DDF-2015
	3	ENDF/B-VIII.0
	4	Hybrid library of ENDF/B-VIII.0 where available and JENDL/DDF-2015 for nuclides exclusive to that library.
	5	Hybrid library of JENDL/DDF-2015 where available and ENDF/B-VIII.0 for nuclides exclusive to that library.
acmin	(D=0.0)	Threshold/cut-off value of radioactivity/inventory which must be exceeded for a nuclide to be reported in DCHAIN output.
	= 0.0	Threshold automatically set to the total radioactivity times 10^{-10}.
	> 0.0	Threshold radioactivity in [Bq].
	< 0.0	$-1\times$ The threshold inventory per unit volume in [atoms/cm^3].
istabl	(D=0)	Toggle printing of stable nuclides in the output files.
	= 0 (default)	Not printed.
	= 1	Printed.
mxnuclei =	3000(default)	Maximum number of nuclides to score. 0: Score the yield of all nuclides on the nuclear chart (136 charge numbers * 236 mass numbers * 3 meta stables = 96,288). >0: Score the yield of nuclides up to this number, and ignore the rest by outputting a warning message. A smaller mxnuclei results in smaller size of allocated memory and shorter computational time, but larger chance to miss the yields of some important nuclides.
iphtout	(D=0)	Output option of PHITS [source] section from DCHAIN. 0 or 1: No output for [source] section based on RI source. 10 or 11: RI source with proj=all. 20 or 21: RI source with proj=photon. 30 or 31: RI source with proj=electron. 40 or 41: RI source with proj=positron. 50 or 51: RI source with proj=alpha. When the last digit of iphtout is 1, [source] based on DCHAIN's gamma-ray database written in e-type=4 is also output (e.g. both RI source with proj=all and DCHAIN database sources are generated when iphtout=11). When iphtout is specified in a negative value, [source] sections for each output timing are separately generated in different files ("*_t1.pht,""*_t2.pht"$\cdots$).
iaonucl	(D=0)	Number of nuclides shown in the EPS file generated by DCHAIN, corresponding to angelout_nucludes in DCHAIN.
aonucl		Name of nuclides shown in the EPS file generated by DCHAIN. e.g. Cs-137 Ba-137m H-3
aoreg		Region numbers shown in the ESP file generated by DCHAIN, corresponding to angelout_region in DCHAIN.
dversion	(D=1)	Version of DCHAIN
	= 0	DCHAIN-SP
	= 1	DCHAIN-PHITS
	= 2	DCHAIN-PHITS with user-defined neutron energy bin specified in "data/dchain_EnGroup.dat"

In addition to the above parameters, the DCHAIN parameters can also be specified in [t-dchain] section. The specifiable parameters are:

imode, jmode, idivs, iregon, inmtcf, ichain, itdecs, itdecn, isomtr, ifisyd, ifisye, iyild, iggrp, ibetap, acmin, istabl, igsdef, igsorg, ebeam, prodnp, hnxslib, hdcylib, iwrtchn, chrlvth, iwrchdt, iwrchss, idosecf, ixsrall, irdonce, foamout, foamvals, ipltmode, ipltaxis

The respective meanings of these parameters along with more detailed explanations of DCHAIN's inputs and outputs are given in the DCHAIN manual, which you can find in the "\phits\dchain-sp\manual\" folder.

Example 61: Example of input for [t-dchain] tally

```
 1:        mesh = reg                    <-region mesh
 2:         reg = 100                    <-cell number
 3:        file = testDC.spd             <-file name of DCHAIN input
 4:       title = [t-dchain] test calc.
 5:         amp = 1.0E12                 <-source power (source/sec)
 6:
 7:     timeevo = 4                      <-number of irradiation and cooling steps
 8:              3.0 h  1.0              <-irradiation for 3 hours
 9:              2.0 h  0.0              <-cooling for 2 hours
10:              3.5 h  1.0              <-irradiation for 3.5 hours
11:             15.5 h  0.0              <-cooling for 15.5 hours
12:
13:     outtime = 3                      <-number of output timing
14:              3.0 h                   <-3 hours later from the 1st irradiation start time
15:             -1.0 h                   <-1 hour later from the end of the last irradiation step
16:             -3.0 h                   <-3 hour later from the end of the last irradiation step
```

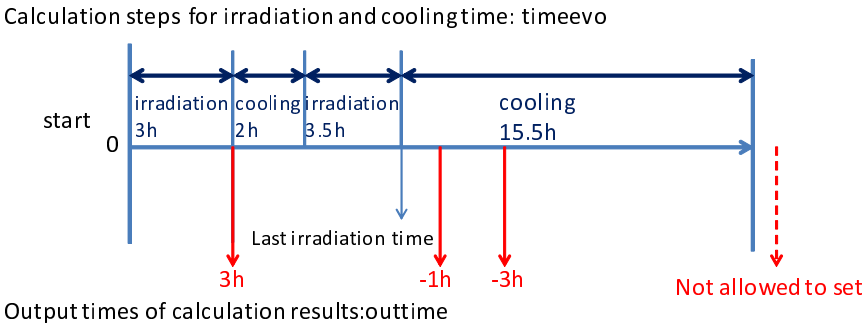


Figure 7.5: Relation between steps for irradiation and cooling and output times.

Example 62: Example for the setting of target material compositions and volumes for target=1

```
           .......
:       target = 1                     <-target material composition ON
:       non    reg    vol              <-omissible
:         1     1   8000.0             <-serial number, cell number, volume
:      tg-list = 2                     <-number of the nuclides
:              H-1   6.689E-02         <-Element ID, Atomic Number,
:              O-16  3.345E-02           and Density of the atom (10^24/cm^3)
:       2      2   2000.0             <-serial number, cell number, volume
:      tg-list = 1                     <-Number of the nuclides
:             Fe-56  8.385E-02
```

To indicate an isotope, the symbol of the chemical element must be connected with its atomic number using the character '-.'

Important notices for using [t-dchain]:

- The following parameters must be defined in the [parameters] section:
  - file(21): set the placement of the data folder for DCHAIN.

- The volume of each tally region must be defined in the [volume] section.

Files generated by [t-dchain] are listed below.

- The basic input file of DCHAIN: file name is set in the [t-dchain] tally.
- Neutron energy spectra with 1968 energy groups below 20 MeV : ***.dtrk
- Nuclear production yields: ***.dyld
- Information on the link to the folder containing the DCHAIN data library is in “dch_link.dat.”

Note that when DCHAIN is executed, files of names shown below are deleted.
“yield.out, out-gsdef, out-gamsporg, out-allreg, spd-act.out, angel-data.ang, out-phits, out-dcychains”

7.16 [T-WWG] section

This tally gives parameters used for the [Weight Window] section. The tally serves as a Weight Window Generator (WWG) to automatically obtain effective settings of [Weight Window] for a user-defined virtual space. When using this function, a test calculation using an input file including the [t-wwg] tally must be performed. Long calculations can then be performed using the [Weight Window] section generated from this test calculation. In the case of mesh=reg, you have to give the volume of each cell in [volume] section. When you perform the restart calculation or use the sumtally function with an input file including [t-wwg], set axis other than wwg and ireschk=1 in the [parameters] section.

This tally scores the fluence of particles in each history j, particle k, energy mesh l, and geometrical mesh m, denoted as Φ_{jklm}. Then, at the end of each batch, it determines the lower limit of their weight window, $W_{klm,\mathrm{low}}$, as written below:

$$W_{klm,\mathrm{low}} = C_{kl}\left[\left(\Sigma_j\Phi_{jklm}\right)_{k,l,x\%\mathrm{ile}} + \Sigma_j\Phi_{jklm}\right], \tag{38}$$

where $\left(\Sigma_j\Phi_{jklm}\right)_{k,l,x\%\mathrm{ile}}$ is the x-percentile value of $\Sigma_j\Phi_{jklm}$ for particle k and energy mesh l among all geometrical mesh m. The value of x can be determined by an input parameter named pedestal whose default value is 0.1 that indicates 10 percentiles. The basic idea of this equation is to make $W_{klm,\mathrm{low}}$ proportional to $\Sigma_j\Phi_{jklm}$, but the introduction of the pedestal term, $\left(\Sigma_j\Phi_{jklm}\right)_{k,l,x\%\mathrm{ile}}$, is required to avoid extremely low $W_{klm,\mathrm{low}}$ that could cause excessive particle splitting.

C_{kl} in Eq. (38) is the normalization constant which depends on an input parameter named normww as written below:

$$C_{kl} = \begin{cases} \dfrac{W_{\max}}{\left[\left(\Sigma_j\Phi_{jklm}\right)_{k,l,x\%\mathrm{ile}}+\left(\Sigma_j\Phi_{jklm}\right)_{k,l,\max}\right]} & \text{for normww} = 0 \\ \dfrac{W_{\max}}{\left[\left(\Sigma_j\Phi_{jklm}\right)_{k,l,x\%\mathrm{ile}}+\left(\Sigma_j\Phi_{jklm}\right)_{k,l,\max}\right]}\dfrac{\left(\Sigma_{jm}\Phi_{jklm}\right)_{k,l}}{\left(\Sigma_{jm}\Phi_{jklm}\right)_{k,l,\max\text{ or }\min}} & \text{for normww} = 1 \text{ or } -1 \\ \dfrac{W_{\max}}{\left[\left(\Sigma_j\Phi_{jklm}\right)_{k,l,x\%\mathrm{ile}}+\left(\Sigma_j\Phi_{jklm}\right)_{k,l,\max}\right]}\dfrac{\left(\Sigma_{jm}\Phi_{jklm}\right)_{k,l}}{\left(\Sigma_{jm}\Phi_{jklm}\right)_{\max\text{ or }\min}} & \text{for normww} = 2 \text{ or } -2 \end{cases} \tag{39}$$

where $\left(\Sigma_j\Phi_{jklm}\right)_{k,l,\max}$ is the maximum value of $\Sigma_j\Phi_{jklm}$ for k and l, $\left(\Sigma_{jm}\Phi_{jklm}\right)_{k,l}$, $\left(\Sigma_{jm}\Phi_{jklm}\right)_{k,l,\max\text{ or }\min}$ and $\left(\Sigma_{jm}\Phi_{jklm}\right)_{\max\text{ or }\min}$ are the spatial-integrated fluence and their maximum or minimum values, respectively, and $W_{\max}$ is an input parameter named wwmax whose default value is slightly below 1 (0.99) to avoid the split of particle with the weight of 1.0. The normww parameter is an option that determines whether the number of Monte Carlo (MC) fluence, i.e., fluence without considering the particles ' weight, is adjusted among particles and energy groups. In the case of normww=1, the generated [weight window] is adjusted so that the MC fluences in each energy group are equivalent. Thus, if you want to focus on high-energy neutron transport, you can narrow the energy-mesh width of high-energy neutrons and set normww=1, so that $W_{klm,\mathrm{low}}$ for that high-energy group will be smaller overall.

The fundamental concept of [t-wwg] is to generate a [weight window] section that can distribute MC fluence as uniformly as possible to all locations within the regions defined in [t-wwg]. On the other hand, if you want to evaluate the dose in a certain region of interest (ROI) with high accuracy in a short time, you can generate a [weight window] that can navigate particles to the ROI by combining the history counter and the maximum biasing factor, $B_{\max}$, defined by the chwei parameter. For activating this algorithm, you must define the [counter] section to increase a counter value when a particle enters the ROI and specify the chwei(i) parameter in [t-wwg], where i is the counter number defined in [counter]. Then, this tally determines the biasing factor for each mesh, B_{klm}, as written below:

$$B_{klm} = \frac{\Sigma_j H_{ij}\varphi_{jklm}}{\left(\Sigma_j H_{ij}\varphi_{jklm}\right)_{k,l,\max}}\left[B_{\max} - 1\right] + 1. \tag{40}$$

where H_{ij} is the i-th history-counter value of history j, and φ_{jklm} is the MC fluence for the condition. The strategy behind this equation is to maximize the biasing factor up to $B_{\max}$ =chwei(i) at the geometrical mesh m that gives the maximum value of $\Sigma_j H_{ij}\varphi_{jklm}$, i.e. the location where the MC particles pass through most frequently before (or after) arriving at the ROI, while minimizing the factor to 1 where $\Sigma_j H_{ij}\varphi_{jklm}$=0. The lower limit of the weight window, $W_{klm,\mathrm{low}}$, is calculated by replacing $\Sigma_j\varphi_{jklm}$ with $\Sigma_j\Phi_{jklm}/B_{klm}$ in Eqs. (38,39). Please see the literature[131] and the lecture notes[132] in more detail.

[131] Sato et al. Nucl. Instr. & Meth. B, 2024; 557: 165535.
[132] "phits/lecture/advanced/weightB" & "phits/lecture/advanced/shielding"

The [t-wwg] parameters are formatted as follows.

Table 7.57: [t-wwg] parameters (1)

name	value	explanation
mesh =	reg, xyz, tet	Mesh type: only reg, xyz and tet can be set; a mesh type subsection is required below this option.
part =	all (default), particle name	Tally particle.
material =	(optional) all, number of materials	Specify materials for scoring. all : default (same as no definition). To set number of materials, define the material numbers in the next line.
(next line)	2 5 8	Material numbers.
e-type =	1, 2, 3, 4, 5	Energy mesh: an energy mesh subsection is required below this option.
t-type =	1, 2, 3, 4, 5 (optional)	Time mesh: a time mesh subsection is required below this option.
unit =	1(optional)	1: [1/cm^2/source]
axis =	eng, reg, tet, t xy, yz, xz wwg	x axis value of output data. 2-dimensional. They are the same as [t-track]. To obtain [Weight Window] section, set axis=wwg. To perform restart calculation, set two (or more) axis parameters, and then set the first axis to reg, eng, or t.

Only reg, xyz and tet can be set as mesh in this tally, because the parameters in [Weight Window] are defined only for these mesh types. Note that the surfaces of xyz-mesh should be slightly shifted from surfaces of cells. Please be especially careful if you are using parameters to link geometry boundary and xyz-mesh ranges. After version 3.34, if the maximum and minimum xyz mesh values for [weight window] and [t-wwg] are defined as integers, the minimum value is shifted by $-1*$deltxyz and the maximum value by $+2*$deltxyz from the integer (see FAQ 2.8). mesh=tet can be only used for geometry containing tetrahedral-mesh geometry. The weight window factors for each constituent tetrahedron element for the tetrahedral-mesh geometry can be obtained. In short, axis should be set wwg to obtain the parameters for [Weight Window]. Although eng, reg, tet, xy, yz, xz, and t can also be set, their results are not related with [Weight Window]. axis= xy, yz, or xz are valid only with rshow=1.

Table 7.58: [t-wwg] parameters (2)

name	value	explanation
file =	file name	Define file names. This is required by each setting of axis.
resfile =	(optional, D=file)	Define a file name of the past tally in the restart calculation: even if several axis parameters were defined, specify only one resfile.
factor =	(optional, D=1.0)	Normalization factor.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
rshow =	0 (default), 1, 2, 3	When mesh=xyz and axis=xy,yz,xz, region border (1), material name (2), and region name (3) are plotted using this option. A xyz mesh section must be added below this option.
resol =	1 (default)	This option multiplies the region line resolution by a factor of resol with the rshow option set to define the line thickness.
width =	0.5 (default)	The option defines the line thickness.
volume	(optional)	This option defines the volume for each region for reg mesh. Volume definitions are required below this option. Values defined in [volume] are used in the case of no definition. If special description such as using ( ) is used to specify a region in reg=, its internally defined region number is output in input echo when this volume subsection is not defined.
epsout =	0 (default), 1, 2	When epsout=1, results are plotted into eps files. The eps file is named by replacing the extension with ".eps." When epsout=2, error bars are also displayed in the eps file, except for the 2-dimensional type (axis=xy, yz, xz, wwg).
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: do not show.
stdcut =	(optional, D=-1)	Threshold value of STD cut off.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

Table 7.59: [t-wwg] パラメータ (3)

name	value	explanation
pedestal =	(optional, D=0.1)	Percentile value (x in Eq. (38)) / 100 for calculating the pedestal of $W_{klm,\mathrm{low}}$.
normww	(optional, D=-2)	Normalization option (see Eq. (39)). 0: No adjustment. ±1: Adjust C_{kl} so that the MC fluences in each energy group are equivalent. ±2: Adjust C_{kl} so that the MC fluences in each particle and energy group are equivalent.
wwmax	(optional, D=0.99)	規格化定数。Normalization constant corresponding to the maximum value of $W_{klm,\mathrm{low}}$ (see Eq. (39)).

When normww is set to a positive value, the overall weight window values become lower, so the weight of each particle is controlled primarily by particle splitting. Conversely, if normww is set to a negative value, the overall weight window values become higher, leading to more frequent Russian roulette. For calculation conditions with large variance per history, such as the case of sources with a wide energy distribution, a negative value will give better statistics per unit computational time, since the calculation time per history will be shorter. This is the reason why normww=-2 is selected as the default value. However, care should be taken to define particle type and energy mesh in the case of normww=-2, as most most particles are killed by Russian roulette if there is even one group with extremely low particle fluence in each particle and energy group. For the part parameter, it is recommended to set all or neutron photon in the case of neutron & photon only transport calculations. Regarding the energy groups, it is recommended to set the number of energy bins just a few (or less), with a slightly narrower group width for the energy bands that contribute the most to the tally. If you are not sure which setting is optimal, we encourage you to use part=all and ne=1. In that case, the calculation results will not depend on normww.

Table 7.60: [t-wwg] parameters (4)

name	value	explanation
ctmin(i) = ctmax(i) =	(optional, D=-9999) (optional, D= 9999)	Minimum value for *i*-th counter. Maximum value for *i*-th counter.
chmin(i) = chmax(i) =	(optional, D=-9999) (optional, D= 9999)	Minimum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1). Maximum value for *i*-th history-counter. This parameter cannot be specified in the batch variance mode (istdev=1).
chwei(i) =	(optional, D=0)	Magnitude of particle-navigation capability in the generated [weight windows] using i-th history counter, i.e., $B_{\max}$ in Eq. (40). Only one of these parameters can be defined per [t-wwg], even if their counter IDs are different. The restart calculation or use the sumtally function is now allowed when this parameter is defined.
chplane=	(optional, D=all)	The plane (xy, yz, or xz) for calculating the maximum value of $\left(\Sigma_j H_{ij}\varphi_{jklm}\right)_{k,l,\max}$ in Eq. (40). The default value is all, which indicates that the maximum value is calculated from all geometrical meshes. When using particle-navigation function for duct streaming calculations, specifying a plane perpendicular to the duct will reduce the computation time. This parameter is valid only when mesh=xyz and chwei is defined.
elowthre	(optional, D=0)	Threshold energy [MeV] for the low-energy unbiasing method. Below this energy, the lower limit of the weight window is multiplied by elowthre. If it is set to 0, the low-energy unbiasing method is not used. Note that number of energy bin (ne) must be set to 1 when using this method.
elowbias	(optional, D=5.0)	The multiplying factor for the lower limit of weight window for energies below elowthre.

An example output of this tally is as follows.

```
[ Weight Window ]
     mesh =  reg

set:  c71[0.0]  c72[c71+2.13496E-08]  c73[1.25478E-04]
set:  c74[0.0]  c75[c74+9.29569E-09]  c76[2.08987E-04]

    part = neutron
     eng =  2
          1.00000E-03   1.00000E+03

    reg          ww1                    ww2
    1   (8.94092E-05+c71)/c73  (2.08987E-04+c74)/c76
    2   (1.25478E-04+c71)/c73  (1.26817E-04+c74)/c76
    3   (8.53835E-05+c71)/c73  (5.78131E-05+c74)/c76
```

In this sample, the [Weight Window] parameters for neutrons with two energy bins are output. In principle, it is not necessary to change these parameters, but c71 or c74 should be specified when adding a constant value for each Weight Window.

7.17 [T-WWBG] section

This tally gives parameters used for the [ww bias] section. The tally serves as a Weight Window Bias Generator (WWBG) to automatically obtain effective settings of [ww bias], which is used to bias [weight window]. The [ww bias] function is useful when the [weight window] parameters for a certain region are biased after automatically generating the [weight window] section by [t-wwg]. Figure 5.50 shows a flowchart how to perform the transport calculation using [weight window] and [ww bias]. First, generate the [weight window] section by [t-wwg]. If the obtained parameters of [weight window] are enough to effectively use the variance reduction technique, the [ww bias] section is not needed. However, if they are not enough, the variance reduction technique can be used more effectively by biasing the [weight window] parameters for a certain region with [ww bias]. There are two methods to set the [ww bias] section; an automatic method using [t-wwbg], which is a Weight Window Bias Generator (WWBG) and will be explained below, and a manual method by a user. Only one [t-wwbg] section can be set in one input file. [t-wwbg] is effective only when icntl=15 in [parameters]. For details of [ww bias], see Sec. 5.20. When performing a transport calculation with [ww bias] and [weight window], set icntl=0 and iwwbias=1 in [parameters].

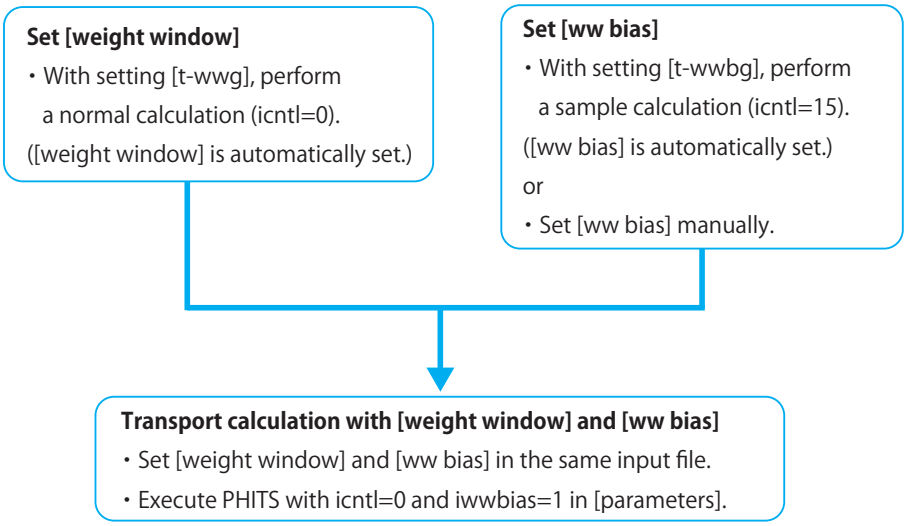


Figure 7.6: The flowchart of the connection calculation between [weight window] and [ww bias].

[t-wwbg] determines the bias values in stages of some cylindrical regions, which are with a central axis on a vector defined by two points (x0,y0,z0) and (x1,y1,z1), shown in Fig. 7.7. Figure 7.7 shows a cross-section view of three different size cylinders. To define the cylindrical regions, the parameters n-mesh, r-mesh, z-mesh, and f-mesh are also required. n-mesh is the number of the cylinders. The differences of radii and heights of the cylinders are given by r-mesh, z-mesh, respectively. In these parameters, the same number of values as the n-mesh must be given. The bias values can be set from the inside of the cylindrical regions in f-mesh. f-mesh must be set of (n-mesh)+1. The last value of f-mesh is the bias value in the outside of the cylinders. The

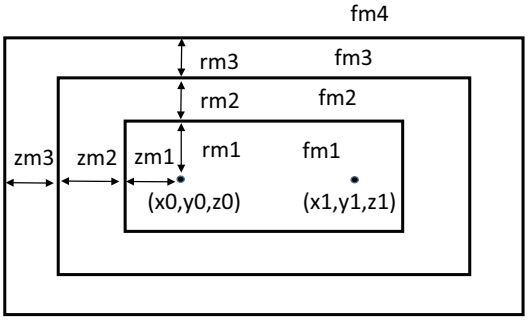


Figure 7.7: Parameters to define cylindrical regions.

cylindrical regions can be defined regardless of the geometry to calculation of the particle transport simulation. Note that if the defined cylindrical regions overlap the outer void given in `[cell]`, the outer void region can be extended by the `r-out` parameter.

The `[t-wwbg]` parameters are formatted as follows.

Table 7.61: `[t-wwbg]` parameters

name	value	explanation
`mesh =`	`reg`, `xyz`, `tet`	Mesh type: only `reg`, `xyz` and `tet` can be set; a mesh type subsection is required below this option.
`part =`	`all` (default), particle name	Tally particle.
`e-type =`	`1, 2, 3, 4, 5`	Energy mesh: an energy mesh subsection is required below this option.
`t-type =`	`1, 2, 3, 4, 5` (optional)	Time mesh: a time mesh subsection is required below this option.
`axis =`	`xy, yz, xz, wwbg`	Restart calculation cannot be performed. When `xy, yz, zx`, the `rshow` option is needed.
`file =`	`file name`	Define file names. This is required by each setting of `axis`.
`angel =`	(optional)	ANGEL parameters.

`x-txt =`
`y-txt =`
`z-txt =`
	(optional)
	(optional)
	(optional)
		x axis title.
		y axis title.
		z axis title.

`rshow =`	`0` (default), `1, 2, 3`	When `mesh=xyz` and `axis=xy,yz,xz`, region border (1), material name (2), and region name (3) are plotted using this option. A `xyz` mesh section must be added below this option.

`resol =`
 	
`width =`
	`1` (default)
	
	`0.5` (default)
		This option multiplies the region line resolution by a factor of `resol` with the `rshow` option set to define the line thickness. The option defines the line thickness.

`epsout =`	`0` (default), `1`	When `epsout=1`, results are plotted into eps files. The eps file is named by replacing the extension with ".eps."
`x0, y0, z0 =`		x, y, z coordinates of an initial point.
`x1, y1, z1 =`		x, y, z coordinates of a terminal point.
`n-mesh =`	n	The number of cylindrical regions.
`r-mesh =`	rm1,rm2,$\cdots$,rmn	The difference of radii of cylinders. The number of `n-mesh` must be set.
`z-mesh =`	zm1,zm2,$\cdots$,zmn	The difference of heights of cylinders. The number of `n-mesh` must be set.
`f-mesh =`	fm1,fm2,$\cdots$,fmn,fm(n+1)	Bias values in cylindrical regions. The number of `(n-mesh)`+1 must be set.
`r-out =`	(omissible, D=0)	Radius to extend the outer void [cm].

The same region specifications are required to use [ww bias] with [weight window] and therefore the same region definitions are required between [t-wwg] and [t-wwbg] to create [weight window] automatically using [t-wwg].

The following is an example of [t-wwbg].

Example 63: Example of [t-wwbg].

```
 1:   [ T - WWBG ]
 2:        mesh =  xyz
 3:      x-type =   2
 4:          nx =  30
 5:        xmin = -75
 6:        xmax =  75
 7:      y-type =   2
 8:          ny =  30
 9:        ymin = -75
10:        ymax =  75
11:      z-type =   2
12:          nz =  40
13:        zmin =   0.0
14:        zmax = 200
15:        axis = wwbg
16:        file = wwbg.out
17:        part =  neutron
18:      e-type =   1
19:          ne =   2
20:                0.0  1e-3  1.0
21:          x0 =  0.0
22:          y0 = -20
23:          z0 =  50
24:          x1 =  0.0
25:          y1 =  20
26:          z1 =  150
27:      n-mesh =   3
28:      r-mesh =   10  10  10
29:      z-mesh =   10  10  10
30:      f-mesh =   1.0  0.5  0.1  0.05
31:       r-out =  1000
```

In this example, the initial and terminal points are (0, -20, 50) and (0, 20, 150), respectively. The three cylindrical regions are defined with the central axis on the vector defined the two points. Figure 7.8 shows a spatial distribution of the bias values given by the example input. The radii of the three cylinders are defined by r-mesh, the differences of them are 10cm. The heights of the cylinders increase by 20cm, which is two times of 10cm given by z-mesh. Bias values are given as 1.0, 0.5, 0.1, and 0.05 from the inside of the cylindrical regions. The change of the biases is shown in Fig. 7.8.

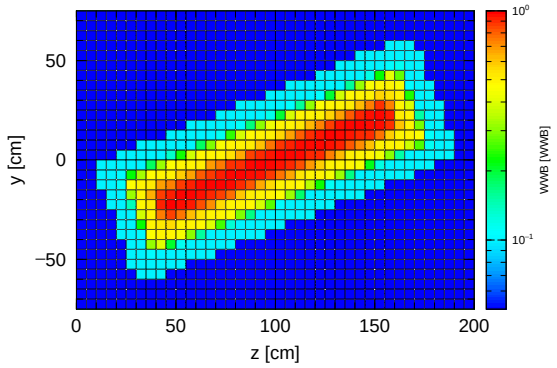


Figure 7.8: Spatial distribution of bias values given by Example 63.

7.18 [T-Volume] section

When you set mesh=reg in tally sections, you have to give volumes of each cell to estimate results in the unit of (/volume). PHITS has a function of automatic calculation to obtain the volumes using [t-volume]. In this function, PHITS calculates the volumes by performing a Monte Carlo integration with setting the material of each cell to be void and a special source region.

[t-volume] is effective only when setting icntl=14 in [parameters] section. Only one [t-volume] section can be set in one input file.

The format of parameters in [t-volume] is as follows.

Table 7.62: [t-volume] parameters

name	value	explanation
mesh =	reg	Mesh type: only reg can be set; a mesh type subsection is required below this option.
file =	file name	Define file names.
resfile =	(omitted, D=file)	Define a file name of the past tally in the restart calculation.
r-out =	(omitted, D=0.0)	Radius of the outer void. [cm]
title =	(omitted)	Title.
method =	(omitted, D=0)	Option for Monte Carlo integration. 0: By particle track lengths. 1: By source points.
s-type =	1, 2	Source type for volume calculation. 1: Sphere source. The coordinates of the sphere, (x0, y0, z0), and its radius, r0, have to be defined. 2: Rectangular solid source. The coordinates x0, x1, y0, y1, z0, and z1 have to be defined.
x0, y0, z0, x1, y1, z1, r0 =		Coordinates and radius.
stdcut =	(omitted, D=-1)	Threshold value of STD cut off.

The source type in the volume calculation is specified by s-type. In s-type=1, the source generates on a sphere of the center coordinates (x0,y0,z0) and the radius r0 with the inward direction. This is the same condition as dir=-all in s-type=9 of [source] section. In s-type=2, the source uniformly generates on surfaces of a rectangular solid which are defined by 6 planes, x=x0, x1, y=y0, y1, and z=z0, z1. Its direction is inward. In either case, you have to set the source region to be large so as to cover the all specified cells.

Attention!: the boundaries in the T-volume must be smaller than the boundaries of the geometry. There must not be any "outside" cells (-1) inside these boundaries.

When specifying stdcut, PHITS automatically stop the calculation depending on values of STD (standard deviation). This function is available when stdcut is positive and itall=0,1 is set in [parameters] section. When "all" relative values of STD of the tally result are larger than 0 and smaller than stdcut at the last of one batch. If stdcut in two (or more) tally sections is set, all the results of the tally sections have to satisfy the conditions in order to work the function.

When executing PHITS using [t-volume], the information on the source used in the volume calculation is output in [source] of the summary file file(6) (D=phits.out). [source] written in the input file of PHITS is not output in the summary file.

An example of [t-volume] is as follows.

Example 64: Example of [t-volume]

```
1:   [ T - V o l u m e ]
2:     mesh =  reg            # mesh type is region-wise
3:      reg = 101  102  103  104  105
4:     file = volume.out      # file name of output for [volume]
5:   s-type =    1            # 1: Sphere source, 2: Rectangular source
6:       x0 =  0.0            # (D=0.0) x of sphere center
7:       y0 =  0.0            # (D=0.0) y of sphere center
8:       z0 =  0.0            # (D=0.0) z of sphere center
9:       r0 =  50.0           # radius of sphere
```

[t-volume] section should have the setting of mesh=reg. The cell numbers should be specified after reg=. By setting s-type and the related parameters such as x0, you can define the special source region for the volume calculation. Note that the source region should be set to cover all cells specified in reg=, otherwise the calculated volumes would be wrong. On the contrary, if you set an extremely large source region, the statistical uncertainties of the calculated results would be large. The obtained volumes are outputted in a file named by file=. The format of the output file is as follows.

```
[ T - V o l u m e ] off
     mesh =  reg            # mesh type is region-wise
          ....   ....     ....   ....
          ....   ....     ....   ....

[ V o l u m e ]
  non      reg      vol      non
    1      101   5.0370E+02  0.2909
    2      102   2.4727E+03  0.1634
          ....   ....     ....   ....
          ....   ....     ....   ....

# Information for Restart Calculation
          ....   ....     ....   ....
          ....   ....     ....   ....
```

Because the volumes are outputted in the format of [volume] section, the results can be used in the main calculation with icntl=0 by infl command. Here, values of the last column in [volume] are statistical uncertainties (relative values). If the uncertainties are large, you can perform the restart calculation by adding istdev=-1 or -2 in [parameters] section, because the output file of [t-volume] has information for the restart mode.

7.19 [T-Userdefined] section

This tally is used for estimating and outputting physical quantities that cannot be calculated by the other tallies. To use this tally, the user must change "usrtally.f" and re-compile PHITS.

When [t-userdefined] is defined in an input file, PHITS calls the subroutine "usrtally" at each moment in the PHITS simulation, i.e., with the same timing used for calling the dumpall option. A subroutine for outputting all information is written in the default of "usrtally.f," and the user can set the output so that only required information is output by revising this file: please see the read-me file or sample input file in "\phits\utility\usrtally\" for more detail.

In [t-userdefined], several parameters can be used: file, for specifying a file name; nudtvar, for specifying the number of variables udtvar(*i*); udtvar(*i*)[133], for specifying a numerical value. These parameters can be used in the subroutine 'usrtally' without recompiling PHITS.

Table 7.63: [t-userdefined] parameters

name	value	explanation
file =	file name	Output file names used in usrtally subroutine. A maximum of 50 files can be used (the device numbers of these files are from 151 to 200).
nudtvar = udtvar(*i*) =	0(default) Numerical value (D=0)	The number of available udtvar(*i*). Parameters used in usrtally subroutine: these can be defined upto *i* =nudtvar.

Example 65: An example input for [t-userdefined]

```
 1:        file = output1.dat            <-its device number is iudtf(1)=151
 2:        file = output2.dat            <-its device number is iudtf(2)=152
 3:     nudtvar = 2                      <-nudtvar
 4:   udtvar(1) = 20.0                   <-udtvar(1)
 5:   udtvar(2) = -10.0                  <-udtvar(2)
```

The following parameters can be used in the subroutine usrtally.

(1) NCOL:
This is an intrinsic variable in the program and denotes identification of process.

NCOL
- 1 : start of calculation
- 2 : end of calculation
- 3 : end of a batch
- 4 : source
- 5 : detection of geometry error
- 6 : recovery of geometry error
- 7 : termination by geometry error
- 8 : termination by weight cut-off
- 9 : termination by time cut-off
- 10 : geometry boundary crossing
- 11 : termination by energy cut-off
- 12 : termination by escape or leakage
- 13 : (n,x) reaction
- 14 : (n,n'x) reaction
- 15 : sequential transport only for tally
- 16 : surface cross for WW of xyz mesh

[133] This parameter plays the same role as udtpara*i* ($i = 0 - 9$) before ver. 3.01. udtpara*i* can be also set after ver. 3.02.

(2) npe, me:
These are the number of used PEs (Processor Elements) and ID number of each processor, respectively, in the distributed-memory parallel computing.

(3) ipomp, npomp:
These are ID number of each core and the total number of used cores, respectively, in the shared memory parallel computing.

(4) iusrtally:
This is a parameter to control whether `subroutine usrtally` is used or not. If `[t-userdefined]` is defined in an input file, this parameter is set to be 1.

(5) iudtf(50):
These are device numbers of output files defined with `file=`. For example, if there is the earliest file defined in `[t-userdefined]`, its device number is `iudtf(1)=151`.

(6) nudtvar:
The number of available `udtvar(`*i*`)`. This is given as `nudtvar` in the input file.

(7) udtvar(*i*):
These are numerical values defined as `udtvar(`*i*`)` in the input file. `udtvar(`*i*`)` upto *i*=`nudtvar` can be used. If `udtvar(`*i*`)` is not defined in the input file, it is set to 0.

(8) NOCAS, NOBCH, RCASC, RSOUIN:

NOCAS	:	current history number in this batch
NOBCH	:	current batch number
RCASC	:	real number of NOCAS+maxcas*(NOBCH-1)
RSOUIN	:	sum of the weight of source particle

(9) NO, IDMN, ITYP, KTYP, JTYP, MTYP, RTYP, OLDWT:

NO	:	cascade id in this history
IDMN	:	material id
ITYP	:	particle type
KTYP	:	particle kf-code
JTYP	:	charge number of the particle
MTYP	:	baryon number of the particle
RTYP	:	rest mass of the particle (MeV)
OLDWT	:	weight of the particle at (x,y,z)

(10) QS:
This is dE/dx for electrons at (x,y,z).

(11) IBLZ1, IBLZ2, ILEV1, ILEV2:

IBLZ1	:	cell id at (x,y,z)
IBLZ2	:	cell id after crossing
ILEV1	:	level structure id of the cell at (x,y,z)
ILEV2	:	level structure id of the cell after crossing

(a) ILAT1:
This is a variable of level structure of cell.

(b) ILAT2:
This is a variable of level structure of cell.

(12) COSTH, UANG(1), UANG(2), UANG(3), NSURF:

COSTH	:	cosine of an angle of incidence in a surface crossing
UANG(1,2,3)	:	x,y,z component of a normal vector of its surface, respectively
NSURF	:	internal number of the surface (This is different from the surface number defined in the `[surface]` section.)

(13) NAME, NCNT(1), NCNT(2), NCNT(3):

NAME	:	collision number of the particle
NCNT(1,2,3)	:	values of counter 1, 2, and 3

(14) WT, U, V, W:

WT	:	wight of the particle at (xc,yc,zc)
U, V, W	:	unit vector of momentum of the particle

(15) E, T, X, Y, Z:

E	:	energy of the particle at (x,y,z) (MeV)
T	:	time of the particle at (x,y,z) (nsec)
X, Y, Z	:	position coordinate of the preceding event point (cm)

(16) EC, TC, XC, YC, ZC:

EC	:	energy of the particle at (xc,yc,zc) (MeV)
TC	:	time of the particle at (xc,yc,zc) (nsec)
XC, YC, ZC	:	position coordinate of the particle (cm)

(17) SPX, SPY, SPZ:

SPX, SPY, SPZ	:	unit vector of spin direction of the particle

(18) NZST:
This is charge state of the particle.

(19) NCLSTS:
This variable means the number of produced particle and nucleus.

(a) MATHZ, MATHN, JCOLL, KCOLL:

MATHZ	:	Z number of the mother nucleus
MATHN	:	N number of the mother nucleus
JCOLL	:	reaction type id1
KCOLL	:	reaction type id2

JCOLL and KCOLL indicate the following meaning.

JCOLL

0	:	nothing happen
1	:	Hydrogen collisions
2	:	Particle Decays
3	:	Elastic collisions
4	:	High Energy Nuclear collisions
5	:	Heavy Ion reactions
6	:	Neutron reactions by data
7	:	Photon reactions by data
8	:	Electron reactions by data
9	:	Proton reactions by data
10	:	Neutron event mode
11	:	delta ray production
13	:	Photon reactions by EGS5
14	:	Electron reactions by EGS5

KCOLL

0	:	normal
1	:	high energy fission
2	:	high energy absorption
3	:	low energy n elastic
4	:	low energy n non-elastic
5	:	low energy n fission
6	:	low energy n absorption

(b) ICLUSTS, JCLUSTS, QCLUSTS, JCOUNT:
These variables have a array and denote the information on the produced particle and nucleus.

ICLUSTS kind of particle

0	:	nucleus
1	:	proton
2	:	neutron
3	:	pion
4	:	photon
5	:	kaon
6	:	muon
7	:	others

JCLUSTS(i)

i = 0	:	angular momentum
= 1	:	proton number
= 2	:	neutron number
= 3	:	ityp
= 4	:	status of the particle 0: real, <0 : dead
= 5	:	charge number
= 6	:	baryon number
= 7	:	kf code

QCLUSTS(i)

i = 0	:	impact parameter
= 1	:	x-component of unit vector of momentum
= 2	:	y-component of unit vector of momentum
= 3	:	z-component of unit vector of momentum
= 4	:	$e_{tot} = \sqrt{p^2 + m^2}$ (GeV)
= 5	:	rest mass (GeV)
= 6	:	excitation energy (MeV)
= 7	:	kinetic energy (MeV)
= 8	:	weight
= 9	:	time (nsec)
= 10	:	x coordinate (cm)
= 11	:	y coordinate (cm)
= 12	:	z coordinate (cm)

7.20 [T-Gshow] section

This tally produces graphical geometry output for regions bounded by `xyz` mesh. These outputs can be obtained without transport calculation using the option `icntl=7` in the `[parameters]` section.

The option of `output=10` (`gshow=5` for other tallies) is useful for lattice and tetrahedral geometry with very fine mesh.

Table 7.64: `[t-gshow]` parameters

name	value	explanation
`mesh =`	`xyz`	Mesh type: only `xyz` mesh; a mesh type subsection is required below this option.
`axis =`	`xy, yz, xz`	2-dimensional.
`file =`	`file name`	Define file names. This is required by each setting of `axis`.

`output =`	`1`
	`2`
	`3`
	`4`
	`5`
	`6`
	`7`
	`8`
	`9`
	`10`
		Region boundary
		Region boundary + material color
		Region boundary + material name
		Region boundary + material color + material name
		Region boundary + region name
		Region boundary + material color + region name
		Region boundary + LAT number
		Region boundary + material color + LAT number
		No output
		Material color in pixel style

`resol =` `width =`	`1` (default) `0.5` (default)	This option multiplies the region line resolution by a factor of `resol`. The option defines the line thickness.
`title =`	(optional)	Title.
`angel =`	(optional)	ANGEL parameters.
`sangel =`	(optional)	Special format for ANGEL parameters.
`x-txt =` `y-txt =`	(optional) (optional)	*x* axis title. *y* axis title.
`epsout =`	`0` (default), `1`	Results are plotted into eps files. This eps file is named by replacing the extension into ".eps ".
`bmpout =`	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When `mesh=xyz, axis` and `axis=xy, yz, xz`, it is available.
`vtkout =`	0(default), 1	Output the tally results in the `xyz`-mesh in the input format of ParaView. This file is named by replacing the extension with ".vtk." When `mesh=xyz, axis` and `axis=xy, yz, xz`, it is available.
`vtkfmt =`	0(default), 1	Format of output file for ParaView. 0: ascii, 1: binary.
`trcl =`	(optional)	Coordinate transformation number or definition for `r-z` or `xyz` mesh.
`gslat =`	1(default), 0	1: show lattice boundary in `gshow`, 0: do not show.

`output=7,8` can be used only when the cells in bottom level are themselves in the lattice; the output then gives lattice numbers in the format `(4,1,2)`. For example, Fig. 5.41 in Sec. 5.6.4 is generated by the input shown below.

Example 66: `[t-gshow]` example

```
 1:   [ T - gshow ]
 2:        mesh = xyz
 3:      x-type = 2
 4:          nx = 100
 5:        xmin = -10.
 6:        xmax =  10
 7:      y-type = 1
 8:          ny = 1
 9:               -5.0 5.0
10:      z-type = 2
11:       nz    = 100
12:       zmin  = -10.
13:       zmax  =  10.
14:        axis = xz
15:      output = 8
16:        file = cell-example6.dat
17:      epsout = 1
```

7.21 [T-Rshow] section

This tally produces graphical geometry output for regions bounded by xyz mesh with color plotting in proportion to physical quantities of the regions. Usually, this tally uses results obtained by a PHITS calculation with reg mesh as input data to display the distribution of the physical quantity. These outputs can be obtained without transport calculation using the option icntl=9 in the [parameters] section.

A scale of the color gradation can be changed by setting ANGEL parameters zlog and zlin to log and linear scales, respectively. Default setting is zlin.

Table 7.65: [t-rshow] parameters

name	value	explanation
mesh =	xyz	Mesh type: only xyz mesh; a mesh type subsection is required below this option.
axis =	xy, yz, xz	2-dimensional.
file =	file name	Define file names. This is required by each setting of axis.
output =	1 2 3 4	Region boundary Region boundary + material name Region boundary + region name Region boundary + LAT number
resol = width =	1 (default) 0.5 (default)	This option multiplies the region line resolution by a factor of resol. The option defines the line thickness.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
x-txt = y-txt = z-txt =	(optional) (optional) (optional)	*x* axis title. *y* axis title. *z* axis title.
reg = value reg val		Region definition. Value definition with the same format as volume definition. For details see Sec. 6.1.2.
iechrl =	72 (default)	Maximum number of columns for volume input echo.
epsout =	0 (default), 1	Results are plotted into eps files. This eps file is named by replacing the extension into ".eps ".
bmpout =	0(default), 1	Generate Bitmap figure of 2-dimensional tally output. This file is named by replacing the extension with ".bmp." When mesh=xyz, axis and axis=xy, yz, xz, it is available.
trcl =	(optional)	Coordinate transformation number or definition for r-z or xyz mesh.
gslat =	1(default), 0	1: show lattice boundary in gshow, 0: do not show.

For example, Figure 7.9 can be obtained by the `[t-rshow]` tally shown below from the example (6) shown in Sec. 5.6.4.

Example 67: `[t-rshow]` example

```
[ T - rshow ]
     mesh = xyz
   x-type = 2
       nx = 100
     xmin = -10.
     xmax =  10.
   y-type = 1
       ny = 1
            -5.0  5.0
   z-type = 2
       nz = 100
     zmin = -10.
     zmax =  10.
     axis = xz
   output = 1
     file = cell-example6-rshow.dat
   epsout = 1
      reg = (201<101[-1 1 0]<1) (201<101[0 1 0]<1) (201<101[1 1 0]<1)
            (201<101[-1 0 0]<1) (201<101[0 0 0]<1) (201<101[1 0 0]<1)
            (201<101[-1 -1 0]<1) (201<101[0 -1 0]<1) (201<101[1 -1 0]<1)
    value
   non     reg      val     # reg definition
    1  1000001   1.0000E+00 # ( 201 < 101[ -1 1 0 ] < 1 )
    2  1000002   2.0000E+00 # ( 201 < 101[ 0 1 0 ] < 1 )
    3  1000003   3.0000E+00 # ( 201 < 101[ 1 1 0 ] < 1 )
    4  1000004   4.0000E+00 # ( 201 < 101[ -1 0 0 ] < 1 )
    5  1000005   5.0000E+00 # ( 201 < 101[ 0 0 0 ] < 1 )
    6  1000006   6.0000E+00 # ( 201 < 101[ 1 0 0 ] < 1 )
    7  1000007   7.0000E+00 # ( 201 < 101[ -1 -1 0 ] < 1 )
    8  1000008   8.0000E+00 # ( 201 < 101[ 0 -1 0 ] < 1 )
    9  1000009   9.0000E+00 # ( 201 < 101[ 1 -1 0 ] < 1 )
```

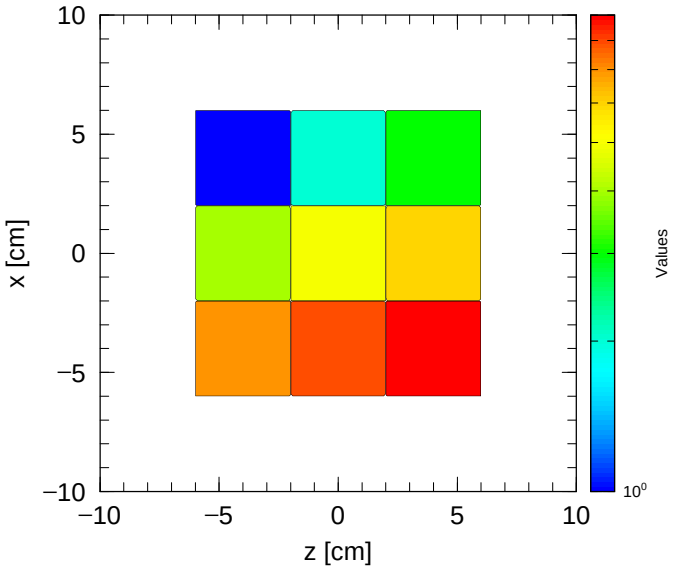


Figure 7.9: Example of `[t-rshow]`.

7.22 [T-3Dshow] section

This tally produces graphical geometry output by 3-dimensional view. This tally can be generated without transport calculation using the option `icntl=11` in the `[parameters]` section.

Table 7.66: `[t-3dshow]` parameters (1)

name	value	explanation

`output =`	`0`
	`1`
	`2`
	`3` (default)
		Draft
		Only region boundary
		Without region boundary
		Region boundary + color

`material =` (next line)	(optional) `all`, number of materials `2 5 8`	Specify materials for display. `all` : default (same as no definition). To set number of materials, define the material numbers in the next line. The number of materials can be set as a negative value: in this case the specified materials are not included for display. Material numbers.
`x0, y0, z0 =`	(D=0.0)	Coordinates of focal point. Center of screen is defined by this point and the eye point.
`e-the =`	(D=80)	Polar angle of coordinates of the eye point, θ, from the focal point [degree].
`e-phi =`	(D=140)	Azimuthal angle of coordinates of the eye point, ϕ, from the focal point [degree].
`e-dst =`	(D=w-dst*10)	Distance between the eye point and the focal point [cm].
`l-the =`	(D=e-the)	Polar angle of coordinates of the light source, θ, from the focal point [degree].
`l-phi =`	(D=e-phi)	Azimuthal angle of coordinates of the light source, ϕ, from the focal point [degree].
`l-dst =`	(D=e-dst)	Distance between the light source and the focal point [cm].
`w-wdt =`	(D=100)	Width of screen frame [cm].
`w-hgt =`	(D=100)	Height of screen frame [cm].
`w-dst =`	(D=200)	Screen frame distance from the focal point [cm]. A straight line drawn between the center of the screen frame and the focal point crosses the screen surface vertically, and passes through the eye point.
`w-mnw =`	(D=100)	Number of meshes in horizontal direction.
`w-mnh =`	(D=100)	Number of meshes in vertical direction.
`w-ang =`	(D=0.0)	Angle of frame [degrees].
`heaven =`	(D=y)	Topside direction; set `x, -x, y, -y, z, -z`.
`mirror =`	(D=0)	=-1; Mirror transformation in horizontal direction.

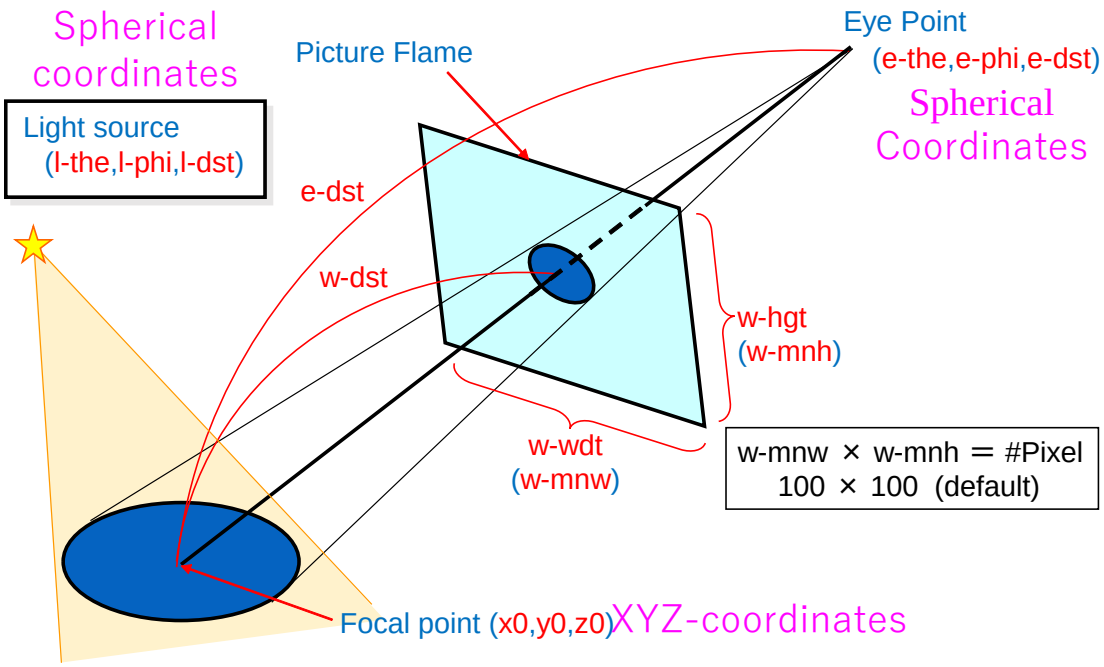


Figure 7.10: 3dshow tally: focal point (`x0,y0,z0`), eye point (`e-the,e-phi,e-dst`), light source (`l-the,l-phi,l-dst`), and picture flame (`w-wdt,w-hgt,w-dst`).

Table 7.67: [t-3dshow] parameters (2)

name	value	explanation
line =	0 (default), 1	When output=1, 3, 0: material boundary + surface boundary; 1: material boundary + surface boundary + region boundary.
r-out =	(D=50000)	Radius of outer void including eye point, and light source [cm].
shadow =	(D=0)	Shadow level (0:no shadow, 2 is recommended).
bright =	(D=0.8)	Brightness limit (1:max, 0:no brightness).
dark =	(D=0.2)	Darkness limit (1:no darkness, 0:max).
box =	(D=0)	Number of penetrations per box; maximum=5.
box	10 numbers	Box definition (see below).
matinbox =	(optional) all, number of materials	Materials in the box for display. all: default (same as no definition); If number of materials is set, material numbers should be defined in the next line (number of materials cannot be set as negative).
(next line)	2 5 8	Material numbers.
reginbox =	(optional) all, region numbers	Regions in the box for display. all: default (same as no definition); If the matinbox is defined for a region, the region is not displayed.
resol =	1 (default)	This option multiplies the region line resolution by a factor of resol.
width =	0.5 (default)	The option defines the line thickness.
file =	file name	Define file names.
title =	(optional)	Title.
angel =	(optional)	ANGEL parameters.
sangel =	(optional)	Special format for ANGEL parameters.
x-txt =	(optional)	*x* axis title.
y-txt =	(optional)	*y* axis title.
z-txt =	(optional)	*z* axis title.
epsout =	0 (default), 1	Results are plotted into eps files. This eps file is named by replacing the extension into ".eps ".
axishow =	(D=1) 0, 1, 2	0: No. Axis not shown. 1: Small axis is shown in the lower-left of the figure. 2: Large axis is shown in the center of the figure.

The definitions of the rules for reg= and reginbox= are the same as that for the region mesh in Sec. 6.1.1.

To saving calculation time, an outer void defined by the radius r-out has been introduced. A larger r-out value must be used when using a large geometry or when placing the light source and eye point at long distances. As this new outer void definition can be seen in input echo, input echo cannot be used by icntl=11 as an input for the next calculation.

No shadow is created if the eye point and light source are set in the same position.

7.22.1 Box definition

A maximum of five penetration boxes can be defined; defined boxes become transparent. To define a `box`, three points must first be set as $\mathbf{b}_0(x_0,y_0,z_0)$, $\mathbf{b}_1(x_1,y_1,z_1)$, and $\mathbf{b}_2(x_2,y_2,z_2)$, respectively. The fourth point $\mathbf{b}_3$ from $\mathbf{b}_0$ is defined to lie L cm in the vertical direction of the plane defined by these three points, i.e., in the ($\mathbf{b}_2$ - $\mathbf{b}_0$) direction. In the `box` definition, a coordinate transformation such as `trcl=` transform number or `trcl = (........)` can be used prior to the definition of the points.

This function may fail when a void region is included in the penetration box; in this case, the user should fill the void region with a material of very low density, e.g., air.

The box definition is shown below. Each relation is also shown in Fig. 7.11.

```
box = 2
box   x0  y0  z0
      x1  y1  z1
      x2  y2  z2  L
box   trcl = 2
      x0  y0  z0
      x1  y1  z1
      x2  y2  z2  L
box   *trcl = (0 0 0  0 90 90  90 60 150  90 30 60  -1)
      0.0  0.0  0.0
     -5.0  0.0  0.0
      0.0  0.0  5.0   5.0
```

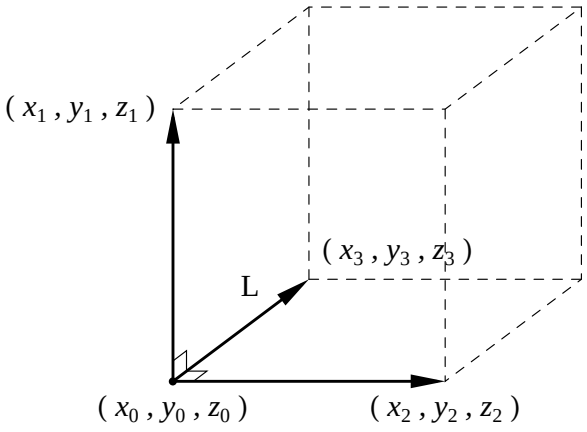


Figure 7.11: Example of box definition.

7.22.2 3dshow example

Example 68: [t-3dshow] example (1)

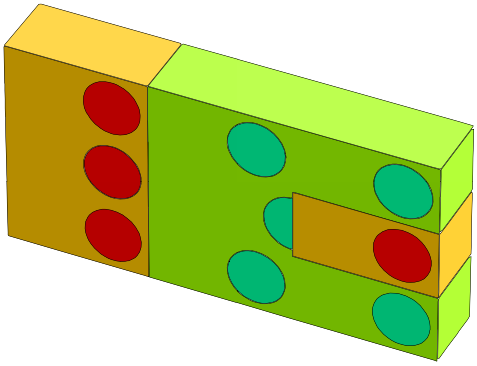
```
 1:  [cell]
 2:      1  0 -1 fill=1
 3:      2  0 -41 42 -43 44 -45 46  u=1 fill=5
 4:     22  0 -41 42 -43 44 -45 46  u=1 trcl=(0 0 20) fill=6
 5:     23  like 22 but trcl=(0 0 40) fill = 7
 6:      5  0 -21 22 -23 24 -25 26  u=5 lat=1 fill=3
 7:      6  0 -21 22 -23 24 -25 26  u=6 lat=1 fill= -1:1 0:0 0:0  2 2(0 0 5) 2
 8:      7  0 -21 22 -23 24 -25 26  u=7  fill= -1:1 0:0 0:0  2 3 2 lat=1
 9:      3  1  3.97300E-02  3 u=2
10:      4  6  4.18280E-02 -3 u=2
11:     13  5  8.47130E-04 -3 u=3
12:     14  3  1.23620E-01  3 u=3
13:     8  -1 +1
14:  [surface]
15:      1   rpp -15 15 -5 5  -5 55
16:      21   px    5
17:      22   px   -5
18:      23   py    5
19:      24   py   -5
20:      25   pz   15
21:      26   pz   -5
22:      41   px   15
23:      42   px  -15
24:      43   py    5
25:      44   py   -5
26:      45   pz   15
27:      46   pz   -5
28:      5   rpp -20 20 -5 5 -5 35
29:      6   rpp -20 20 -5 5 -5 15
30:      7   rpp -20 20 -5 5 35 55
31:      3   c/y   0 10  4
```

In the above geometry, the overall body is a rectangular solid with rectangular solid lattices including cylinders on the interior. A graphical plot for the geometry can be created using 3dshow as follows.

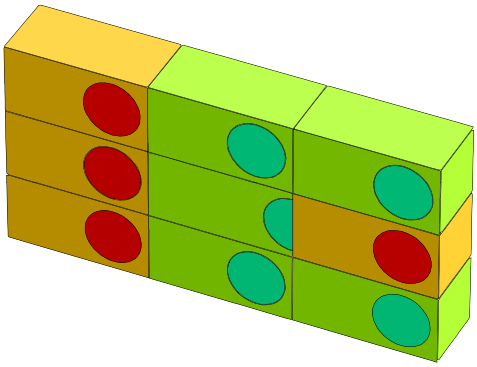
Example 69: [t-3dshow] example (2)

```
 1:  [t-3dshow]
 2:     output = 3
 3:     heaven = x
 4:     resol = 2
 5:     width = 0.1
 6:     x0 =  0
 7:     y0 =  0
 8:     z0 = 25
 9:     e-the = 70
10:     e-phi = 50
11:     e-dst = 1000
12:     l-the = 50
13:     l-phi = 25
14:     l-dst = 2000
15:     w-wdt = 60
16:     w-hgt = 40
17:     w-dst = 150
18:     file = dshow.dat
```

The output result is then



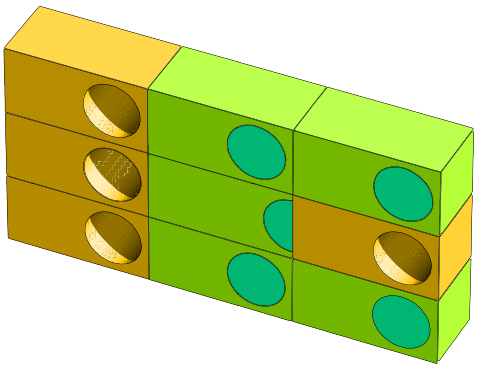
A region boundary can be added using option `line=1` to produce the following `box`, which shows how the lattices are set up:



Setting material number 5 to be transparent and adding shadows using

```
material = -1
            5
  shadow =  2
```

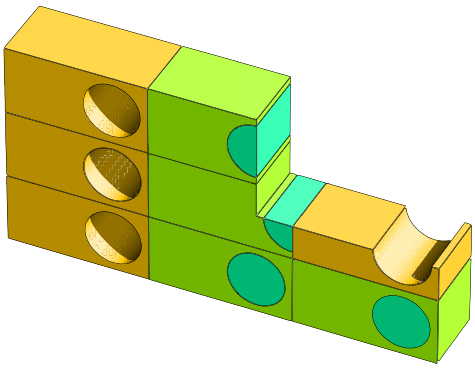
produces the following box:



Next, defining the following box:

```
box = 1
box    0  10  30
     100  10  30
       0  10  100 100
```

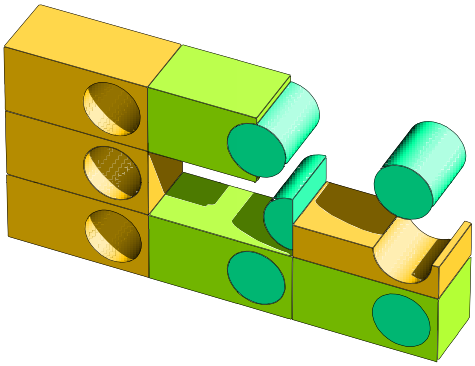
produces a transparent box that shows the inside of its body:



As a final example, by adding

```
reg = ( 3 < 6[0 0 0] )
matinbox = 1
           6
```

the regions defined by `reg = (3 < 6[0 0 0] )` become transparent and material number 6 becomes visible:



Any number of complex structure types can be created by combining the above options.

7.23 [T-4Dtrack] section

This tally outputs particle track information (position and time). By loading the output file into the `Particle tracks` tab of PHIG-3D, the trajectories can be visualized on the geometry.

Table 7.68: [T-4Dtrack] parameters (1)

name	value	explanation
`file`	file name	Output file name
`reg`	`all` (default)	Specify the region number for writing out tracks. Refer to Section 5.39 for the format.
`part`	`all` (default)	Specify the particles for which to write out tracks. If not specified, tracks for all particles will be written out. If `part` includes `nucleus`, unstable nuclei will also be included in the output. However, only nuclei with half-lives between `HalfLifeMin` and `HalfLifeMax` will be considered. Refer to Section 6.7.1 for the format.
`stepminEM`	= `0.0` (default)	Minimum step distance (cm) for electrons, positrons, and photons. Adjusting this value can reduce the data size significantly without compromising the quality of the visualizing using PHIG-3D. See the following (1) for details.
`SkipRate`	= `0.0` (default)	Specify the ratio for skipping the output of particle track endpoints. This setting applies to all particles. Set a value between 0 and 1. For example, if `SkipRate=0.9` is specified, the output of endpoints will be skipped with a 90% probability. The size of the output file will be approximately (1 - `SkipRate`) times the original size. For exceptions to skipping, see (2) below.
`SkipRate-of(kf-code)`	= `0.0` (default)	Apply `SkipRate` only to particles specified by kf-code. For example, if `SkipRate-of(2112)=0.9` is specified, the output of endpoints for neutrons will be skipped with a 90% probability. If both `SkipRate` and `SkipRate-of` are set simultaneously, the value of `SkipRate-of` takes precedence.

(1) A particle track is essentially composed of continuous steps. The mean free paths of electrons, positrons, and photons are generally very small, so writing out every step would result in a very large data size. Therefore, to save data size, steps whose distance from the starting point of a step is less than `stepminEM` are not written out.

(2) The output of track points is not skipped if any of the following three conditions are met: 1. The particle reaches the energy cutoff. 2. The particle reaches the boundary of a region. 3. An interaction occurs, and the number of final states is 2 or more.

Table 7.69: [T-4Dtrack] parameters (2)

name	value	explanation
HistoryMax	None (default)	The maximum number of events to write out. If omitted, all events are written out.
HistoryFile	file name	Specifies the file name containing the event numbers to be written out. If not specified, all events are written out. File structure: Write the NOCAS (event number) and NOBCH (batch number) of the desired event on each line (Example 1). The numbers can be read in real number format, so dump data files specifying NOCAS and NOBCH can also be loaded.
format	= t4d (default) = pict	Specify the output format. Output in t4d format. For details, refer to "/phits/utility/t-4dtrack/T4D-format/readme-en.pptx". Output in pict format used by EGS5.
verbose	= 0 = 1 (default) = 2	Specify the output level. For details, refer to "/phits/utility/t-4Dtrack/format/readme-en.pptx". Output the minimum necessary information for track display in PHIG-3D. Also output track cell information and other details. For examples of how to use cell information, see "/phits/utility/t-4Dtrack/OriginTrackFinder/readme-en.pptx". Output counter information and more. When outputting counter values, it is necessary to use ctmin or ctmax in a separate tally.
precision	= 6 (default)	Number of digits for outputting real numbers.

An example of the input for the file specified by HistoryFile:

```
# NOCAS  NOBCH
      1      1
      2      1
      3      1
      1      2
      2      2
      3      2
```

Table 7.70: [T-4Dtrack] parameters (3)

name	value	explanation
PinnedOption	= 0 (default) = 1 = 2	Choose whether to keep (pin) a specified particle in the visualization after it stops due to falling below the energy cutoff, instead of immediately removing it. The duration for which it remains can be specified. Do not pin. Pin generated nuclei. The target nuclei can be specified by their half-life. Pin particles specified in PinnedParticles.
PinnedTimeOption	= 0 (default) = 1 = 2	Specify the type of duration for pinning. Fixed duration. Specified by PinnedTimeFix. aHalf × (the half-life of an unstable nucleus) max(0, aHalf×log10(half-life/s)+bHalf) (ns)
PinnedTimeFix	= 10.0 (default)	Fixed pinning duration (ns). Used with PinnedTimeOption = 0.
aHalf	= 1.0 (default)	Scale parameter for half-life. Used with PinnedTimeOption = 1,2.
bHalf	= 0.0 (default)	Scale parameter for the logarithmic term of the half-life. Used with PinnedTimeOption = 2.
HalfLifeMin	= 1.0e-22 s (default)	Minimum half-life for targeted unstable nuclei. Used with part = nucleus, PinnedTimeOption = 1,2. Enter a number followed by at least one space, then enter the unit of time. Available units are s, min, h, d, y, representing seconds, minutes, hours, days, and years, respectively.
HalfLifeMax	= 3.0e+32 s (default)	Maximum half-life for targeted unstable nuclei. Used with part = nucleus, PinnedTimeOption = 1,2. The method of entering the value is the same as for HalfLifeMin.
PinnedParticles	None (default)	Enter the names or kf codes of the particles to be pinned. If entering multiple, separate them with spaces.

8 Processing dump file

You can output the information on transport particles on dump file by [t-cross], [t-time], [t-product] tallies in either ascii or binary format. This "dump-a.f" program can convert the dump data from ascii to binary (or vice-versa). It can also divide (or combine) dump files into several files. The latter function is useful when the dump file is employed as the source data of the PHITS simulation using the MPI parallelization with different numbers of cores. In addition, you can get the information that cannot be obtained by the conventional tally functions in PHITS by analyzing the dump file. For that purpose, you need to make your own program in "dump-a.f", and compile it by yourself. The followings are the instruction how to use this program.

Executable files for this program are "dump-a_win.exe", "dump-a_mac.exe", and "dump-a_lin.exe" for Windows, Mac, and Linux, respectively. After running the executable file, you have to supply either input file name or the variable of each parameter in the terminal. The meaning of each input parameter is given below.

- 1st card (iasb): The format of dump file (0: binary, 1: ascii)
- 2nd card (ipur): Purpose (0: convert format, 1: divide/combine, 2: user program)
- 3rd card (chin): Input dump data file name (comments should NOT be written in this line)
- 4th card (isdmp(0)): Number of data in a record
- 5th card (isdmp(1-isdmp(0))): ID number of each record (see also Table 5.58 and 5.59)
- 6th card (chot): Output dump data file name (comments should NOT be written in this line)
- 7th card (irec): Number of records to be read (0: all data)
- 8th card (ninp): Number of input dump files (only for divide/combine mode, 0 for single file)
- 9th card (nout): Number of output dump files (only for divide/combine mode, 0 for single file)

The characters in the parentheses are the variable name in the program.

Sample input files for converting the format ("dump-convert.inp") and divide/combine the files ("dump-divcom.inp") are in "phits/utiliity/dump-a" folder. Note that the information on the dump particles generated by the same history is written in the same file in the divide/combine mode in order to keep the correlation of each particle.

9 Region error check

In developing a complex geometry, it is diffcult to define the geometry without mistakes such as double-defined or undefined regions. Correspondingly, a function for automatically detecting double-defined or undefined regions was implemented after version 2.67. This geometry check function works when specifying a tally for generating the two-dimensional view of a geometry: such tallies include `[t-gshow]`, `[t-rshow]`, and other tallies involving setting `axis=xy,yz,xz` or the `gshow` (`icntl=8`) or `rshow` (`icntl=10`) options.

Figure 9.1 shows an example of a two-dimensional view with geometry errors. Double-defined regions are painted in black, while undefined regions are in purple. When an undefined region is detected, its surrounding regions may disappear from the figure.

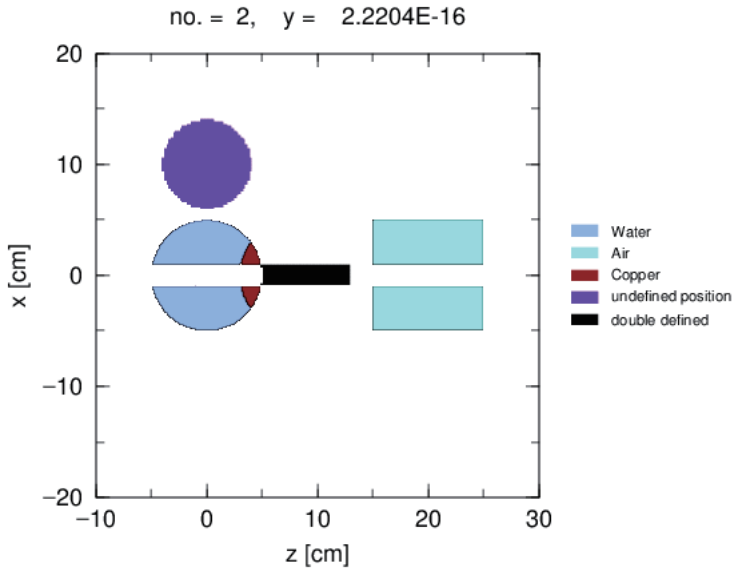


Figure 9.1: Output including geometry errors.

When PHITS detects a geometry error, a geometry error file named "***.err" is output (*** indicates the original tally output file name without the extension). In this file, the (x, y, z) coordinates together with the overlapping cell numbers are written as follows:

```
Errors of cell definition in EPS Page No. =   1
Overlapped Cell IDs  x, y, z  coodinates
(Cells 0       0  indicate undefined region)
     100     102    -4.847761E+00  1.234568E-11 -1.211940E+00
       0       0    -4.241791E+00 -2.500000E+00 -8.079602E-01
```

The first line indicates that cell numbers 100 and 102 overlap at the point `x=-4.847761E+00, y=1.234568E-11, z=-1.211940E+00`. The second line indicates that an undefined region is detected at the point `x=-4.241791E+00, y=-2.500000E+00, z=-8.079602E-01`.

Geometry errors can easily be found using this geometry check function. Note that the function can detect geometry error only when an error occurs on the grid points of the `xyz` mesh of the tally, while geometry errors outside the tally region cannot be detected. Even in the tally region, a small error region might also remain undetected if the region does not contain a grid point.

10 Compilation of PHITS

PHITS must be re-compiled by yourself for some circumstances; e.g. you use a function of user definition source5.3.18 or user-defined tally7.19. Our recommended Fortran compiler is Intel Fortran 11.1 (or later). [134]

Table 10.1 summarizes the combination of the Fortran compiler and OS by which we have succeeded in compiling PHITS for four different parallel options: single mode, MPI, OpenMP, and hybrid of MPI and OpenMP.

Table 10.1: The available fortran compilers for each OS. ∗ OpenMP acceleration performance is not so high when using gfortran.

OS	Intel Fortran				gfortran			
	single	MPI	Open MP	hybrid	single	MPI	Open MP	hybrid
Windows	✓	✓	✓	✓	✓	✓		
Mac	✓		✓		✓		✓∗	
Linux	✓	✓	✓	✓	✓	✓		

10.1 Compilation on Windows OS

10.1.1 Compilation using Microsoft Visual Studio with Intel Fortran

The "\phits\bin\" folder contains a solution file ("phits-intel.sln") required for compiling PHITS using Microsoft Visual Studio coupled with Intel Fortran. You can use this software free of charge. Please refer to the following document for installation instructions: "phits/document/Install-IntelFortran-OneAPI-en.pdf". If you are using an older version of Intel Fortran (ifort), please double-click "phits-ifort.vfproj" and select the "Active (Release)" mode to compile PHITS. This procedure is the same as the one used prior to version 3.35, which used "phits-intel.vfproj."

10.1.2 Compilation using makefile

The PHITS code can be compiled using the "make" command. However, this method is no longer supported owing to the poor compiler availability. For this purpose, the "makefile" file in "\src" folder should be revised to fit for the host computer. For example, to compile using the gfortran compiler on Windows OS, the `ENVFLAGS` written in "makefile" should be set to `WinGfort`. To execute PHITS via parallel computing using MPI or OpenMP, it is necessary to set `USEMPI` or `USEOMP`, respectively, to `true`. The name of the executable file is given as "phits_XXX.exe," where XXX is a parameter set to `ENVFLAGS`. For example, when `ENVFLAGS=WinGfort`, the name is phits_WinGfort.exe.

The compiler options given in "makefile" are simply examples, and users may need to change the setting to fit their own computer environment.

The attached "makefile" can be used for GNU make. If an error is encountered with the "make" command, try the "gmake" command instead.

To use gfortran for Windows, the latest installer version can be downloaded from "Downlaod Latest Version" on the following web site:

- MinGW-w64 - for 32 and 64 bit Windows
  (`https://sourceforge.net/projects/mingw-w64/files/?source=navbar`)

Running the downloaded installer, select "x86_64" or "i686" for 64-bit or 32 bit PC, respectively, as "Architecture." It is unnecessary to change other parameters during the installation process.

The followings are the procedure of how to compile PHITS with the gfortran compiler.

(1) Revise "makefile" in "\phits\src\" folder: set `ENVFLAGS=WinGfort`. Note that OpenMP cannot be activated.

[134] Intel fortran of OneAPI is available for free of charge as explained in "phits/document/Install-IntelFortran-OneAPI-en.pdf".

(2) Open a command prompt by clicking "mingw-w64.bat" in the gfortran installation folder.

(3) Use the "cd" command and go to "\phits\src\" folder. Type "mingw32-make" to compile PHITS.

The compiled PHITS by gfortran can be used as follows:

(1) Copy "phits_WinGfort.exe" to "\phits\bin\."

(2) Open "phits.bat" in "\phits\bin\", and add

```
set PATH=C:\Program Files\mingw-64\...
```

written in the 2nd line of "mingw-w64.bat," . Then set "PHITS_EXE" to the PHITS executable file compiled by gfortran, i.e., "phits_WinGfort.exe."

(3) Run PHITS via Windows explorer using "sendto → PHITS" command.

(4) After finishing the PHITS calculation, some warnings, such as "Note: The following floating-point exceptions are signaling: IEEE_DENORMAL." , may show up, but can be be ignored.

10.2 Compilation on Mac OS

The PHITS code can be compiled using the "make" command. For this purpose, the "makefile" file in "\src" folder should be revised to fit for the host computer. For example, to compile using Intel Fortran Compiler on Mac[135], the `ENVFLAGS` written in "makefile" should be set to `MacIfort`. To execute PHITS via parallel computing using MPI or OpenMP, it is necessary to set `USEMPI` or `USEOMP`, respectively, to `true`. The name of the executable file is given as "phits_XXX.exe," where XXX is a parameter set to `ENVFLAGS`. For example, when `ENVFLAGS=MacIfort`, the name is phits_MacIfort.exe. If `USEOMP` is also set to `true`, the name would become phits_MacIfort_OMP.exe. The compiler options given in "makefile" are simply examples, and users may need to change the setting to fit their own computer environment.

The attached "makefile" can be used for GNU make. If an error is encountered with the "make" command, try the "gmake" command instead.

Use ifort if your machine has Intel OneAPI. For use of ifort, set `ENVFLAGS=MacIfort` in the "makefile." Use gfortran if OneAPI is not available. The Homebrew package manager is encouraged to install gfortran. Execute following commands on the terminal if your machine does not have Homebrew.

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Execute this command on the terminal after installation of Homebrew.

```
brew install gcc
```

That ' s it. To compile the PHITS code with gfortran, set `ENVFLAGS=MacGfort` in the "makefile."

10.3 Compilation on Linux OS

The PHITS code can be compiled using the "make" command. For this purpose, the "makefile" file in "\src" folder should be revised to fit for the host computer. For example, to compile using Intel Fortran on Linux[136], the `ENVFLAGS` written in "makefile" should be set to `LinIFX`. To execute PHITS via parallel computing using MPI or OpenMP, it is necessary to set `USEMPI` or `USEOMP`, respectively, to `true`. The name of the executable file is given as "phits_XXX.exe," where XXX is a parameter set to `ENVFLAGS`. For example, when `ENVFLAGS=LinIFX`, the name is phits_LinIFX.exe. If `USEOMP` is also set to `true`, the name would become phits_LinIFX_OMP.exe. The compiler options given in "makefile" are simply examples, and users may need to change the setting to fit their own computer environment.

The attached "makefile" can be used for GNU make. If an error is encountered with the "make" command, try the "gmake" command instead.

[135] Note that Intel compiler for Mac are not distributed after 2025.

[136] Please refer to the following document for installation instructions: "phits/document/Install-IntelFortran-OneAPI-en.pdf".

10.4 Compilation with KURBUC

The actual source code of proton and ion track structure simulation module, “kurbuc.f,” is not included in src folder with respect to its copyright. Thus, “kurbuc.f” included in src folder is a dummy file, and executable file of PHITS created by the normal compilation process cannot perform the proton and ion track structure simulation.

If you want to perform proton and ion track structure simulation using your own PHITS executable file, you must replace the object file of kurbuc (“kurbuc.o” or “kurbuc.obj”) by that corresponding to your machine environment included in “src/kurbuc-obj” folder, and re-create the PHITS executable file. The easiest way to re-create PHITS executable file is to save a source file of PHITS, e.g. “main.f,” and compile PHITS again. If you cannot find an object file suitable to your machine environment or have problem with re-creating PHITS executable file using our provided object file, please contact to PHITS office.

10.5 Array sizes

The array sizes described in the "param.inc" file may need to be checked and modified. The important variables you may need to change in "param.inc" are `kvlmax` (maximum number of cells and materials), `itlmax` (maximum number of tally entries), and `isrc` (maximum number of multi-sources).

The default param.inc is shown below.

File 2: param.inc

```
************************************************************************
*                                                                      *
*     'param.inc'                                                      *
*                                                                      *
************************************************************************
!     when you increase mdas over 250000000, you have to define integer*8 instead of integer in the next line
      integer mdas,mcmx,mci,mmdas,mmmax,nbnds,mct,multmax,pnlmax
      integer kvlmax, kvmmax, itlmax, inevt, isrc, nbchmax, maxchapara
      integer mxprodxs
      real*8  das
      parameter ( mdas   = 80000000 )
      parameter ( kvlmax =     3000 )
      parameter ( kvmmax =  1000000 )
      parameter ( itlmax =      200 )
      parameter ( inevt  =       71 )
      parameter ( isrc   =     3000 )
      parameter ( nbchmax=    10000 )
      parameter ( multmax=      500 )
      parameter ( pnlmax =     3000 )
      parameter ( maxchapara = 1000 )
      parameter ( mxprodxs =  20000 )

*----------------------------------------------------------------------*
*                                                                      *
*       pnlmax : maximum number of material in photonuclear library    *
*                                                                      *
*----------------------------------------------------------------------*

      common /mdasa/ das( mdas )
      common /mdasb/ mmmax

*----------------------------------------------------------------------*
*                                                                      *
*       mdas   : total memory * 8 = byte                               *
*       mmmax  : maximum number of total array                         *
*                                                                      *
*       kvlmax : maximum number of regions, cell and material          *
*       kvmmax : maximum number of id for regions, cel and material    *
*                                                                      *
*       itlmax : number of maximum tally entry                         *
*       inevt  : number of collision type for summary                  *
*       isrc   : number of multi-source                                *
*       nbchmax: maximum number of batch assigned to parallel MPI node *
*       pnlmax : maximum number of material in photonuclear library    *
*       mxprodxs : maximum number of produced nuclides in prodxs       *
*                                                                      *
*----------------------------------------------------------------------*
```

11 Additional explanation for parallel computing

There are two types of parallel computing: distributed-memory parallel computing using the MPI protocol, and the shared-memory parallel computing using OpenMP architecture. Parallel PHITS calculation can be performed using either of these methods, or by using a hybrid method. To execute distributed-memory parallel computing, the MPI protocol must be installed in the user's computer; by contrast, there is no additional software required for shared-memory parallel computing. Note that when the same number of CPU core is set, the computational time for the distributed-memory parallel computing is often shorter than for the shared-memory parallel computing. Each executable file must be generated according to the type of the parallel computing employed: see Sec. 10 for details. Fortran compilers recommended to make the executable file are Intel Fortran Compiler 11.1 (or later) and gfortran 4.7 (or 4.8) for the distributed-memory parallel computing, while only the Intel compiler can be used for the shared-memory parallel computing.

In distributed-memory parallel computing, jobs are distributed to each CPU core in batch units. When all jobs assigned to each core are finished, the main core gathers their results. In this mode, all cores individually use an amount of memory equivalent to that used in single processing; thus, the total RAM memory in the computer system must be larger than the product of the memory used for single processing and the core number. As a result, this type of parallel computing is not suitable for calculations requiring a large amount of memory, for instance, calculations involving voxel phantoms.

In shared-memory parallel computing, jobs are distributed to each CPU core in the units of history. In this case, the cores share a large portion of the memories used by PHITS aside from those defined as "thread private" variables. Thus, the memory required in this parallel computing mode is nearly the same as that required for single processing. One disadvantage of shared parallel computing relative to memory-distributed computing is its slower computational time as a result of the competition for access to shared memories. This disadvantage becomes very important for calculations in which memories are frequently updated, such as those using the `[t-sed]` tally.

11.1 Distributed memory parallel computing

11.1.1 Installation of MPI protocol

It is necessary to install a MPI protocol for executing the distributed-memory parallel version of PHITS. In this manual, the installation procedure of MPICH2 into a Windows PC is described, which is necessary for using the MPI version of PHITS included in the PHITS package, i.e. "c:\phits\bin\phitsXXX_win_mpi.exe." For other cases, please set up your machine environment with MPI protocol by yourself.

MPICH2 is mainly developed in Argonne National Laboratory. For easy installation, we recommend to download its version 1.4 from `http://www.mpich.org/static/downloads/1.4/` because the Windows installer is provided in the version. For 64-bit and 32-bit Windows, please select "mpich2-1.4-win-x86-64.msi" and "mpich2-1.4-win-ia32.msi," respectively. During the installation, you have to select "C:\Program Files\MPICH2" (default) as the installation folder, and select "Everyone" as users. You can input any "Passphrase," but you have to set the same "Passphrase" for each Windows PC if you use multiple machines for the MPI calculation. After the installation of MPICH2, you have to right-click "smpd-install.bat" contained in "\phits\document\mpi\", and execute it as the administrator.

11.1.2 Execution of MPI version of PHITS on Windows

For executing the MPI version of PHITS on Windows, you have to write `$MPI`=*M* before the 1st section of your input file, where *M* is the number of parallelization. For example, you have to write

```
$MPI = 4
```

if you want to run PHITS as fast as possible in your computer with 4 CPU cores. Note that the actual number of processing elements (PEs) is $M + 1$, where the additional element is used for controlling each process.

Then, you have to right-click the input file, and send to "phits" in the same manner as executing the single version of PHITS. You have to input your username and password when you first-time run the MPI version of PHITS. If you write both `$OMP` and `$MPI` in your input file, the command written behind is superseded.

11.1.3 Execution of MPI version of PHITS using command line

For Linux, Mac, or multiple Windows PCs connected via network, you have to use command line for executing the MPI version of PHITS. For example, you have to type

```
mpirun -np 5 phits_LinIfort_MPI.exe
```

where "mpirun" is the executable file name of your MPI protocol, "phits_LinIfort_MPI.exe" indicates the PHITS-executable file name and the number of processing elements (PEs) is set following "-np." This command can be sent using a parallel computing submission protocol such as "qsub," in which case the name of the PHITS input file should be written in a text file named `phits.in` with the first line given as

```
file = input_file_name
```

where "input_file_name" is the name of the PHITS input file. This rule is only effective for distributed memory parallel computing. `file=phits.in` can also be written in the first line, with the contents of the PHITS input file added following the second line of `phits.in`: please see Sec. 2.3.2 for further details.

11.1.4 Adjustment of maxcas and maxbch

In distributed-memory parallel computing, jobs are distributed to each CPU core in units of batch. Hence, the number of batches (`maxbch`) should be a multiple of PE−1 (one of the PEs is used for control). If not, PHITS will automatically change `maxbch` to a multiple of PE−1 and adjust the number of histories per batch, `maxcas`, to make the total history number equivalent to the value set in the input file. In this case, relevant comments will be output at the end of the input echo.

In restart mode (`istdev<0`), adjustment of `maxcas` is not performed, as this value should be set to the same value as was written in past tally results.

11.1.5 Treatment of abnormal ending

When PHITS stops as the result of an abnormal end in a PE, the PE is removed from the operation and a total result by remaining PE is given as the final result.

11.1.6 Output file name of dump, dumpall, [t-userdefined]

When using distributed memory parallel computing, output file of `dump, dumpall, or [t-userdefined]` is divided and the IP number is assigned at the end of each divided-filename, such as ".005". If the core number is 4 or 5 digits, the numbering system will increase the number of digits.

11.1.7 Read-in file definition in PHITS

The read-in files used in PHITS are Decay-Turtle source files. Although these files generally have only a small effect on network traffic, they can occasionally reach sizes as large as 100 MB and, if there is a read-in for every event, their effect to network traffic can increase. Thus, the Decay-Turtle data file should be copied and placed in each PE as "\wk\j9999\turtle\sours.dat" and then defined as `file=/wk/j9999/turtle/sours.dat` in the PHITS input.

11.2 Shared-memory parallel computing

11.2.1 Execution

Except when Linux or PHITS is directly specified by a command input, PHITS calculation using shared-memory parallel computing can be executed by adding "`$OMP=`*N*" (*N* is the number of CPU cores to be used) before the first section in a PHITS input file. When $N = 0$, all cores in the computer are used. If $N = 1$ is set, the parallel computing is not used. From version 2.73, the installed executable file of the OpenMP version is available only on

64-bit Windows systems. Note that the function of `$OMP` is available only for using `phits.bat`(on Windows OS) or `phits.sh`(on Mac OS).

In the case when Linux or PHITS is directly specified by a command input, PHITS with shared-memory parallel computing can be executed using the following command:

```
phits_LinIfort_OMP.exe < phits.inp
```

where phits_LinIfort_OMP.exe indicates a PHITS-executable file compiled using the OpenMP option and "phits.inp" is the PHITS input file. Any name can be used for the PHITS input file, as the file name restriction to `phits.in` is not valid in this case. To specify the number of cores to be used for parallel computing, the environment variable "OMP_NUM_THREADS" should be defined. Note that this variable should be set to equal the real number of CPU-cores, not the number of total threadds, when using hyper-threading technology, as parallel computing using this technology does not work in PHITS calculation. The environment variable can be changed as follows:

```
export OMP_NUM_THREADS=8
```

Note that in hybrid parallel computing the environment variables need to be set individually on each node.

From version 2.73, the installed executable OpenMP version file is available only for 64-bit Windows systems, as an error owing to heap memory shortage sometimes occurred on Windows OS when the executable file of the OpenMP version for 32-bit was used with many cores. However, this error is avoided by using the executable file for 64-bit Windows.

11.2.2 Important notices for shared-memory parallel computing

Using only one core for memory-shared parallel computing in PHITS takes approximately twice as much time as is needed for single processing. Therefore, it is useless to select memory-shared parallel computing mode for a computer with only one or two cores.

Segmentation errors occur in executing memory-shared parallel computing PHITS on Linux may be caused by the overuse of the stack memory; in this case, the stack size must be increased using the following command:

```
export OMP_STACKSIZE=1G
```

which sets the stack size to 1 GB.

In principle, the results obtained from single processing and shared-memory parallel computing should be the same. Thus, the user should inform us (phits-office) if inconsistencies are found between the results obtained by these two modes, as these may represent bugs in the programming.

12 FAQ

12.1 Questions related to parameter setting

Q1.1 Input file that works before ver. 2.88 does not work after that version.

A1.1 Several revisions were made in terms of the input file format after ver. 2.89 to avoid frequently-occurring mistakes. In general, it is not necessary to change input file when PHITS is updated, but in some cases, it is necessary. If you encounter an error after the update, please check the following points.

(1) After ver. 2.89, c␣ (␣ is a space), cannot be used as a comment remark in `[material]` section in the default setting. You have to change the comment remarks c␣ in `[material]` to $ or #, or set `icommat=1` in `[parameters]`.

(2) After ver. 2.93, low energy neutrons are transported using nuclear data library in the default setting. Thus, you may encounter an error "There is no cross section table(s) in xsdir" even when you want to transport only photon and electrons. In that case, you have to set `nucdata=0` in `[parameters]` to disable the use of nuclear data library.

(3) After ver. 2.96, it is not allowed to define two or more `[parameters]` sections in an input file.

(4) After ver. 2.96, unnecessary tallies are automatically disabled, depending on the `icntl` parameter. Consequently, `set:` and `infl:` commands are ignored when they are written in the disabled sections. When `set:` or `infl:` commands are ignored, PHITS outputs warning.

Q1.2 Electrons and positrons are not transported.

A1.2 Transport of electrons and positrons are ignored in the default setting because they are time consuming. If you want to transport them, you have to activate EGS5 mode by setting `negs=1,2` in `[parameters]`.

Q1.3 Which nuclear reaction model settings give the most accurate result?

A1.3 In general, the default models give the best results in most cases. However, it is desirable to activate JQMD-2.0 (`irqmd=1`) for precisely simulating nucleon-nucleon interactions, and SMM (`ismm=1`) for precisely estimating the residual nuclide yields after high-energy nuclear reactions, though they are time consumptive.

Q1.4 What kind of simulation is the event-generator mode suited to?

A1.4 Event generator mode is best suited to simulations in which it is necessary to obtain event-by-event information, e.g., detector response calculations and the design of semi-conductor devices. It is also useful for simulations in which the energies and types of charged particles produced by low-energy neutron interactions must be determined. Specifically, event generator mode is generally most suited to simulations using `[T-Deposit]`, `[T-LET]`, `[T-SED]`, `[T-Yield]`, and/or `[T-Product]` tallies. On the other hand, it is not suited to simulations only using `[T-Track]` and/or `[T-Cross]` tallies, such as shielding calculations. See "5.2.27 Event Generator Mode" for further details.

Q1.5 When should the mode for statistical uncertainty (the setting of `istdev`) be changed?

A1.5 We generally recommend using the history variance mode (`istdev=-2` or `2`), in which the statistical uncertainty depends on the total history number (`maxcas*maxbch`), except in the case of shared-memory parallel computing, in which only the batch variance mode (`istdev=-1` or `1`) can be selected. However, the computational time occasionally becomes extremely long in history variance mode, especially in for tallies using a large number of memories, e.g., an `xyz` mesh tally with very fine structure. When performing a PHITS calculation under such conditions, please switch to batch variance mode and set `maxbch` to be greater than 10.

Q1.6 Is it possible to use nuclear data libraries other than JENDL-4.0?

A1.6 Yes. PHITS can use nuclear data libraries that are written in the ACE format, i.e., the MCNP format. Note that the instruction changes for neutron libraries below 20 MeV and others.

The following are instructions on how to use a new nuclear data library in PHITS:

I For neutron libraries below 20 MeV.

I-(1) using a text editor, open the "xsdir" file contained in the package of the new nuclear data library and copy all nuclear data addresses (e.g., 1001.80c 0.999167 xdata/endf71x/H/1001.710nc 0 1 4 17969 0 0 2.5301E-08);

I-(2) using a text editor, open the file "xsdir.jnd" included in the PHITS "data" folder and paste the copied addresses below `directory` around the 180th line: above the first address information of JENDL-4.0 (1001.50c 0.999167 neu/H_001.j40n_300K 0 1 1 30720 0 0 2.585E-08) because PHITS automatically finds the data library of the previously written nucleus in "xsdir.jnd";

I-(3) create a new folder with the appropriate name written in the address file in the PHITS "XS" folder and copy the data files from the new nuclear data library to the created folder.

II For other libraries such as proton, deuteron, photo-nuclear reaction, high energy neutron libraries, etc.

II-(1) using a text editor, open the "xsdir" file contained in the package of the new nuclear data library and copy all nuclear data addresses (e.g., 1001.80c 0.999167 xdata/endf71x/H/1001.710nc 0 1 4 17969 0 0 2.5301E-08) and then using a text editor, open the file "xsdir.jnd" included in the PHITS "data" folder and paste the copied addresses at the end of the file;

II-(2) create a new folder with the appropriate name written in the address file in the PHITS "XS" folder and copy the data files from the new nuclear data library to the created folder;

II-(3) explicitly specify the library ID using `lib(i)` in the `[parameters]` section of your PHITS input file (e.g., `lib(2)=80c` for the above example) and set the maximum energy of the library using `dmax(i)`. If the nuclear data libraries are not available for certain nuclides specified in `[material]`, PHITS automatically employs the nuclear reaction models for the nuclides.

The latest version of ENDF (ENDF/B-VIII.0) in the ACE format can be downloaded from the link below.
`https://nucleardata.lanl.gov/ace/lib80x/`
There are two xsdir files ("xsdir" & "xsdir_2.0") in the package, but only xsdir is compatible with PHITS. You have to replace + by `ptable` when you copy "xsdir" to "xsdir.jnd".

Q1.7 How can neutron fluxes emitted from photo-nuclear reactions with low statistical uncertainty be found?

A1.7 The photonuclear cross section can be biased using the `pnimul` parameter. For example, the probability of photo-nuclear reaction is doubled when `pnimul` is set `=2.0` and the weights of secondary particles emitted from the photo-nuclear reaction are 0.5. If a very high value of `pnimul`, e.g., above 100.0, is selected, the photon fluxes might be altered; therefore, it should be confirmed that the photon fluxes are not significantly altered as a result of changing this parameter.

Q1.8 I cannot restart PHITS simulation as a result of an error related to the inconsistency of a tally. Why is this?

A1.8 This occurs as the result of the loss of significant digits in the PHITS input file. In this case, set `ireschk=1` to induce PHITS to skip the consistency check.

12.2 Questions related to errors occurred in compiling or executing PHITS

Q2.1 I got an error in compiling PHITS. How can I correct this to finish compiling?

A2.1 Intel Fortran Compiler 11.1 (or later) and gfortran 4.7 (or later) can be used. Note that the PHITS office recommends using Intel Fortran Compiler 11.1 (or later). Intel fortran of OneAPI is available for free of charge as explained in "phits/document/OneAPI-Fortran-manual.pdf". Intel Fortran is recommended because PHITS compiled by Intel Fortran is much faster than that by gfortran. See section 10 for further details.

Q2.2 "Access violation" or "Segmentation fault" occurred during the execution of PHITS.

A2.2 It is probably due to a bug related to memory usage in PHITS. Please append `$DBG=1` to the first line of the input file that causes the problem and run PHITS in debug mode. If you get any error messages, please send the messages to PHITS office, together with the information on your machine environment. If you used your own compiled version of PHITS, please recompile PHITS in the debug mode (change DEBUG? = false to true in "makefile"), execute the debug mode with your input file, and send the error message to the office.

Q2.3 An error occurred when I tried to use `infl:` in my PHITS input file.

A2.3 When the `infl:` command is used in the PHITS input file (e.g., "phits.inp"), it is necessary to type `file=phits.inp` in the first line of "phits.inp" file. Otherwise, another input file (e.g., "phits.in") can be created with the first line `file=phits.inp` and used as the PHITS input file. For example:

```
phits_LinIfort < phits.in
```

See manual section 2.3.2 for further details.

Q2.4 An error occurred when I tries to execute PHITS on a Linux console, but I can execute it on Windows using the same input file.

A2.4 There are many possible causes of this error, but the most probable is a difference in the 'return code' used in Linux and Windows. If the input file is prepared in a Windows computer and transferred to a Linux system using FTP software, the status of the transfer mode must be checked, i.e., 'ASCII mode' must be selected in the FTP software.

Q2.5 Can PHITS be executed on Cygwin?

A2.5 Yes. The Cygwin option can be found in the PHITS "makefile."

Q2.6 Which is most suitable for compiling PHITS—Intel, Fortran, or gfortran?

A2.6 We generally recommend using the Intel Fortran compiler because it can make PHITS executable files that are faster than those made using gfortran. When using gfortran, it is vital that you set '-O0' instead of using optimization options because of the possibility that an executable file with optimization will not work correctly. Thus, an executable file compiled using Intel Fortran is on average 3–5 times faster than one compiled using gfortran. Note that the Intel Fortran compiler distributed as OneAPI toolkit is free of charge since 2020.

Q2.7 How should the distributed memory parallel computing (MPI) or shared memory parallel computing (OpenMP) be set depending on the situation?

A2.7 Calculation using MPI is generally faster than calculation using OpenMP if the MPI protocol is installed in the computer. However, because the protocol is not generally pre-installed in Windows or Mac OS systems as a default setting, we recommend the use of OpenMP for executable files. To perform PHITS calculation on huge memories such as high energy nuclear data files or voxel phantoms, the hybrid MPI-OpenMP setting must be used to avoid the possible memory shortages that could occur if only MPI were to be used. In the hybrid setting, the number of MPI processes should be increased, but only up to the point at which the used memory does not overflow the capacity of the computer. Note that PHITS uses a number of CPU cores appropriate to the number of OpenMP processes to control the MPI processes. For example, to set the number of OpenMP processes to eight on a computer with 128 CPU cores, the maximum number of MPI processes can be specified as $128/8 - 1 = 15$.

Q2.8 I got many warning messages about lost particles. What should I do?

A2.8 Several reasons can be considered as the cause of this problem. The most probable reason is wrong definition of your geometry, such as double-defined and undefined regions. Please check your geometry using `[t-gshow]` or `icntl=8`. You may also have to check the following issues.

(1) If you set the location of source particles exactly on (or source direction exactly parallel to) a boundary of certain cells, you have to slightly change the definition of your source.

(2) If you rotate a lattice structure by `[transform]`, you have to set the size of outer frame surface to be slightly smaller than the lattice structure inside it. Please see "phits/lecture/advanced/voxel" in more detail.

(3) If you set `[weight window]` (or `[t-wwg]` to generate it) with `mesh=xyz` that exactly overlaps with a boundary of certain cells, you have to slightly change the definition of the `xyz` mesh. Please be especially careful if you are using parameters to link geometry boundary and `xyz`-mesh ranges. After version 3.34, if the maximum and minimum xyz mesh values for `[weight window]` and `[t-wwg]` are defined as integers, the minimum value is shifted by $-1*$`deltxyz` and the maximum value by $+2*$`deltxyz` from the integer.

If you cannot figure out the reason of lost particles, but the frequency of lost particles is not so high (approximately less than once per several 1000 histories), you can ignore the message. It is probably due to the loss of significant digit.

Q2.9 I fail to execute PHITS with OpenMP parallelization in Linux. The library "libiomp5.so" is required.

A2.9 In order to execute PHITS with OpenMP parallelization, you may be required to install the "libiomp5.so" library. In this case, please install the libomp-dev library by using installation software provided in each Linux distribution. For instance, for Ubuntu case, you can install the library by

```
sudo apt-get install libomp-dev
```

This will create a library "/usr/lib/x86_64-linux-gnu/libomp.so.5" Then please move to this directory

```
cd /usr/lib/x86_64-linux-gnu/
```

and make a symbolic link to libiomp5.so

```
sudo ln -s libomp.so.5 libiomp5.so
```

With this, you should be able to execute PHTIS with OpenMP parallelization.

Q2.10 PHITS execution fails in Linux with an error `GLIBC_2.34 not found`.

A2.10 The executable files for Linux in the PHITS package are using updated GLIBC library. Therefore PHITS execution fails in a computer with older version of Linux OS such as Ubuntu 20.04 LTS. To solve this problem, you need to update OS of your PC or recompile PHITS under your environment by yourself. Some Fortran compiler such as Intel Fortran or Gfortran is required for recompilation. The source code of PHITS is located at phits/src. Please properly set the value of ENVFLAGS in makefile according to your environment and compile PHITS using make command (please see also Sec. 10.3).

Q2.11 I encountered symptoms that are clearly PHITS bugs, such as infinite loops and abnormal termination of calculations. What should I do?

A2.11 Please append `$DBG=1` to the first line of the input file that causes the problem and run PHITS in debug mode. If you get any error messages, please send the messages to PHITS office, together with the information on your machine environment. Additionally if you could also send us the last random number file rijkdmp.inp (or rijkdmp***.inp for MPI parallel calculation), which can be obtained by setting `idmprijk=1` option in `[parameters]`, our debugging process will become easier.

12.3 Questions related to Tallies

Q3.1 What is the difference between the `[t-heat]` and the `[t-deposit]` tallies?

A3.1 These two tallies essentially produce the same values of deposition energies. After version 3.05, `[t-deposit]` gives the deposition energies not only by ionization of charged particles but also kerma of neutral particles. Thus, we do not recommend to use `[t-heat]` after that version.

Q3.2 The track length or fluence of heavy ions calculated by `[t-track]` or `[t-cross]` is strange.

A3.2 This might be caused by a mis-definition of the energy mesh in the tally section. The energies of heavy ions should be defined in MeV in the tally section but should be written in MeV/n (per nucleon) in the `[source]` section. Note that the unit in the tally can be changed to MeV/n by setting `iMeVperu=1` in `[parameters]`.

Q3.3 Results obtained by `[T-LET]` and/or `[T-SED]` tally are strange.

A3.3 The density of the material selected by `letmat` must be checked: if a material not used in your geometry is selected, it is necessary to define its absolute atomic densities (1H 6.893407e-2 16O 3.446704e-2) in the `[material]` section. In addition, if a very fine mesh (more than 10 meshes per decade) in `[t-let]` is set, the results might have zigzag structure. In that case, the LET mesh should be widened.

Q3.4 How can the statistical uncertainty be estimated from the tally output?

A3.4 In Version 2.50 and later the standard deviations or standard errors are correctly output in the tally results. See section "5.2.2 Number of history and bank" for details.

Q3.5 Can the `dump` function be used when executing PHITS in distributed memory parallel computing?

A3.5 This is possible from version 2.30 onward. Please ask the PHITS offce for more information on its detailed usage.

Q3.6 Tally results in boxes obtained using `mesh=reg` and `mesh=xyz` are inconsistent with each other.

A3.6 PHITS automatically calculates the volume of tally regions only in the cases of `mesh=xyz` or `r-z`. Thus, if `mesh=reg` is set and the `[volume]` section is not specified, the volume of the tally region is assumed to be 1 cm^3.

Q3.7 Why do some events deposit energies greater than the incident energy when using `[t-deposit]` with the `output=deposit` option.

A3.7 When exothermic nuclear reactions occur, the total energy of secondary particles becomes greater than the incident energy. Such events are also observed when `mesh=xyz` or `r-z` and `nedisp=t0` based on the output of the algorithm for considering the energy straggling of charged particles in PHITS. In such case, it is necessary to set `mesh=reg` in `[t-deposit]` and define cells corresponding to each `xyz` or `r-z` mesh in the `[cell]` section.

Q3.8 Can I switch x and z axes in 2D drawing of the tally results?

A3.8 Yes. For that purpose, you have to swap `x-type` and `z-type` mesh information, and add the following 3 lines:

```
trcl = 0 0 0 0 0 1 0 1 0 1 0 0
x-txt = x [cm]
y-txt = z [cm]
```

Then, you will see eps file rotated by 90 degree.

Q3.9 Computational time becomes longer when I set up many tallies in my input file. What can I do?

A3.9 You can decrease the computational time by setting `istdev=1` (batch variance mode) or `italsh=1` (sharing tally memory, only for OpenMP), particularly when you define tally consuming large memory size. In addition, `[t-yield]` and `[t-dchain]` cost numerous computational time between PHITS 2.94 and 3.24. If you feel your simulation using `[t-yield]` and `[t-dchain]` is too slow, please contact PHITS office. We will provide the latest version.

Q3.10 Is it possible to easily output the statistical error of the sum-over data?

A3.10 No. You have to set another tally to directly output the sum-over data by setting the mesh number equal to 1.

Q3.11 Is it possible to stop the PHITS simulation when the statistical errors of the tally data only for certain meshes become lower than `stdcut`?

A3.11 No. When `stdcut` is set, PHITS checks the statistical errors of all data in the tally. Thus, you have to limit the meshes only to the regions of your interest in the case of specifying `stdcut` in the tally.

Q3.12 I would like to know the PHITS tallies corresponding to MCNP ones.

A3.12 Please see below for the answer.

- F1: Surface Current = `[t-cross]` with `output=current`
- F2: Surface Flux = `[t-cross]` with `output=flux`
- F4: Track Length cell flux = `[t-track]`
- F4 with FM card: Reaction number = `[t-interact]`,`[t-yield]`,`[t-product]`
- F5: Flux at a point and ring = `[t-point]`
- F6: Energy Deposition = `[t-deposit]` with `output=dose`
- F8: Pulse Height = `[t-deposit]` with `output=deposit`

Q3.13 Tally results obtained from `[t-cross]` with `mesh=reg` are inconsistent with the corresponding data obtained from `[t-cross]` with `mesh=xyz` or `r-z`.

A3.13 Surface area must be defined in `[t-cross]` with `mesh=reg`, otherwise the area is always assumed to be 1.0 cm^2. If a boundary of geometry coincides with a tally plane at cylindrical surface in `r-z` mesh or at a rotated surface in xyz mesh, the tally results might be underestimated due to the problem of significant digit. In such cases, the mesh size should be changed slightly so that it does not coincide with the boundary plane.

12.4 Questions related to source generation

Q4.1 How PHITS outputs be normalized when using an isotropic source (`s-type=9` or `10`, `dir=-all`)?

A4.1 If there is nothing inside the sphere of an isotropic source, the fluence inside the sphere is normalized to $1/\pi/r_1^2$ (/source), where r_1 is the radius of the sphere. Therefore, to convert the tally output (/source) to the unit fluence, it is necessary to multiply the result by πr_1^2. It should be noted that, as the weight control method is employed in generating an isotropic source, event-by-event information cannot be derived from simulations using an isotropic source. To obtain event-by-event information for isotropic irradiation, set `dir=iso` in the `[source]` section (PHITS must be updated to use this function).

Q4.2 What should be done if a source particle is not created in the desired location in a cell?

A4.2 To set the source generating surface (or point) precisely on the surface of a certain cell, PHITS sometimes mis-identifies the cell where it should be located. In this case, the source surface should be moved so that it differs slightly from the cell surface.

12.5 Questions related to PHIG-3D

Q5.1 When starting PHIG-3D on Windows, it stops with the error message of "VCOMP140.DLL not found".

A5.1 Please install Microsoft Visual C++ 2015 redistributable package from website below.
`https://www.microsoft.com/en-US/download/details.aspx?id=53840`
If this cannot resolve the above error, please contact the phits office.

Q5.2 PHIG-3D does not start when we define `set: cXX [X.X]` at the beginning of the sentence in input file.

A5.2 Please define `set:cXX[X.X]` at later line than `[Parameters]` section.

Q5.3 In the `[Material]` section, when we write "C" at the beginning of the sentence in input file, an error maggesage of "too few material input" is displayed. For example, an error will occur in the following situations.

```
mat[5]
        C 2 H 4
```

A5.3 Please revise as follow.

```
mat[5] C 2 H 4
```

Q5.4 When I try to start PHIG-3D on Mac, I get an error message "Cannot be opened because the developer cannot be verified."

A5.4 Please select the PHIG-3D executable file, right-click while holding down the Control key, and select "Open" from the shortcut menu to launch it. You can also grant permissions from "System Preferences" → "Security and Privacy."

Q5.5 A `Not enough free memory` error occurs when I press "draw" button.

A5.5 Please stop other applications that may consume a lot of memory in your computer. If you still observe the same error even after stopping the applications, please click "Setting" in the left tab and reduce "Points/cell" parameter.

12.6 Other questions

Q6.1 Can I use PHITS for commercial purpose?

A6.1 Basically yes. You can use PHITS for commercial purposes such as the design of radiation detector to be sold, and radiation shielding calculation for the accelerator facilities for medical use. However, you have to make an onerous contract if you want to sell a product that includes the executable file of PHITS, such as the treatment planning system based on PHITS. In that case, please contact us.

Q6.2 What kinds of physical processes cannot be handled by PHITS?

A6.2 PHITS cannot handle interactions involved with more than (or equal to) two moving particles such as particle collision experiments, transports and generations of non-ionizing radiation such as visible light, and change of chemical or material properties after irradiation. More detailed information can be found in Appendix A.

Q6.3 I would like to install PHITS to my new computer, but I cannot find the installer. What can I do?

A6.3 Please send a message to PHITS office via “contact” page by selecting “PHITS upgrade patch request” and writing “I would like to install PHITS in a new PC” in the column of “PHITS version”. We will check your registration status and send the latest package in free of charge if you are registered, or ask you to submit application form if not.

APPENDIX

A List of physical processes that cannot be handled by PHITS

- Criticality calculation
- Interaction between two (or more) moving particles (e.g. Simulation for particle collision experiments)
- Effect of electric fields generated by radiation (e.g. Laser acceleration of particles, plasma)
- Electron-induced nuclear reactions via virtual photon production
- Transport and generation of photons below 1 keV (e.g. Cherenkov light, synchrotron radiation, luminescence)
- Physical processes related to electron current (e.g. electron avalanche)
- Charge exchange interactions [137]
- Polarization effect [138]
- Chemical reactions (e.g. Transport and generation of radicals)
- Change of chemical and material profiles after irradiation (e.g. time evolution of defects)
- Particle-induced X-ray emission (PIXE)
- Generation of delayed neutrons

[137] Feasible when the track-structure mode is used
[138] Feasible in the simulation of nuclear florescence resonance (NRF)

B Evaluated nuclear data libraries

The PHITS package includes the evaluated nuclear data libraries of JENDL-4.0 for neutrons below 20 MeV for 405 nuclei listed in Table B.1, JENDL-5[139] for neutrons and protons below 200 MeV for approximately 30 selected nuclei listed in Table B.2, and JENDL-5 for deuteron below 100 MeV/n for 9 nuclei listed in Table B.3. Note that the file size of the complete dataset of JENDL-5 is too large to be included in the PHITS package. If you wish to use JENDL-5 for protons and neutrons for other nuclides, or JENDL-5 for photo-nuclear and alpha-induced nuclear reactions, please follow the instructions below.

(1) Run "jendl5_setup_win.exe" or "jendl5_setup_mac_lin.sh" in "phits/XS" folder. If you cannot find these files in the folder, please update PHITS after version 3.34.

(2) Specify the installation method and the folder to be used.

(3) Specify the address file ("xsdir.jnd") to be edited.

(4) Select the libraries you wish to install.

The installer performs the following steps:

(i) Download and extract the selected JENDL-5 compressed data (7zip format). This process can be skipped depending on the selection made in (2) above.

(ii) Add the address information of the newly downloaded data to the specified "xsdir" file. In this case, JENDL-5 is used in prior to other libraries for all nuclides because it is appended before the information of JENDL-4 (on the line above)[140]. Also, JENDL-5 data ("data/natural_abundance_20c.dat") will be used for natural element expansions.

If you wish to use nuclear data libraries other than JENDL, please see FAQ 1.6. In addition, "phits/XS" folder contains other cross-section data such as the thermal neutron scattering kernel based on JENDL-4.0 (tsl), photo-atomic data library based on EPDL97[141] (pho), electro-atomic data library based on EEDL[142] (ele, not used any more), photo- and electro-atomic data for EGS mode (egs), ionization and excitation cross-section data for track-structure mode (tra), photo-atomic data library for adjoint mode (adj).

[139] Iwamoto O, Iwamoto N, Kunieda S, et al. Japanese evaluated nuclear data library version 5: JENDL-5. J Nucl Sci Technol. 2023;60(1):1-60.

[140] Neutron library will give priority to the last downloaded data. Thus, please be careful when you selected the neutron data library with temperature other than 300 K. You can change the priority by manually editing the order of the address information written in "xsdir.jnd."

[141] Cullen et al., UCRL-50400, 6, Rev.5, 1997.

[142] Perkins et al., UCRL-50400, 31, 1991.

Table B.1: List of target nuclei whose neutron data in JENDL-4.0 are included in the PHITS package ($E \leq$20 MeV, 300 K)

$^{1}_{1}$H	$^{2}_{1}$H	$^{3}_{2}$He	$^{4}_{2}$He	$^{6}_{3}$Li	$^{7}_{3}$Li	$^{9}_{4}$Be	$^{10}_{5}$B	$^{11}_{5}$B	$^{12}_{6}$C
$^{14}_{7}$N	$^{15}_{7}$N	$^{16}_{8}$O	$^{19}_{9}$F	$^{23}_{11}$Na	$^{24}_{12}$Mg	$^{25}_{12}$Mg	$^{26}_{12}$Mg	$^{27}_{13}$Al	$^{28}_{14}$Si
$^{29}_{14}$Si	$^{30}_{14}$Si	$^{31}_{15}$P	$^{32}_{16}$S	$^{33}_{16}$S	$^{34}_{16}$S	$^{36}_{16}$S	$^{35}_{17}$Cl	$^{37}_{17}$Cl	$^{40}_{18}$Ar
$^{39}_{19}$K	$^{40}_{19}$K	$^{41}_{19}$K	$^{42}_{20}$Ca	$^{43}_{20}$Ca	$^{44}_{20}$Ca	$^{46}_{20}$Ca	$^{48}_{20}$Ca	$^{45}_{21}$Sc	$^{46}_{22}$Ti
$^{47}_{22}$Ti	$^{48}_{22}$Ti	$^{49}_{22}$Ti	$^{50}_{22}$Ti	$^{50}_{23}$V	$^{51}_{23}$V	$^{50}_{24}$Cr	$^{52}_{24}$Cr	$^{53}_{24}$Cr	$^{54}_{24}$Cr
$^{55}_{25}$Mn	$^{54}_{26}$Fe	$^{56}_{26}$Fe	$^{57}_{26}$Fe	$^{58}_{26}$Fe	$^{59}_{26}$Fe	$^{59}_{27}$Co	$^{58}_{28}$Ni	$^{59}_{28}$Ni	$^{60}_{28}$Ni
$^{61}_{28}$Ni	$^{62}_{28}$Ni	$^{64}_{28}$Ni	$^{65}_{28}$Ni	$^{63}_{29}$Cu	$^{65}_{29}$Cu	$^{64}_{30}$Zn	$^{65}_{30}$Zn	$^{66}_{30}$Zn	$^{67}_{30}$Zn
$^{68}_{30}$Zn	$^{70}_{30}$Zn	$^{69}_{31}$Ga	$^{71}_{31}$Ga	$^{70}_{32}$Ge	$^{72}_{32}$Ge	$^{73}_{32}$Ge	$^{74}_{32}$Ge	$^{76}_{32}$Ge	$^{75}_{33}$As
$^{74}_{34}$Se	$^{76}_{34}$Se	$^{77}_{34}$Se	$^{78}_{34}$Se	$^{79}_{34}$Se	$^{80}_{34}$Se	$^{82}_{34}$Se	$^{79}_{35}$Br	$^{81}_{35}$Br	$^{80}_{36}$Kr
$^{82}_{36}$Kr	$^{83}_{36}$Kr	$^{84}_{36}$Kr	$^{85}_{36}$Kr	$^{86}_{36}$Kr	$^{85}_{37}$Rb	$^{86}_{37}$Rb	$^{87}_{37}$Rb	$^{84}_{38}$Sr	$^{86}_{38}$Sr
$^{87}_{38}$Sr	$^{88}_{38}$Sr	$^{89}_{38}$Sr	$^{90}_{38}$Sr	$^{89}_{39}$Y	$^{90}_{39}$Y	$^{91}_{39}$Y	$^{90}_{40}$Zr	$^{91}_{40}$Zr	$^{92}_{40}$Zr
$^{93}_{40}$Zr	$^{94}_{40}$Zr	$^{95}_{40}$Zr	$^{96}_{40}$Zr	$^{93}_{41}$Nb	$^{94}_{41}$Nb	$^{95}_{41}$Nb	$^{92}_{42}$Mo	$^{94}_{42}$Mo	$^{95}_{42}$Mo
$^{96}_{42}$Mo	$^{97}_{42}$Mo	$^{98}_{42}$Mo	$^{99}_{42}$Mo	$^{100}_{42}$Mo	$^{99}_{43}$Tc	$^{96}_{44}$Ru	$^{98}_{44}$Ru	$^{99}_{44}$Ru	$^{100}_{44}$Ru
$^{101}_{44}$Ru	$^{102}_{44}$Ru	$^{103}_{44}$Ru	$^{104}_{44}$Ru	$^{105}_{44}$Ru	$^{106}_{44}$Ru	$^{103}_{45}$Rh	$^{105}_{45}$Rh	$^{102}_{46}$Pd	$^{104}_{46}$Pd
$^{105}_{46}$Pd	$^{106}_{46}$Pd	$^{107}_{46}$Pd	$^{108}_{46}$Pd	$^{110}_{46}$Pd	$^{107}_{47}$Ag	$^{109}_{47}$Ag	$^{110m}_{47}$Ag	$^{111}_{47}$Ag	$^{106}_{48}$Cd
$^{108}_{48}$Cd	$^{110}_{48}$Cd	$^{111}_{48}$Cd	$^{112}_{48}$Cd	$^{113}_{48}$Cd	$^{114}_{48}$Cd	$^{116}_{48}$Cd	$^{113}_{49}$In	$^{115}_{49}$In	$^{112}_{50}$Sn
$^{114}_{50}$Sn	$^{115}_{50}$Sn	$^{116}_{50}$Sn	$^{117}_{50}$Sn	$^{118}_{50}$Sn	$^{119}_{50}$Sn	$^{120}_{50}$Sn	$^{122}_{50}$Sn	$^{123}_{50}$Sn	$^{124}_{50}$Sn
$^{114}_{50}$Sn	$^{115}_{51}$Sb	$^{116}_{51}$Sb	$^{117}_{51}$Sb	$^{118}_{51}$Sb	$^{119}_{51}$Sb	$^{120}_{52}$Te	$^{122}_{52}$Te	$^{123}_{52}$Te	$^{124}_{52}$Te
$^{125}_{52}$Sn	$^{126}_{52}$Sb	$^{127m}_{52}$Sb	$^{128}_{52}$Sb	$^{129m}_{52}$Sb	$^{130}_{52}$Sb	$^{132}_{52}$Te	$^{127}_{53}$Te	$^{129}_{53}$Te	$^{130}_{53}$Te
$^{131}_{53}$I	$^{135}_{53}$I	$^{124}_{54}$Xe	$^{126}_{54}$Xe	$^{128}_{54}$Xe	$^{129}_{54}$Xe	$^{130}_{54}$Xe	$^{131}_{54}$Xe	$^{132}_{54}$Xe	$^{133}_{54}$Xe
$^{134}_{54}$Xe	$^{135}_{54}$Xe	$^{136}_{54}$Xe	$^{133}_{55}$Cs	$^{134}_{55}$Cs	$^{135}_{55}$Cs	$^{136}_{55}$Cs	$^{137}_{55}$Cs	$^{130}_{56}$Ba	$^{132}_{56}$Ba
$^{134}_{56}$Ba	$^{135}_{56}$Ba	$^{136}_{56}$Ba	$^{137}_{56}$Ba	$^{138}_{56}$Ba	$^{140}_{56}$Ba	$^{138}_{57}$La	$^{139}_{57}$La	$^{140}_{57}$La	$^{140}_{58}$Ce
$^{141}_{58}$Ce	$^{142}_{58}$Ce	$^{143}_{58}$Ce	$^{144}_{58}$Ce	$^{141}_{59}$Pr	$^{143}_{59}$Pr	$^{142}_{60}$Nd	$^{143}_{60}$Nd	$^{144}_{60}$Nd	$^{145}_{60}$Nd
$^{146}_{60}$Nd	$^{147}_{60}$Nd	$^{148}_{60}$Nd	$^{150}_{60}$Nd	$^{147}_{61}$Pm	$^{148}_{61}$Pm	$^{148m}_{61}$Pm	$^{149}_{61}$Pm	$^{151}_{61}$Pm	$^{144}_{62}$Sm
$^{147}_{62}$Sm	$^{148}_{62}$Sm	$^{149}_{62}$Sm	$^{150}_{62}$Sm	$^{151}_{62}$Sm	$^{152}_{62}$Sm	$^{153}_{62}$Sm	$^{154}_{62}$Sm	$^{151}_{63}$Eu	$^{152}_{63}$Eu
$^{153}_{63}$Eu	$^{154}_{63}$Eu	$^{155}_{63}$Eu	$^{156}_{63}$Eu	$^{157}_{63}$Eu	$^{152}_{64}$Gd	$^{153}_{64}$Gd	$^{154}_{64}$Gd	$^{155}_{64}$Gd	$^{156}_{64}$Gd
$^{157}_{64}$Gd	$^{158}_{64}$Gd	$^{160}_{64}$Gd	$^{159}_{65}$Tb	$^{160}_{65}$Tb	$^{154}_{66}$Dy	$^{156}_{66}$Dy	$^{158}_{66}$Dy	$^{159}_{66}$Dy	$^{160}_{66}$Dy
$^{161}_{66}$Dy	$^{162}_{66}$Dy	$^{163}_{66}$Dy	$^{164}_{66}$Dy	$^{162}_{68}$Er	$^{164}_{68}$Er	$^{166}_{68}$Er	$^{167}_{68}$Er	$^{168}_{68}$Er	$^{170}_{68}$Er
$^{169}_{69}$Tm	$^{168}_{70}$Yb	$^{170}_{70}$Yb	$^{171}_{70}$Yb	$^{172}_{70}$Yb	$^{173}_{70}$Yb	$^{174}_{70}$Yb	$^{176}_{70}$Hf	$^{174}_{72}$Hf	$^{176}_{72}$Hf
$^{177}_{72}$Hf	$^{178}_{72}$Hf	$^{179}_{72}$Hf	$^{180}_{72}$Hf	$^{181}_{72}$Hf	$^{182}_{72}$Hf	$^{181}_{73}$Ta	$^{180}_{74}$W	$^{182}_{74}$W	$^{183}_{74}$W
$^{184}_{74}$W	$^{186}_{74}$W	$^{184}_{76}$Os	$^{186}_{76}$Os	$^{187}_{76}$Os	$^{188}_{76}$Os	$^{189}_{76}$Os	$^{190}_{76}$Os	$^{192}_{76}$Os	$^{197}_{79}$Au
$^{196}_{80}$Hg	$^{198}_{80}$Hg	$^{199}_{80}$Hg	$^{200}_{80}$Hg	$^{201}_{80}$Hg	$^{202}_{80}$Hg	$^{204}_{80}$Hg	$^{204}_{82}$Pb	$^{206}_{82}$Pb	$^{207}_{82}$Pb
$^{208}_{82}$Pb	$^{209}_{83}$Bi	$^{223}_{88}$Ra	$^{224}_{88}$Ra	$^{225}_{88}$Ra	$^{226}_{88}$Ra	$^{225}_{89}$Ac	$^{226}_{89}$Ac	$^{227}_{89}$Ac	$^{227}_{90}$Th
$^{228}_{90}$Th	$^{229}_{90}$Th	$^{230}_{90}$Th	$^{231}_{90}$Th	$^{232}_{90}$Th	$^{233}_{90}$Th	$^{234}_{90}$Th	$^{229}_{91}$Pa	$^{230}_{91}$Pa	$^{231}_{91}$Pa
$^{232}_{91}$Pa	$^{233}_{91}$Pa	$^{230}_{92}$U	$^{231}_{92}$U	$^{232}_{92}$U	$^{233}_{92}$U	$^{234}_{92}$U	$^{235}_{92}$U	$^{236}_{92}$U	$^{237}_{92}$U
$^{238}_{92}$U	$^{234}_{93}$Np	$^{235}_{93}$Np	$^{236}_{93}$Np	$^{237}_{93}$Np	$^{238}_{93}$Np	$^{239}_{93}$Np	$^{236}_{94}$Pu	$^{237}_{94}$Pu	$^{238}_{94}$Pu
$^{239}_{94}$Pu	$^{240}_{94}$Pu	$^{241}_{94}$Pu	$^{242}_{94}$Pu	$^{244}_{94}$Pu	$^{246}_{94}$Pu	$^{240}_{95}$Am	$^{241}_{95}$Am	$^{242}_{95}$Am	$^{242m}_{95}$Am
$^{243}_{95}$Am	$^{244}_{95}$Am	$^{244m}_{95}$Am	$^{240}_{96}$Cm	$^{241}_{96}$Cm	$^{242}_{96}$Cm	$^{243}_{96}$Cm	$^{244}_{96}$Cm	$^{245}_{96}$Cm	$^{246}_{96}$Cm
$^{247}_{96}$Cm	$^{248}_{96}$Cm	$^{249}_{96}$Cm	$^{250}_{96}$Cm	$^{245}_{97}$Bk	$^{246}_{97}$Bk	$^{247}_{97}$Bk	$^{248}_{97}$Bk	$^{249}_{97}$Bk	$^{250}_{97}$Bk
$^{246}_{98}$Cf	$^{248}_{98}$Cf	$^{249}_{98}$Cf	$^{250}_{98}$Cf	$^{251}_{98}$Cf	$^{252}_{98}$Cf	$^{253}_{98}$Cf	$^{254}_{98}$Cf	$^{251}_{99}$Es	$^{252}_{99}$Es
$^{253}_{99}$Es	$^{254}_{99}$Es	$^{254m}_{99}$Es	$^{255}_{99}$Es	$^{255}_{100}$Fm					

Table B.2: List of target nuclei whose proton and neutron data in JENDL-5 are included in the PHITS package (E <200 MeV, 300 K)

$^{1}_{1}$H	$^{2}_{1}$H	$^{6}_{3}$Li†	$^{7}_{3}$Li†	$^{9}_{4}$Be†	$^{12}_{6}$C	$^{13}_{6}$C	$^{14}_{7}$N	$^{16}_{8}$O	$^{27}_{13}$Al
$^{28}_{14}$Si	$^{29}_{14}$Si	$^{30}_{14}$Si	$^{40}_{20}$Ca	$^{42}_{20}$Ca	$^{43}_{20}$Ca	$^{44}_{20}$Ca	$^{46}_{20}$Ca	$^{48}_{20}$Ca	$^{54}_{26}$Fe
$^{56}_{26}$Fe	$^{57}_{26}$Fe	$^{58}_{26}$Fe	$^{63}_{29}$Cu	$^{65}_{29}$Cu	$^{204}_{82}$Pb	$^{206}_{82}$Pb	$^{207}_{82}$Pb	$^{208}_{82}$Pb	$^{209}_{83}$Bi

†Proton data only.

Table B.3: List of target nuclei whose deuteron data in JENDL-5 are included in the PHITS package (E <100 MeV/n, 300 K)

$^{6}_{3}$Li	$^{7}_{3}$Li	$^{9}_{4}$Be	$^{12}_{6}$C	$^{13}_{6}$C	$^{27}_{13}$Al	$^{63}_{29}$Cu	$^{65}_{29}$Cu	$^{93}_{41}$Nb

Index

<source>, 87, 118
$S(\alpha,\beta)$, 9
Δ angular distribution, 62
δ-ray, 21, 57, 291
γ decay, 59
-all, 20, 352
[Frag Data], 52
[T-4Dtrack], 2, 4, 35, 335
[T-Dchain], 3
[T-WWBG], 2
[T-WWG], 2
[User Defined Interaction], 2
[Volume], 3
[Weight Window], 56
[anatally], 47, 234
[cell], 1, 6, 15, 34–36, 76, 78, 137, 141, 147, 155–160, 162–166, 168, 171, 175, 206, 305
[counter], 10, 20, 21, 34, 209, 230, 261, 269, 278, 284, 301
[data max], 9, 11, 34, 52, 190
[delta ray], 21, 34, 185, 291
[elastic option], 10, 34, 71, 189
[electro magnetic field], 34, 88, 183
[end], 35, 36
[forced collisions], 4, 20, 34, 41, 199
[frag data], 12, 16, 34, 191
[importance], 12, 34, 41, 75, 194, 262
[libout], 4
[magnetic field], 34, 88, 177, 183
[mat name color], 34, 75, 206
[mat time change], 34, 176
[material], 3, 11, 13, 34, 36, 74, 80, 137, 140, 155, 158–160, 162–166, 168, 187, 206, 262, 267, 288, 294, 302, 305, 347, 348, 352
[multiplier], 2, 4, 19, 20, 34, 79, 202, 203
[paraemters], 4, 37, 199
[parameters], 9, 11, 12, 15, 19, 32, 34, 36, 41, 47, 118, 121, 137, 167, 177, 183, 187, 190, 191, 196, 197, 199, 204, 206, 228, 232, 234, 248, 253, 257, 260, 261, 265, 269, 273, 274, 278, 284, 287, 290, 294, 297, 301–303, 309, 311, 314, 317, 324, 326, 328, 347, 350, 352
[reg name], 34, 208
[repeated collisions], 15, 34, 200
[source], 12, 34, 61, 67, 86, 171, 177, 183, 258, 282, 302, 317, 350, 352
[super mirror], 34, 188
[surface], 6, 34–36, 82, 102, 141, 155–160, 162–166, 168, 171, 321
[t-3dshow], 35, 47, 208, 215, 230, 328
[t-DPA], 11
[t-adjoint], 61, 258
[t-cross], 3, 4, 6, 7, 11, 16, 19, 20, 35, 61, 74, 103, 121, 202, 215, 221, 228, 230, 249, 347, 350, 351
[t-dchain], 1, 3, 5, 9, 10, 12, 15, 17–19, 27, 29, 32, 35, 210, 215, 302, 351
[t-deposit2], 5, 8, 35, 48, 138, 215, 267
[t-deposit], 1, 3, 8, 9, 12, 15–18, 20, 35, 48, 60, 85, 138, 215, 235, 261, 267, 270, 291, 347, 350, 351
[t-dpa], 1, 13, 18, 35, 85, 215, 285
[t-gshow], 19, 35, 47, 161, 208, 215, 225, 228, 324, 339
[t-heat], 20, 35, 215, 270, 350
[t-interact], 4, 5, 8, 17, 20, 35, 74, 121, 215, 298, 351
[t-let], 20, 35, 138, 215, 288, 291, 347, 350
[t-point], 7, 8, 17, 35, 74, 121, 202, 215, 238, 255, 351
[t-product], 5, 7, 18, 20, 35, 47, 74, 85, 100, 103, 121, 215, 230, 347, 351
[t-rshow], 35, 47, 208, 215, 228, 229, 326, 339
[t-sed], 7, 10, 17, 20, 35, 138, 185, 215, 291, 344, 347, 350
[t-star], 20, 35, 215, 298
[t-time], 35, 74, 103, 121, 215, 230, 295
[t-track], 12, 17, 18, 35, 61, 74, 121, 202, 203, 215, 235, 238, 244, 249, 255, 258, 302, 347, 350
[t-userdefined], 20, 35, 215, 319, 345
[t-volume], 35, 47, 215, 317
[t-wwbg], 35, 47, 196, 215, 314
[t-wwg], 1, 4, 20, 35, 196, 215, 309
[t-yield], 1, 9, 10, 12, 18, 19, 35, 59, 85, 210, 215, 224, 274, 280, 302, 347, 351
[temperature], 34, 175
[timer], 34, 211
[title], 34, 46
[track structure], 4, 9, 11, 34, 186
[transform], 9, 10, 19, 34, 141, 156, 163, 171–174, 177, 230
[user defined interaction], 4, 34, 212
[user defined particle], 4, 34, 212
[volume], 34, 75, 201, 217, 245, 262, 272, 305, 309, 318, 351
[weight window], 2, 4, 12, 19, 20, 34, 41, 56, 195, 196, 215, 230, 238, 262, 309, 314
[ww bias], 34, 47, 56, 196, 215, 314
#all, 1
$DBG, 3, 30
$MPI, 30, 344
$OMP, 30, 344
2d-type, 227, 228, 246, 250, 258, 262, 268, 272, 276, 282, 287, 289, 292, 295, 300
3dshow, 230

a-curr, 249, 252
a-flux, 11
a-type, 4, 16, 88, 90–92, 99, 101–103, 106, 126, 127, 220, 221, 244, 249, 253, 280, 284, 297
aama.dat, 79
abnormal end, 345
abort, 31

absorption, 62, 299
ACE format, 9, 11
acmin, 306
act, 298
actlow, 119
adjoint mode, 4, 14, 61, 258
ag1, 93, 124, 127
ag2, 93, 124, 127
aHalf, 337
all, 8, 88, 99, 101, 102, 190, 195, 197, 202, 203, 216, 245, 249, 255, 262, 267, 270, 272, 275, 280, 285, 287, 288, 291, 295, 298, 299, 310, 315, 317, 328, 330
alti, 124
am, 188
amp, 305
analysis function, 15, 16, 47, 235
anatally, 3, 7, 9–11, 234, 248, 257, 260
andit, 62
ANGEL, 8, 19, 26, 48, 226, 227, 229, 230, 250, 255, 258, 262, 268, 272, 282, 287, 289, 292, 295, 300, 310, 315, 324, 326, 330
angel, 246, 276
angel parameter, 226, 250, 255, 258, 295
angle mesh, 220, 244, 249, 280
angle straggling, 70
anih, 210
annih, 299
ANOVA, 16, 236
aonucl, 306
aoreg, 306
arc-DPA, 11
area, 254
array size, 343
ascat1, 70
ascat2, 70
ASCII, 73
ATIMA, 16, 57, 88, 138
atmflu, 299
atom, 210
atomic, 280, 282, 299
auge, 210
Auger, 18
auger, 299
Auger electron, 10, 20, 119, 291
automatic calculation of region volume, 35, 47, 215, 317
autorun, 15, 16, 235
axis, 9, 10, 20, 48, 59, 218, 223, 224, 227, 229, 231, 245, 250, 255, 258, 262, 267, 268, 270, 274, 281, 289, 292, 295, 298, 310, 315, 324, 326, 339
axishow, 330

b-curr, 252
bank, 48
batch, 31, 48, 74, 230, 345
batch variance mode, 248, 278, 351
batch.out, 19, 31, 79
Bertini, 52, 62
bHalf, 337
binary, 73, 76
Bitmap, 247, 262, 277, 283, 287, 297, 301, 324, 326
bitrseed, 48
blank, 38, 216
blank line, 36
bmpout, 247, 262, 272, 277, 283, 287, 297, 301, 324, 326
Boolean operator, 155
BOX, 143, 160, 162, 164–166
box, 330
Bragg scattering, 71, 189
brem, 210
brems, 299
bright, 330
BROND-3.1, 305

c-type, 236
C/X, 142
C/Y, 142
C/Z, 142
CAD, 2, 8, 19
cdiam, 291
cell definition, 155, 156, 160, 161
cell number, 101, 155, 196, 215, 216
cell parameter, 155, 156, 162, 163
CENDL-3.1, 305
change, 176
character variable, 6
characteristics x-ray, 19
chard, 66
charge, 223, 274
chart, 9, 59, 223, 274
Chart of the Nuclides, 306
chem, 11, 139, 187
chmax, 230, 248, 253, 257, 258, 265, 269, 272, 278, 284, 287, 290, 294, 297, 301, 311, 313
chmax(i), 230
chmin, 230, 248, 253, 257, 258, 265, 269, 272, 278, 284, 287, 290, 294, 297, 301, 311, 313
chmin(i), 230
chplane, 313
chwei, 311, 313
cluster plot, 227
cmap, 3
cmax, 226, 236
cmin, 226, 236
cmin(i), 51, 191
cmkm, 226
cmmm, 226
cmmt, 226
cmnm, 226
cmpt, 210
cmum, 226
coll, 209, 211

color, 206
color plot, 227
Command Prompt, 24
comment mark, 36, 137, 141, 155, 171
compilation, 340
compile, 86
complex DSB, 11
composition ratio, 137
compound, 137
compton, 299
Compton scattering, 10
COND, 138
contour, 227
coordinate transform number, 141, 163
cos, 223, 250
cosine, 90–92, 94
Cosmic-ray source, 13, 123
cosmic-ray source, 11, 13
cosmic-ray source mode, 2, 8, 10
Coulomb diffusion, 70
counter, 105, 209, 230, 231, 248, 257, 258, 278, 311, 313
cpu time, 75
ctmax, 210, 230, 248, 253, 257, 258, 261, 265, 269, 272, 278, 284, 287, 290, 294, 297, 301, 311, 313
ctmin, 230, 248, 253, 257, 258, 261, 265, 269, 272, 278, 284, 287, 290, 294, 297, 301, 311, 313
Cugnon, 62
current, 249, 252, 351
cut, 102
cut-off energy, 51, 62, 79, 295
cut-off time, 56
cut-off weight, 56
cutoff, 275, 295
cval, 106
CX, 142
CY, 142, 159, 163–165
CZ, 142

dark, 330
data, 101, 102
dbcutoff, 57
dcay, 210
DCHAIN, 2, 3, 6, 7, 9, 11–13, 15–18, 26, 79, 302
dchain, 9, 59, 223, 274
decay, 38, 282, 295, 299
decay-turtle, 86, 96, 345
DECDC, 79, 118
dedxfile, 138
dedxfnc, 262
dedxfnc1, 267
dedxfnc2, 267
deg, 245
delayed neutron, 63
delt, 210
delt0, 70, 78
delta-ray, 34, 185
deltaray, 299
deltb, 78
deltc, 5, 78
deltg, 78, 177, 183
deltm, 61, 74, 78
deltt, 78
deltxyz, 4
delxyz, 78
den, 140
density, 155
depatom, 124
deposit, 8, 19, 20, 48, 262, 351
dfano, 18, 265
dfano1, 269
dfano2, 269
DICOM2PHITS, 13
dipole electromagnet, 177
dir, 3, 10, 20, 88, 90–94, 96–99, 101–103, 106, 124, 317, 352
direction cosine, 96
distributed memory parallel computing, 320
distributed-memory parallel computing, 20, 30, 48, 49, 344, 349
dl0, 134
dl1, 134
dl2, 134
dmax, 3, 190, 255
dmax(1), 6
dmax(12), 7, 52, 64
dmax(13), 7, 52, 64
dmax(14), 7, 52, 64
dmax(2), 52, 274, 280
dmax(i), 51, 190
dmpmulti, 104
DNA damage, 11
DNA damage yields, 17
dnb, 63
dom, 88, 90–92, 97–99, 101–103, 134
Doppler broadening, 10
Doppler effect, 59
dose, 262, 351
DPA, 1
dpa, 285, 287
dpf, 134
dpnmax, 9, 51
drag and drop, 25
drd, 134
dresol, 18, 20, 265
dresol1, 269
dresol2, 269
dtime, 118
duct source, 134
dump, 3, 8, 11, 13, 86, 103, 106, 230, 253, 284, 297, 338, 345
dump file, 103
dump source, 1, 10
dumpall, 47, 73, 79, 81, 253, 284, 297, 319, 345

dversion, 306
DWBA, 62
dxw, 134
dyw, 134

e-dst, 328
e-mode, 10, 20, 60, 85, 204, 255, 261, 270, 274, 280, 285, 288, 298, 302, 303
e-phi, 328
e-smax, 61, 258
e-smin, 61, 258
e-the, 328
e-tyep, 109
e-type, 4, 7, 13, 86, 90–94, 97–99, 101–103, 106, 110, 112–114, 116, 117, 220–222, 244, 249, 253, 255, 258, 262, 280, 284, 295, 297, 298, 310, 315
e0, 86, 90–94, 96–99, 101–103, 106, 109
e1-t, 223, 267
e1-type, 267
e12, 223, 267
e2-t, 223, 267
e2-type, 267
e21, 223, 267
EAF-2010, 305
EAF-3.1, 306
EBITEM, 6, 7, 14, 59, 274
EEDL, 355
eg0, 116
eg1, 116, 117, 124
eg2, 116, 117, 124
eg3, 116, 117
EGAF, 6, 7
EGS, 2, 5
EGS5, 16, 17, 19, 20, 52, 60, 64, 79, 137, 177, 183, 187, 298, 347
eielfmax, 52
eielfmin, 52
einclmax, 52
einclmin, 52
eisobar, 52
ejamnu, 52
ejampi, 52
ejamqmd, 52
elastic, 275, 282, 299
elastic scattering, 58, 80
electric field map, 183
electro-magnetic field map, 16
electromagnetic field, 14, 71, 74
electron, 270, 272
element, 137
Element file, 167
element symbol, 137
elf, 183
ELIB, 137, 138
ELL, 143
ellipse torus, 9
elowbias, 313
elowthre, 313
elst, 210
emax, 200, 203
emcnf, 63
emcpf, 63
emin, 185, 200, 287, 288
emin(1), 51, 187
emin(12), 19, 52, 60, 64, 187
emin(13), 19, 52, 60, 64, 187
emin(14), 52, 64
emin(19), 187
emin(2), 52
emin(i), 51
emumax, 68
emumin, 68
enclos, 253
ENDF/B-VII.1, 305
ENDF/B-VIII.0, 305, 306
energy mesh, 220, 244, 249, 255, 258, 280, 295, 298, 310, 315
energy straggling, 70
eng, 195, 196, 223, 245, 250, 255, 258, 262, 281, 295, 298, 310
eng-t, 223, 262
eng1, 267
eng2, 267
eng2let, 250, 280
ENSDF, 14, 306
environ, 2, 123, 125
EPDL97, 355
eps, 224, 230, 247, 252, 257, 258, 262, 268, 272, 277, 283, 287, 290, 294, 297, 301, 311, 313, 315, 324, 326, 330
epseudo, 15, 52
epsout, 228, 229, 247, 252, 257, 258, 262, 268, 272, 277, 283, 287, 290, 294, 297, 301, 311, 313, 315, 324, 326, 330
epstfl, 14, 66
eqmdnu, 52
escape, 295
esmax, 51
esmin, 51
ESTEP, 138
et0, 116
et1, 116
et2, 116
et3, 116
ets_dea, 300
ets_e-exc, 300
ets_elast, 300
ets_ioniz, 300
ets_ioniz_e-exc, 300
ets_p-exc, 300
ets_plasmon, 300
ets_r-exc, 300
ets_v-exc, 300

ETSART, 4
etsmax, 51, 187
etsmin, 51, 187
etsmode, 10
evaporation, 59
event generator mode, 60, 85, 230, 347
event-generator mode, 298
exa, 98
Excel, 228
execution, 23
ExPORT-PHITS, 1
Extended statistical indicators, 4, 238

f-curr, 252
f-mesh, 315
factor, 5, 88, 104, 225, 245, 250, 255, 258, 262, 268, 272, 275, 282, 287, 289, 292, 295, 300, 310
FAQ, 347
fcl, 199
FENDL/A-2.0, 305
FENDL/A-3.0, 305
FENDL/D-1, 306
figure of merit, 238
file, 48, 103, 191, 224, 225, 231, 245, 250, 255, 258, 262, 268, 270, 274, 281, 285, 289, 292, 295, 298, 303, 310, 315, 317–319, 324, 326, 330, 335
file(1), 5, 52, 79
file(10), 79
file(11), 79
file(12), 79
file(13), 79
file(14), 79
file(15), 73, 79
file(18), 76, 79, 167
file(19), 76, 79
file(20), 52, 79
file(21), 79, 306
file(22), 32, 79
file(23), 79
file(24), 79, 118, 123
file(25), 79
file(26), 79
file(27), 79
file(28), 79
file(29), 79, 138
file(30), 2, 79
file(4), 79
file(6), 4, 37, 41, 72, 75, 79, 137, 190, 206, 317
file(7), 52, 61, 79, 137
FILL, 77, 155, 159, 160, 162, 164–166, 168, 169
fiss, 210
fission, 282, 299
flight mesh, 78
fluence, 244, 249
FLUENT, 16
fluo, 210
flux, 249, 252, 351
foamout, 16, 247, 262, 277, 283, 287
forced collisions, 56
format, 336
Fortran, 38, 117, 127, 129, 132, 227, 341

gap, 177
GAS, 138
gasegs, 66
GEM, 59, 210
geometry check, 339
gfortran, 340
ghigh, 202
Ghostscript, 25
ghostview, 230
ginfo, 246, 276
glat, 125
glong, 125
glow, 202
gmake, 340, 341
gmumul, 68
GQ, 142
gravity, 71
gravx, 71
gravy, 71
gravz, 71
groups, 221
gshow, 8, 9, 16, 19, 47, 208, 228, 230, 246, 252, 258, 262, 272, 276, 282, 287, 289, 293, 297, 300, 339
gslat, 15, 75, 229, 246, 253, 258, 262, 272, 276, 284, 287, 290, 293, 297, 301, 311, 313, 324, 326
gsline, 15, 75

HalfLifeMax, 337
HalfLifeMin, 337
Hashimoto’s formula, 58
HDF5, 2
heat, 270
heaven, 328
heavyion, 299
HEX, 143
High-energy event generator mode, 9
history counter, 4
history-counter, 12, 230, 248, 253, 257, 258, 265, 269, 272, 278, 284, 287, 290, 294, 297, 301, 311, 313
HistoryFile, 336
HistoryMax, 336
HLIB, 137, 138
hnxslib, 305
HSB, 206

iadjoint, 61
ianataldchain, 234
iangform, 16, 249
iannih, 5, 18, 119

iaonucl, 306
iaprim, 66
iauger, 66
iaugers, 119
ibatch, 12, 89
ibod, 35
ibound, 66
ibrdst, 66
icday, 124
icells, 15, 76, 79
icenv, 123–125
ichkmat, 15, 78
icmonth, 124
icntl, 4, 6, 7, 10, 13, 14, 37, 47, 73, 79, 80, 196, 204, 228, 232, 234, 284, 314, 317, 324, 326, 328, 339, 347
icommat, 36, 80, 137, 347
icput, 42, 75
icrdm, 58
icrhi, 58, 192
ICRP, 8
ICRU Report, 291
ICRU90, 14, 138
icxnp, 8, 11, 58
icxsni, 58, 192
icxspi, 58
icyear, 124
idam(i), 80
idbg, 35
idcylib, 306
idelt, 78
idmpmode, 103, 104, 253, 284, 297
idosunit, 7
idpa, 13, 285
idpara, 73, 231
idwba, 62
iechrl, 244, 258, 262, 268, 272, 274, 283, 287, 290, 291, 297, 300, 326
iedgfl, 66
iegsout, 64
iegsrand, 64
ieispl, 66
ielas, 58
ielctf, 71, 183
ieleh, 52
ielms, 58
iels, 210
ierrout, 12, 72
iextstat, 238, 248, 257
ifission, 59
ifixchg, 19, 57, 88
ifx, 1
igamma, 6, 19, 59, 79, 85, 274, 280
igchk, 78
igeomem, 48
igerr, 78
iggcm, 75
igmuppd, 15, 68
ih2o, 17, 57
iidfs, 62
ikerman, 60
ikermap, 60
ilchain, 11
imagnf, 71, 78, 177
iMeVperU, 10, 121
iMeVperu, 20, 74, 204, 350
imout, 74
imp, 194
impacr, 66
implicit capture, 56
Importance, 13
importance, 56, 75, 194
imsegs, 64
imubrm, 68
imucap, 67
imuint, 68
imuppd, 68
in, 209, 211
INC-ELF, 2, 18, 52
incelf, 52
INCL, 52
inclg, 52
include files, 343
incohr, 66
infl, 6, 227, 318
infl:, 29, 36, 347
info, 225, 274
infout, 41, 72
inmed, 62
input echo, 47, 56, 74, 75, 197
input file, 33
installation, 23
installation folder, 79
Intel Fortran, 340, 341
internal conversion electron, 19
interpolation, 202
inucr, 4, 6, 7, 80
inxslib, 305
Ionization Potential, 17
ionization potential, 57
ipara, 75
ipegs, 64
iphter, 66
iphtout, 306
ipngdr, 67
ipnint, 13, 64, 67
iprdst, 66
iprofr, 10, 66
iprtb2, 305
iqmdscm, 67
iraylr, 66
iredufmt, 303
ireschk, 2, 48, 309, 348
irlet, 21, 57

irqmd, 52, 347
irskip, 48, 233
isaba, 63
isbias, 93, 99
iscinful, 52
iscorr, 87
ismm, 59, 347
iso, 20, 352
isobar, 52
isomer, 2, 280
ispfs, 89
isrc, 343
istabl, 306
istdbat, 17, 56
istdcut, 17, 56
istdev, 48, 104, 230, 233, 248, 262, 278, 318, 345, 347, 351
isumtally, 232
itall, 12, 17, 32, 74, 89, 230, 235, 248, 253, 257, 260, 265, 269, 273, 278, 284, 287, 290, 294, 297, 301, 303, 311, 317
italsh, 15, 48, 351
itdecn, 16, 305
itdecs, 16, 305
itetra, 2, 77, 79
itetvol, 77
itgchk, 2, 7, 77
itimrand, 48
itlmax, 343
its_e-exc, 300
its_elas, 300
its_hit, 300
its_ioniz, 300
its_ioniz_e-exc, 300
ITSART, 1, 5, 7, 9
itstep, 74
ityp, 195, 197
iunr, 63
iunrst, 66
ivoxel, 15, 76, 79, 167
iwwbias, 56, 196, 314
iwwbnk, 49
izst, 88, 177, 183

JAM, 19, 52, 58
JAMQMD, 52
JEFF-3.1A, 305
JEFF-3.3, 305
JENDL, 11, 348
JENDL-4.0, 52, 137, 305, 348, 355
JENDL-4.0/HE, 9, 12, 14, 18, 355
JENDL-5, 1, 3, 5, 278, 355
JENDL/AD-2017, 305
JENDL/DDF-2015, 306
JENDL/DEU-2020, 12
JENDL/HE-2007, 11, 58
JENDL4, 7
jmode, 6
jmout, 5, 74
jpsf, 103
JQMD, 38, 52, 59
JQMD-2.0, 52, 347

K/X, 142
K/Y, 142
K/Z, 142
kerma, 1
kerma approximation, 20, 60, 85, 261, 270, 350
kerma factor, 14, 80
kf code, 223
kf-code, 38, 231
kmout, 74, 137
knockelec, 299
knoe, 210
KURBUC, 5, 6, 15, 139, 187, 342
kurbuc_e-cap, 300
kurbuc_e-exc, 300
kurbuc_e-stp, 300
kurbuc_elast, 300
kurbuc_hit, 300
kurbuc_ioniz, 300
kurbuc_ioniz_e-exc_e-cap, 300
KUROTAMA, 58
kvlmax, 343
KX, 142
KY, 142
KZ, 142

l-dst, 328
l-phi, 328
l-the, 328
l-type, 220, 288
lagrange, 202
Lagrangian interpolation, 2
Landau, 70
LAT, 77, 155, 160, 162, 164–166, 168, 169, 324, 326
LaTeX, 206, 208
lattice, 14, 88, 155, 159, 162, 164–166, 175, 177, 185, 194, 195, 197, 199, 201, 209, 211, 216, 217, 254
lattice coordinate, 216
lattice coordinate system, 160, 161, 164–166, 216
LET, 262, 267, 288
let, 223, 289
LET mesh, 220
letmat, 138, 261, 262, 288, 294, 350
letmat1, 267
letmat2, 267
level structure, 216
lib, 5
lib(i), 9, 51, 190
library, 47, 52, 137–139
LIKE *n* BUT, 162, 163
LIKE n BUT, 155, 162

lin, 202
line, 330
line connection, 221
line thickness, 230
lineal energy, 291
Linux, 23, 28, 169, 341, 346
lionprd, 2, 55
log, 202
lost particle, 78
lpolar, 66
Lynch's formula, 70

m-source, 258
Mac, 23, 25, 169, 341, 346
macro body, 141, 143, 154
magnetic field, 14, 71, 74
magnetic field map, 14, 179
make, 340
makefile, 11, 340, 341
manatally, 234, 236
MARS-PF, 79
marspf.in, 79
mass, 223, 274
mass density, 138
mass ratio, 74
MAT, 155, 163
mat, 137, 140, 176, 190, 203, 206
matadd, 75, 155, 206
material, 137, 229, 247, 258, 262, 270, 277, 280, 285, 288, 291, 295, 298, 310, 328
material density, 74, 138
material name, 228
material number, 75, 137, 139, 155, 163, 203, 206, 228, 287, 297, 301
Mathematical expression, 38
matinbox, 330
maxact, 298
maxangel, 247, 252, 257, 265, 268, 277, 283, 287, 290, 294, 296, 301
maxbch, 47, 48, 73, 104, 345, 347
maxbnk, 48
maxcas, 47, 48, 73, 104, 345, 347
maximum, 116, 117, 143
maximum value, 221, 222
Maxwellian, 16
MCNP, 74, 351
mdas, 11
mdbatima, 42, 57
mean, 298
mesh, 4, 15, 18, 20, 99, 161, 164–166, 195, 196, 216–219, 238, 244, 245, 249, 254, 258, 262, 267, 270, 274, 280, 285, 288, 291, 295, 298, 303, 310, 315, 317, 324, 326, 351
mesh definition, 218–221, 228
mesh type, 221
mesh width, 221, 222
meta-stable, 3
method, 317
mgf, 177, 183
microbody, 6
Microsoft Visual Studio, 340
mID, 9, 11, 186
MinGW-w64, 340
mingw32-make, 341
minimum, 116, 117, 143
minimum value, 221, 222
mirror, 328
mixture, 137
mm, 188
model, 192, 291
Moliere, 70
MorP, 8, 298
mother, 1, 200, 275, 280, 285, 298
MPI, 20, 341, 344, 349
MPICH2, 30, 344
mset, 203
mtinfo, 4, 205
multi-source, 87, 88, 101, 118, 130, 258
multiplier, 6, 12, 202, 203, 247, 253, 257, 260
multisource, 6
muon capture, 67
muon-induced nuclear reaction, 67
muonic atom cascade, 19
mwhere, 56
MWO formula, 58
mxlv, 164
mxnuclei, 9, 275, 306
mxspln, 56

n-coll, 200
n-evap, 200
n-mesh, 315
na, 126
name, 206, 208
NASA, 58
natural, 75
natural abundance, 137
natural isotope, 302
naz, 35
nbproden, 238, 248, 257
nc, 236
ncvalout, 4, 37, 72
ndata, 1, 10, 210, 278, 303
ndedx, 42, 57, 138
ndis, 3
ne, 112–114, 202
nedisp, 16, 70, 78, 138, 351
negative pions, 10
negative sense, 156
negs, 7, 21, 51, 52, 60, 64, 187, 255, 298, 347
neispl, 66
nesting structure, 164
neutrino, 69
neutron capture, 63

neutron multiplicity, 62
neutron optics, 110, 114
nevap, 59, 85
nfcseg, 56, 199
nfile, 232, 234
ngem, 19, 59
ni, 118
nlatcel, 265, 301
nlatmem, 265, 301
NLIB, 137, 138
nlost, 42, 78
nm, 116, 117, 124
NMTC, 57
nn, 127
no ionization, 47
no reaction, 47
nocas, 253, 284, 297
nocoh, 63
Node file, 167
non, 177, 187, 208, 218
nonu, 63
noobch, 253, 284, 297
norm, 119, 305
normww, 312
Notepad++, 25
npidk, 10, 62
nrandgen, 48
nrecover, 72
NRF, 3
nsms, 226
nspred, 21, 68, 70, 78, 138
nsps, 226
nssc, 226
nsus, 226
ntetelem, 77
ntetsurf, 77
ntmax, 104, 106
ntrnore, 69
ntt, 128
nuc, 140
nucdata, 51, 52, 347
nucl, 210
nuclcal.out, 79
nuclear, 282, 299
nuclear data, 74, 137, 138
nuclear resonance fluorescence, 67
nucleus, 190, 275
nuclide, 38, 118, 137
nudtvar, 20, 319
number, 202
number of group, 222
number of history, 48
nwsors, 41, 75
nx, 99
ny, 99
nz, 20, 99, 254

o-curr, 252
o-type, 128
oa-curr, 249, 252
oa-flux, 11
ob-curr, 252
of-curr, 252
off, 36
Open MP, 341
OpenMP, 11, 13, 344, 349, 351
opt, 16, 191
out, 209, 211
outer void, 155
output, 4, 8, 11, 19, 20, 48, 100, 161, 225, 249, 252, 262, 270, 275, 280, 282, 287, 295, 299, 324, 326, 328, 351
outtime, 305

P, 142, 162
p-type, 110, 112–114, 116, 117
pairprod, 299
parallel calculation, 73
parallel computing, 29, 48, 341, 344
param.inc, 164, 343
ParaView, 3, 283, 287, 297, 301, 324
part, 8, 9, 20, 190, 194–196, 199, 200, 202, 203, 209, 222, 223, 245, 249, 255, 262, 267, 272, 274, 280, 285, 288, 292, 295, 298, 310, 315, 335
particle density, 74, 138
particle identification, 38
particle parameter, 222
PE, 31, 48, 49, 73, 345
Pearlstein-Niita's formula, 58
pedestal, 1, 312
PEGS, 66
PEGS5, 79
pg1, 93
pg2, 93
Phase-space source, 14, 94
phel, 210
phi, 88, 90–92, 97–99, 101–103
PHIG-3D, 1, 5, 8, 10–13, 35, 352
PHIG3D, 77
PHITS-Chem, 1, 4
PHITS-Pad, 4
PHITS-UDM, 7
phits.in, 345
phits.out, 41
PHITSPATH, 5
photo-nuclear, 3, 67
photo-nuclear data libraries, 9
photo-nuclear data library, 51
photo-nuclear reaction, 348
photoelec, 299
photon, 270
PinnedOption, 337
PinnedParticles, 337
PinnedTimeFix, 337

PinnedTimeOption, 337
PLIB, 137, 138
pnimul, 67, 348
point, 255
polar, 178
polarization, 178
positive sense, 156
pprd, 210
precision, 336
primary, 2, 282
Prob, 8
prob, 298
probability density function, 238
Processing Element, 48, 49, 73
prodenmn, 238, 248, 257
prodenmx, 238, 248, 257
product, 275
proj, 5, 61, 123, 191
pseudo-random number generator, 16
Pulse magnet, 71
pulse magnet, 177
PX, 142, 157–160, 162–166
px, 168
PY, 142, 157–160, 162–166
py, 168
PZ, 142, 157–160, 162–166
pz, 168
pz0, 96

q-type, 126, 127
q:, 36
qc, 188
qp:, 36
quadrupole electromagnet, 177

r, 223, 245, 250, 262, 270, 274, 281, 289, 292, 295, 298
r-from, 254
r-in, 188, 254
r-mesh, 315
r-out, 188, 254, 315, 317, 330
r-to, 254
r-type, 218, 221
r-z, 19, 171, 215, 218, 244, 249, 254, 262, 270, 280, 285, 288, 291, 295, 298, 351
r0, 19, 90, 97, 188, 317
r1, 19, 90, 92, 93, 97, 267
r2, 93, 97, 267
rad, 245
radiation doses, 202
random number, 48
ratio, 137
rayl, 210
RBE, 291
RCC, 143
rdam(i), 80
REC, 143
ref, 209, 211
reflective boundary, 141, 167
reg, 47, 88, 104, 106, 161, 164–166, 175, 177, 183, 187, 189, 194–196, 199–201, 208, 209, 211, 215–218, 223, 229, 244, 245, 249, 254, 262, 267, 268, 270, 274, 280, 281, 285, 288, 289, 291, 292, 295, 298, 303, 310, 315, 317, 326, 335, 351
reginbox, 330
region boundary, 228
region error, 78
region mesh, 215
region name, 208, 228
region number, 194, 199, 201, 209, 211, 254
repeated structure, 155, 158, 159, 161, 162, 166, 194, 216
resc2, 31
resc3, 31
resfile, 31, 224, 245, 250, 255, 258, 262, 268, 270, 274, 281, 285, 289, 292, 295, 298, 310, 317
residual nuclei, 59
resol, 228, 230, 246, 252, 258, 262, 272, 276, 282, 287, 290, 293, 297, 301, 311, 313, 315, 324, 326, 330
resolution, 228–230
restart, 10
restart calculation, 48, 225, 245, 250, 255, 258, 262, 268, 270, 274, 281, 285, 289, 292, 295, 298, 308–310, 317, 318, 345
Restricted LET, 57
rev, 351
RHO, 155
rhomat, 294
RHP, 143
RI source, 3, 6, 7, 305
RI source mode, 8
rigid, 125
rijk, 32
rijklst, 31
ring, 255
RIPL, 14
RIsource.ack, 20
RIsource.dat, 118
rn, 91, 92
RPP, 143, 169
rpp, 168
rseed, 48
rshow, 4, 47, 208, 228, 230, 246, 258, 262, 272, 276, 282, 287, 289, 293, 297, 300, 310, 315, 339
RT-PHITS, 1, 8, 13
ruth_min, 9
rx, 94
ry, 94
rz, 48, 223, 245, 262, 270, 274, 281, 289, 292, 295, 298

S, 142
s-type, 2, 3, 10, 13, 14, 17, 19, 20, 86, 88, 90–94, 96–99, 101, 103, 106, 124, 317, 318, 352

$S(\alpha,\beta)$, 139
samepage, 3, 6, 224, 245, 251, 255, 263, 267, 270, 281, 285, 289, 292, 295, 299
sangel, 17, 227, 246, 250, 255, 258, 262, 268, 272, 276, 282, 287, 289, 292, 295, 300, 324, 326, 330
sangel parameter, 227
SANS, 63
Sato's formula, 58
scdy, 226
schr, 226
SCINFUL mode, 11, 14, 52
SCINFUL-QMD, 14, 52
scm_d, 67
scm_h0, 67
scm_rcls, 67
scmn, 226
scoring mesh, 215, 218, 219
scyr, 226
se-type, 291
se-unit, 291
SED, 291
sed, 292
self-fission neutron source, 7
send to, 24
set, 106
set:, 36, 37, 232, 234, 347, 352
sfile, 232, 234, 236
shadow, 330
shared memory parallel computing, 320
shared-memory parallel computing, 13, 30, 48, 344, 345, 347, 349
Shen, 58
simple, 270, 287
size, 206, 208
SkipRate, 335
SkipRate-of, 335
small angle neutron scattering, 63
SMM, 59, 347
SO, 142
soft error rates, 19, 202
solarmod, 124
sors.f, 129
source, 100, 282
source check, 47
SPAR, 57, 138
special, 278
special reaction cross section data, 278
specific energy, 291
SPH, 143
spin, 105, 178, 231
SQ, 142
standard output, 79
star density, 298
statistical uncertainty, 15, 48, 224
stdcut, 17, 248, 253, 257, 260, 265, 269, 273, 278, 284, 287, 290, 294, 297, 301, 303, 311, 317, 351
stopping power, 16, 138
suf, 102
SumatraPDF, 25
sumfactor, 232
summary output file, 41
sumtally, 10, 37, 47, 232, 309
surface, 254
surface definition, 141
Surface definition source, 86, 102
surface number, 141, 154, 161
surface sense, 156
surface symbol, 141, 142
switching energy, 51
swtm(i), 56
SX, 142
sx, 67, 88, 103, 105, 106, 231
SY, 142
sy, 67, 88, 103, 105, 106, 231
symbol, 38
SZ, 142, 156–158
sz, 67, 88, 103, 105, 106, 231

t, 223, 245, 250, 255, 258, 262, 267, 281, 295, 298, 310
t-e1, 223, 267
t-e2, 223, 267
t-eng, 223, 262
t-type, 104, 106, 128, 220, 221, 244, 249, 253, 255, 258, 262, 267, 280, 284, 295, 297, 298, 310, 315
t0, 128
tab, 35
Table of Isotopes 8th ed., 306
targ, 191
target, 305
tc, 128
td, 128
TENDL, 11
TENDL-2017, 305
tet, 2, 4, 15, 18, 215, 219, 244, 245, 274
TetGen, 168
Tetra-mesh, 86
tetra-mesh, 101
tetra-mesh geometry, 141
tetra-mesh source, 6
Tetra.bin, 77
tetrahedral-mesh, 8
tetrahedral-mesh geometry, 2, 5, 6
tetrahedron, 77, 101
tetrahedron geometry, 9, 16, 19, 20, 101, 167
tetreg, 101
TFILE, 168, 169
tg-list, 305
the, 223, 250
the Moliere theory, 70
thermal neutron scattering kernel, 355
tim, 195
time, 176, 177
time mesh, 220, 244, 249, 255, 258, 280, 295, 298, 310, 315

timeevo, 305
timeout, 48
title, 225, 226, 246, 250, 255, 258, 262, 268, 272, 276, 282, 287, 289, 293, 295, 300, 303, 310, 317, 324, 326, 330
tmax(i), 56
TMP, 155, 175
tmp, 175
tn, 128
torus surface, 9
totfact, 87, 88, 104, 118, 123, 124, 305
track structure mode, 8, 11, 139, 261
track-structure mode, 4, 15, 17, 18, 20, 34, 51, 79, 186, 299
transform, 6, 230, 331
transform number, 171
transmut, 299
TRC, 3, 143
TRCL, 88, 155, 163, 331
trcl, 102, 104, 177, 183, 230, 246, 253, 258, 262, 272, 276, 284, 287, 290, 293, 297, 301, 311, 313, 324, 326
trcle, 183
trclm, 183
ts_bio, 299
ts_hit, 299
ts_ioniz, 299
ts_w, 299
TSFAC, 168, 169
tsmax, 51
tw, 128
TX, 142
TY, 142
typ, 177
TZ, 142

U, 88, 155, 158–160, 162, 164–166, 168, 175, 177, 185, 194, 195, 197, 199, 201, 209, 211, 216, 254
udtpara, 20, 319
udtvar, 20, 319
unit, 18, 20, 74, 225, 231, 245, 250, 255, 258, 262, 267, 272, 275, 281, 285, 289, 292, 295, 298, 310
universe, 77, 88, 101, 158, 160, 161, 164–166, 175, 177, 185, 194, 195, 197, 199, 201, 209, 211, 216, 254
user defined cross sections, 191
user definition source, 106
User-defined anatally, 235
user-defined anatally, 12, 13
User-defined character, 37
user-defined integer variable, 80
user-defined real variable, 80
user-defined tally, 13
user-defined variable, 36, 234
usrdfn1.f, 261, 267
usrdfn2.f, 17, 261, 267
usrdfn3.f, 19
usrelst, 71
usrelst1.f, 189
usrelst2.f, 189
usresol.f, 18, 269
usrmgf3.f, 178
usrmgt, 71
usrmgt1.f, 177
usrmgt2.f, 177
usrsors.f, 19, 86, 106
usrtally.f, 319

val, 326
value, 326
variance of variance, 238
Vavilov, 70
verbose, 336
void, 155
VOL, 155
vol, 201, 217, 218, 244, 262, 268, 272, 274, 283, 287, 290, 291, 297, 300
volmat, 229, 247, 258, 262, 272, 277, 283, 287, 290, 294, 297, 301
volume, 75, 77, 201, 216–218, 244, 245, 258, 262, 268, 272, 274, 283, 287, 290, 291, 297, 300, 311, 313, 326
volume and area calculation, 47, 93
volume correction, 229
voxel, 76, 166
voxel phantom, 166
vtkfmt, 247, 262, 272, 277, 283, 287, 297, 301, 324
vtkout, 247, 262, 272, 277, 283, 287, 297, 301, 324

w-ang, 328
w-dst, 328
w-hgt, 328
w-mnh, 328
w-mnw, 328
w-wdt, 328
wc1(i), 56, 199
wc2(i), 56, 191
WED, 143
weight cutoff, 199, 200
weight value, 52
Weight Window, 13
weight window, 56, 195
Weight Window Bias Generator, 196, 314
Weight Window Generator, 196, 309
wem, 94
wgt, 88, 103
width, 230, 246, 252, 258, 262, 272, 276, 282, 287, 290, 293, 297, 301, 311, 313, 315, 324, 326, 330
Windows, 23, 340, 346
wm, 188
Wobbler magnet, 71, 177
wsurvn, 56
wt0, 96
wupn, 56

wwbg, 315
wwbi, 196
wwg, 310
wwi, 195
wwmax, 312

x, 223, 224, 245, 250, 262, 270, 274, 281, 289, 292, 295, 298
X-ray, 63
x-txt, 229, 246, 252, 262, 268, 272, 276, 282, 287, 289, 292, 295, 300, 315, 324, 326, 330
x-type, 99, 195, 196, 219, 221, 229, 351
x0, 90–94, 96–98, 103, 106, 218, 315, 317, 318, 328
x1, 90, 91, 94, 97, 98, 103, 106, 315, 317
x2, 94, 98
x3, 98
xdec, 5, 226
xexp, 5, 226
xlin, 226
xlog, 226
xmax, 226
xmin, 226
xmrad1, 94
xmrad2, 94
xnum, 63
Xorshift64, 16
xp, 96
xq, 96
xsdir, 3, 9, 12, 23, 79, 137, 348
xsmemory, 11, 48
XY, 142
xy, 10, 48, 223, 227–229, 245, 250, 262, 270, 274, 281, 289, 292, 295, 298, 310, 315, 324, 326, 339
xyz, 12, 13, 18, 20, 47, 48, 99, 171, 195, 196, 215, 219, 228, 244, 249, 254, 262, 270, 280, 285, 288, 291, 295, 298, 310, 324, 326, 339, 347, 351
xyz-mesh type source, 6
xz, 245, 262, 270, 274, 281, 289, 292, 295, 298, 310, 315, 324, 326, 339

y, 223, 224, 245, 250, 262, 270, 274, 281, 289, 292, 295, 298
y-txt, 229, 246, 252, 262, 268, 272, 276, 282, 287, 289, 292, 295, 300, 315, 324, 326, 330
y-type, 99, 195, 196, 219, 221, 229
y0, 90–94, 96–98, 103, 106, 218, 315, 317, 328
y1, 90, 91, 94, 97, 98, 103, 106, 315, 317
y2, 94, 98
y3, 98
ydec, 5, 226
yexp, 5, 226
ylin, 226
ylog, 226
ymax, 226
ymin, 226
ymrad1, 94
ymrad2, 94
yp, 96
yq, 96
yz, 223, 228, 229, 245, 262, 270, 274, 281, 289, 292, 295, 298, 310, 315, 324, 326, 339

z, 20, 223, 245, 250, 262, 270, 274, 281, 289, 292, 295, 298
z-mesh, 315
z-txt, 229, 246, 252, 262, 268, 272, 276, 282, 287, 289, 292, 295, 300, 315, 326, 330
z-type, 20, 99, 195, 196, 218, 219, 221, 229, 254, 351
z0, 90–94, 96–98, 103, 106, 315, 317, 328
z1, 90–92, 94, 97, 98, 103, 106, 315, 317
z2, 98
z3, 98
zlin, 326
zlog, 326
ZP, 142
zx, 223, 229

Ver.2.52	2012/12/27
Ver.2.64	2013/11/19
Ver.2.76	2015/03/23
Ver.2.82	2015/12/16
Ver.2.88	2016/09/29
Ver.3.02	2017/12/01
Ver.3.10	2019/04/03
Ver.3.16	2019/09/26
Ver.3.17	2019/10/29
Ver.3.20	2020/03/19
Ver.3.21	2020/09/14
Ver.3.22	2020/10/09
Ver.3.23	2021/01/12
Ver.3.24	2021/03/12
Ver.3.25	2021/07/09
Ver.3.26	2021/09/01
Ver.3.27	2022/03/22
Ver.3.28	2022/05/09
Ver.3.29	2022/09/12
Ver.3.30	2023/01/10
Ver.3.31	2023/04/07
Ver.3.32	2023/05/16
Ver.3.33	2023/09/01
Ver.3.34	2024/03/26
Ver.3.35	2025/03/28

PHITS development members :
Tatsuhiko Sato[1], Yosuke Iwamoto[1], Shintaro Hashimoto[1,†], Tatsuhiko Ogawa[1],
Takuya Furuta[1], Shinichiro Abe[1], Takeshi Kai[1], Yusuke Matsuya[1,2], Norihiro Matsuda[1],
Yuho Hirata[1], Takuya Sekikawa[1], Lan Yao[1], Pi-En Tsai[1,*], Hunter Ratliff[1,**], Hiroshi Iwase[3],
Yasuhito Sakaki[3], Kenta Sugihara[3], Nobuhiro Shigyo[4], Lembit Sihver[5], and Koji Niita[6]
[1]Japan Atomic Energy Agency (JAEA), [2]Hokkaido University
[3]High Energy Accelerator Research Organization (KEK), [4]Kyushu University,
[5]Vienna University of Technology, Austria,
[6]Research Organization for Information Science & Technology (RIST)
[*]Current affiliation: Zap Energy, USA, [**]Current affiliation: HVL, Norway
[†]Editor of this user's manual

Reference List

Nuclear reaction models

Model name	Reference
JAM (Jet AA Microscopic transport model)	Nara et al., Phys. Rev. C, 2000; 61: 024901
Bertini	Bertini, Phys Rev; 1969, 188.4: 1711
INCL4.6 (Liege Intra-Nuclear Cascade model)	Boudard et al., Phys. Rev. C, 2013; 87: 014606
INC-ELF	Sawada, Nucl. Instr. & Meth. B, 2012; 291, 38-44
JQMD (JAERI Quantum Molecular Dynamics)	Niita et al., Phys. Rev. C, 1995; 52: 2620
JQMD Ver.2 (JAERI Quantum Molecular Dynamics Ver.2)	Ogawa *et al.*, Phys. Rev. C, 92 (2015) 024614
JAMQMD (JAM + JQMD)	Ogawa et al. Ogawa et al., Phys. Rev. C 2018; 98: 024611
DWBA (Distorted Wave Born Approximation)	Hashimoto et al, Nucl. Instr. Meth. *B*, 2014; 333: 27-41.
μ -pair production	Sakaki et al., Nucl. Instr. & Meth. A, 2020; 977, 164323.
μ -interaction model	Abe et al, J. Nucl. Sci. Technol., 2017; 54.1: 101-110.
ν -interaction model	N/A
GEM (Generalized Evaporation Model)	Furihata, Nucl. Instr. Meth. B, 2000; 171: 251
GEM Ver.2 (Generalized Evaporation Model)	N/A
JENDL-4 (Japanese Evaluated Nuclear Data Library ver. 4):	Shibata et al., J. Nucl. Sci. Technol., 2011; 48: 1
JENDL-5 (Japanese Evaluated Nuclear Data Library ver. 5)	Iwamoto et al., J. Nucl. Sci. Technol., 2023; 60: 1
Photo-nuclear reaction models	Noda et al, J. Nucl. Sci. Technol., 2014; 52(1), 57–62.

NRF (Nuclear Resonance Fluorescence model)	Ogawa et al., J. Nucl. Sci. Technol., 2016; 53: 1766
SCINFUL-QMD	D. Satoh et al. J. Nucl. Sci. Technol., 2022; 59: 1047-1060
SMM (Statistical Multifragmentation Model)	Ogawa, et al, Nucl. Instr. Meth. A, 2013; 723, 36-46.
EBITEM (ENSDF-Based Isomeric Transition and isomEr production Model)	Ogawa, et al, J. Nucl. Sci. Technol., 2023, 61(1), 68–73.
Event-generator model Ver.1	Niita, et al. *International Conference on Nuclear Data for Science and Technology*. EDP Sciences, 2007. p. 1167-1169.
Event-generator model Ver.2	Ogawa, et al. Nucl. Instr. Meth. A, 2014, 763: 575-590.
Kurotama model	Iida, et al, J. Phys. Soc. Japan, 2007; 76, 044201, L. Sihver et al., Nucl. Instr. & Meth. B, 2014 ; 334, 34-39
MWO formula	K. Minomo, K. Washiyama, and K. Ogata, J. Nucl. Sci. Technol. 2017 ; 54, 127-130.
Iwamoto's fission model	H. Iwamoto, and S. Meigo, J. Nucl. Sci. Technol., 56, 2019; 160-171.
Muon virtual photo-nuclear reaction model and negative muon atomic capture model	S. Abe and T. Sato, J. Nucl. Sci. Technol., vol. 54, pp. 101-110, 2017.

Atomic reaction models

Model name	Reference
ATIMA (Stopping power calculation code)	https://web-docs.gsi.de/~weick/atima/
EGS5 (Electron Gamma Shower ver. 5):	Hirayama et al., SLAC-R-730 & KEK Report 2005-8
EPDL97 (Evaluated Photon Data Library '97 version):	Cullen et al., UCRL-50400

PHITS-KURBUC (Kyushu University Radiobiology Unit Code)	Liamsuwan et al., Phys. Med. Biol., 2013; 58: 673 Y. Matsuya et al. Phys. Med. Biol., 2021, 66, 06NT02
PHITS-ETS (Electron Track Structure code)	Kai et al., Radiat. Phys. Chem. 2015; 108: 13
PHITS-ETS-for-Si (Electron Track Structure code for Si)	Yuho Hirata *et al* 2022 *Jpn. J. Appl. Phys.* **61** 106004
ITSART (Ion Track Structure model for Arbitrary Radiation and Targets)	Ogawa et al., Sci. Rep. 2021; 11: 24401
ETSART (Electron Track Structure model for ARbitrary Targets)	Hirata et al., Jpn. J. Appl. Phys. 2023; 62: 106001
Delta-ray production model [Delta Ray]	Butts et al, Radiation Research, 1967; 30, 855-871
Muon Bremstrralung and electron positron pair-production model	S. Abe and T. Sato, J. Nucl. Sci. Technol., 2017, 54, pp. 101-110,

Geometry functions

Name of function	Reference
Tetrahedral mesh (LAT=3)	Furuta et al. Phys. Med. Biol. 2017, 62, 4798
ICRP boxel phantom	ICRP Publication 110
mesh-type reference computational phantoms (MRCPs)	ICRP Publication 145

Source functions

Name of function	Reference
Cosmic-ray source (e-type = 28,29)	T.Sato, PLOS ONE 10, 2015, e0144679 ; T.Sato, PLOS ONE 11, 2016, e0160390
Fission source (ispfs > 0)	Verbeke et al, 2014 ; UCRL-AR-228518.
RI source	ICRP Publication 107; Endo et al, JAERI, 1347 (2005)

Tallies

Name of functions	Reference
[T-DPA] tally	Y. Iwamoto et al., Journal of Nuclear Materials, 538, 152261 (2020)
Microdosimetric distribution tally [T-SED] model = 0	Sato et al. Radiat. Prot. Dosim. 2006, 122, 41 Sato, et al, Radiat. Res. 2009, 171, 107-117 .
Microdosimetric distribution tally [T-SED] model = 1	Sato et al. *Phys. Med. Biol.*, 2023; 68.15: 155005.
[T-Dchain] tally + DCHAIN Dchain-sp¥	Ratliff, et al., Nucl. Instr. & Meth. B, 2020; 484: 29-41.
[T-adjoint] tally	Malins et al, *EPJ Web of Conferences*. 2017, 06001.
[T-WWG] & [Counter]	Sato *et al.* Nucl. Instr. & Meth. B, 2024; 557: 165535
Systematic uncertainty calculation	Hashimoto et al, J. Nucl. Sci. Technol. 2019; 56, 345-354.

Utilities

Name of function	Reference
DNA damage code phits-utility¥usrtally¥DNAdamage	Matsuya et al. Int. J. Radiat. Biol. 2022, 98(2), 148–157
Chemical code phits-utility¥usrtally¥ChemCode	Y. Matsuya et al Phys. Med. Biol., 2024, 69, 035005 Matsuya et al., Phys. Chem. Chem. Phys., 2025. DOI: 10.1039/d4cp04216f
PHIG-3D	Ohnishi, SoftwareX, 2021; 14, 100681
OpenMP parallelism	Furuta et al. World Congress on Medical Physics and Biomedical Engineering,. 2013, 2099-2102.
RT-PHITS	Furuta, et al., *Phys. Med. Biol.*, 2022; 67.14: 145002.
RT-PHITS for nuclear medicine	Sato et al., EJNMMI Physics, 2021, 8: 4

ExPORT-PHITS	Sato et al., EJNMMI Physics, 2025. DOI: 10.1186/s40658-025-00743-6
FFT-based and TE-based SMK model phits-utility¥usranatal¥smk_*	Sato et al., Int. J. Radiat. Biol. 2021, 97, 1450-1460