PHITS

Ver. 3.24

User's Manual

序文

本マニュアルは、粒子・重イオン輸送計算コード PHITS (Particle and Heavy Ion Transport code System) の使用方法についてまとめたものです。PHITS は、あらゆる物質中での様々な放射線挙動を模擬するモンテカルロ計算コードです。放射線施設の設計、医学物理計算、放射線防護研究、宇宙線・地球惑星科学など、工学、医学、理学の様々な分野で国内外の研究者・技術者に利用されています。本コードに組み込まれた物理モデルや機能の詳細については、下記の文献[1] [2] やこれを引用する文献をご覧ください。本マニュアルでは、PHITS の実行方法やこれを使用する上で必要となるパラメータについて紹介します。

本マニュアルの内容は表紙に示す PHITS のバージョンに対応したものとなっています。また、予告なしにその内容が変更される場合があります。本マニュアルに関するお問い合わせは、PHITS 事務局 (phits-office@jaea.go.jp) までご連絡ください。

[1] 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] 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).

目 次

1 最近の改良点と開発体制

1.1 最近の改良点

以下に、PHITS バージョン 2.24 以降の主な改良点を記載します。

バージョン 3.24 では，主に次の改良とバグ修正を行いました。（2021 年 3 月 12 日）

- [t-dchain] の xyz メッシュ機能を使った PHITS → DCHAIN-PHITS → PHITS 接続計算による誘導放射能被ばく線量計算を可能としました。詳細は、"/phits/dchain-sp/sample/3-step_dose_xyz" をご参照ください。本開発は，原子力機構・システム計算科学センター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として RIST の三浦孝充氏の協力により実施しました。

- あるヒストリー中に発生する全粒子のカウンター値の最大値を使ってタリーに制限を掛けるヒストリーカウンター機能を [t-track] と [t-deposit] に導入しました。この機能を使えば、特定のイベントが発生した場合に線源発生まで遡ってタリーすることが可能となります。詳細は、"/phits/sample/misc/history_counter" フォルダをご参照ください。本改良は、RIST の古立直也博士のご協力により実施しました。

- 各バッチの計算結果を独立に集計して出力する機能 (itall=4) を [t-track] と [t-deposit] に導入し、バッチ毎に発生させる線源をコントロールする機能（[source] の ibatch パラメータ）を導入しました。これらの機能を組み合わせれば、一度の PHITS 実行で独立した複数の線源によるタリー結果を得ることができ、線源条件のみを変更して多数の計算が必要な場合にその計算時間を短縮することが可能となります。本改良は、RIST の古立直也博士のご協力により実施しました。

- エラー ID の代わりに、エラーが発生したソースコードのファイル名・サブルーチン名・行番号を出力する機能を導入しました。この機能は、[parameters] で ierrout=1 とすることにより有効となります。本開発は，原子力機構・システム計算科学センター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として伊巻正氏の協力により実施しました。

- multiplier サブセクションでパラメータや数式を使えるようにしました。ただし、括弧を使った数式は利用できません。

- ユーザー定義 anatally のサンプルとして、線量平均 LET を計算するモードと、標的核医学治療の生物学的線量を計算するモードを "/phits/utility/usranatal" フォルダに追加しました。

- JENDL Deuteron Reaction Data File 2020 (JENDL/DEU-2020) [3] の [frag data] ファイルを "/phits/XS/fragdata/" に追加しました。このデータファイルを使用することにより、$Li^{6,7}$, Be^{9}, $C^{12,13}$ 標的における重陽子入射反応の放出粒子を精度良く模擬できるようになります。本データファイル作成に関しては、日本原子力研究開発機構 核データ研究グループの中山梓介博士にご協力いただきました。当該ファイルを使用した成果を公開する際は論文[4] を引用してください。また、これを使用したサンプルインプットを "/phits/recommendation/NeutronSource" に追加しました。加速器中性子源の計算を模擬したものとなります。

- DICOM2PHITS を大幅に改良し、PET-CT データなどの読込も可能とした RadioTherapy package based on PHITS (RT-PHITS) を開発しました。詳細は、"/phits/utility/RTphits" をご参照ください。本開発は、量子科学技術研究開発機構の古場裕介博士、米内俊祐博士、松本真之介博士、張維珊博士、大阪大学の渡部直史博士、佐々木秀隆氏の協力により実施しました。

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

[4] 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., DOI:10.1080/00223131.2020.1870010.

- PHITS の入力ファイルを読み込んで 3 次元体系を可視化するソフトウェア PHIG-3D を開発しました。本開発は、海上技術安全研究所の大西世紀博士が開発した Gxsview [5] を元に、同博士の協力により実施しました。また、PHIG-3D の著作権は海上技術安全研究所が有しますが、日本原子力研究開発機構には使用、改造および第三者への再ライセンスが許諾されています。
- 宇宙線線源モード、`[frag data]`、電子磁場などで発生していたいくつかのバグを修正しました。また、Version 3.23 から発生していた OpenMP 及び複数 Lattice 読込に関する不具合を修正しました。ただし、Linux・OpenMP 版で dump 線源データを読み込む機能に関しては、まだ不具合が修正されていませんので、ご注意ください。
- Version 3.22 以降、メモリの使い方を変更したためユーザー定義タリーのサンプル (“/phits/utility/usrtally”）が使えなくなっていた不具合を修正しました。また、これまでご自身で作られたユーザー定義タリーが最新版ソースと併せてコンパイルできなくなっている可能性がありますので、その場合は、“/phits/utility/usrtally/readme-jp.docx” を参照の上、プログラムを修正してください。

バージョン 3.23 では，主に次の改良とバグ修正を行いました。（2021 年 1 月 12 日）

- 太陽活動、地磁気、周辺環境の影響を考慮して大気圏内や宇宙空間における宇宙線フラックスを再現する新たな線源モードを追加しました。詳しくは、5.3.19.9 節をご参照ください。
- 1 つもしくは複数のタリー結果に “usranatal.f” で書かれた演算処理を行って新たなタリー結果を出力する機能（ユーザー定義 anatally）を追加しました。また、そのサンプルとして、ホウ素中性子捕捉療法（BNCT）及び粒子線治療の生物学的線量を確率論的マイクロドジメトリモデル[6]に従って計算するプログラムを組み込みました。詳しくは、マニュアル 6.9.1 節及び `utility\usranatal` をご参照ください。
- 加速器線源 (`s-type=11`) でエネルギー分布を持つ線源 (`e-type`) を定義可能としました。
- 断面積出力モード (`icntl=1`) で光子反応断面積を出力できるようにしました。`ipnint=1,2` とすることで、光核反応の各成分を出力するように切り替わります。
- `[material]` で定義できる同位体数を 92 から 300 に変更しました。
- Importance や Weight Window で粒子が 1 度に 10 個以上に分割された際に warning を出すようにしました。
- `[t-dpa]` セクションにおいて、欠陥性成功率の補正を考慮した arc-dpa (`idpa=1`) を出力できるようにしました。詳細は論文[7]をご参照ください。
- xyz メッシュにおける物質の最大値を 10 から 100 に増加し、新規の入力パラメータを導入するなどの改良を DCHAIN-PHITS に行いました。詳細は `dchain-sp\manual` にあるマニュアルをご参照ください。また、新規の参考文献[8]が出版されましたので、DCHAIN-PHITS を用いた論文発表時にはこれを引用してください。
- OpenMP メモリ共有型並列計算の計算効率を改善するため、スレッドプライベート変数の削減等のコード改良を実施しました。入力ファイルに依って効果は異なりますが、OpenMP 実行時のオーバーヘッドが削減され、OpenMP の並列性能が向上しました。本改良は原子力機構・システム計算科学セ

[5] S. Ohnish, Gxsview: Geometry and cross section viewer for calculating radiation transport, SoftwareX, 14, 100681 (2021).
[6] T. Sato and Y. Furusawa, Radiat. Res. 178, 341-356 (2012).
[7] Y. Iwamoto, S. Meigo, and S. Hashimoto, J. Nucl. Mater. 538 (2020) 152261. `https://doi.org/10.1016/j.jnucmat.2020.152261`
[8] 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`

ンター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として日本ヒューレット・パッカード株式会社の白井信裕氏の協力により実施しました。

- 体系の Lattice 構造データのメモリ領域を動的確保するように改良しました。これにより、大容量のボクセルデータを使用する計算ではメモリ領域の拡張に伴う PHITS の再コンパイルが不要になりました。
- 2 つ以上の lattice 構造を使用した場合に計算が止まるバグを修正しました。このバグは、version3.22 のみ発生していました。
- データリスト型 (`typ=-1 & -2`) かつメッシュ幅が均等でない磁場マップ・電磁場マップを使用した際、電場・磁場の強さが正しく考慮できないバグを修正しました。

バージョン 3.22 では，主に次の改良とバグ修正を行いました。（2020 年 10 月 9 日）

- 加速器でのビーム・エミッタンスを定義できるように、位相空間線源 (`s-type=11`) でガウス分布した位相空間からサンプリング可能としました。また、関連マニュアルを整備しました（5.3.10 節参照）。本開発は、量子科学技術研究開発機構の榊泰直博士にご協力いただきました。
- 磁場マップで磁場を与えたときに電子以外の粒子に対する相対論が考慮できていなかったバグを修正しました。本バグ修正は、九州大学の徳永直輝氏にご協力いただきました。
- バージョン 3.21 で発生していた、`gshow` や `[t-gshow]` で領域番号が出力されないバグを修正しました。

バージョン 3.21 では，主に次の改良とバグ修正を行いました。（2020 年 9 月 14 日）

- ICRU90 に示された阻止能を使った計算機能を追加しました。具体的には、水・空気・炭素中における陽子及び炭素イオンの阻止能データベースを整備し `data\dedx` フォルダに格納しました。電子に関しては、`epstfl` パラメータで ICRU90 に対応した密度補正係数の利用の有無を選択できるようにしました。詳しくは `utility\icru90` をご参照下さい。本改良は、KEK 波戸芳仁博士及び産業技術総合研究所の清水森人博士にご協力いただきました。
- SCINFUL-QMD [9] に格納された断面積を用いて有機シンチレータの応答関数を正確に計算する SCINFUL-QMD モードを導入しました。詳しい使い方は `utility\usrtally\scinful-qmd` をご参照ください。本機能の開発は、日本原子力研究開発機構 放射線挙動解析研究グループの佐藤大樹博士にご協力いただきました。
- Adjoint mode を導入しました。これにより、広範囲に散らばる線源から小さな領域の光子フルエンスを効率的に計算することができます。詳しくはマニュアル `[t-adjoint]` 及び `utility\adjoint` をご参照ください。本機能の開発は、日本原子力研究開発機構　計算科学センターの Alex Malins 博士にご協力いただきました。
- 22 核種の評価済み高エネルギー核データライブラリ JENDL-4.0/HE を PHITS パッケージに格納しました。詳細は APENDIX をご参照ください。
- 長らく整備中だった `icntl=1` の一部機能（断面積・カーマファクター出力モード）を完成させ、その使い方をマニュアルに追記しました。詳しくは `utility\icntl1` をご参照ください。

[9] 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).

- サブアクチノイド (原子番号 89 未満) の原子核が統計崩壊を始めるとき、従来の核分裂モデルに代わって Iwamoto の核分裂モデルを使えるようにしました。本機能の開発は、原子力機構・原子力基礎工学研究センター・核変換システム開発 Gr. の岩元大樹博士の協力により実施しました。また現在は使われていませんが、今後のアクチノイド核分裂モデル改良を視野に、`phits\data` フォルダに核分裂バリアの表 `Fission_barrier_table_PhysRevC_91_024310.dat` [10]を導入しました。

- ガンマ崩壊モデル EBITEM を改良し、`phits\data\RIPL` フォルダに置いた RIPL(Reference Input Parameter Library)-3 [11] の核構造データを読むようにしました。従来は 2012 年頃の古い核構造データを、PHITS のソース内に取り込んで使っていました。これの改良により、計算精度の向上、実行可能ファイルの縮小、実行時のメモリ節約、コンパイルの高速化ができました。使用時には特に操作を必要とせず、自動的に新しいバージョンを使用します。

- XYZ メッシュ空間分布線源 (`s-type=22`) で、各メッシュから等しい数の粒子を発生させ、ウェイトを変化させてその強度を調整するモードを導入しました。

- DCHAIN-PHITS で統計誤差を出力できるようにしました。また、`[t-dchain]` タリーに連続四面体のメッシュ毎にタリー結果を出力する `mesh=tet` のオプションを追加しました。

- 系統的不確かさを評価する機能を `[t-deposit](output=dose)` でも利用できるようにしました。また、1 つのインプットファイルに複数のタリーセクションが書かれている場合も利用できるようになりました。他、分析機能用スクリプトファイル autorun のバグを修正しました。本開発は，原子力機構・システム計算科学センター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として RIST の三浦孝充氏の協力により実施しました。

- OpenMP メモリ共有型並列計算の計算効率を改善するため、タリー変数の非共有化を選択できるオプション `italsh` を追加しました。これまではタリー変数はスレッド間で共有することで、必要メモリ領域を少なくする仕様にしていたのですが、メモリ競合を防止する処理がボトルネックとなり、計算速度が低下する場合があるため、タリーの共有/非共有を選択できるようにしました。本開発は、原子力機構・システム計算科学センター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として日本ヒューレット・パッカード株式会社の吉廣保氏と白井信裕氏の協力により実施しました。

- 新しいパラメータ `gsline` を `[parameters]` セクションに導入し、各タリーの `gslat` パラメータを一括コントロールできるようにしました。また、`gslat` に新たなオプションを追加しました。

- `[cell]` セクション全体を一度バイナリ化して読込を高速化する `icells` パラメータを導入しました。

- ボクセルデータを使用する計算の出力データ量の抑制と計算時間の短縮のため、入力ファイルのエコー (“phits.out”) にボクセルデータを出力しないように改良しました。なお、`ivoxel=3` のオプションを `[parameters]` セクションで指定すると、今まで通り、入力ファイルのエコーにボクセルデータも出力します。

- 同一の物質番号を用いて密度が異なる領域を定義した場合に、各 cell の物質番号と密度をチェックするための新しいパラメータ `ichkmat` を導入しました。従来このチェックは自動的に行っていたのですが、cell 数が多い場合に計算時間が長くなるなどの不具合が発生したため、手動で設定するように変更しました。初期設定ではチェックしなくなるのでご注意ください。密度が異なる同一の物質番号の cell が特定の順番で定義された場合に、gshow でジオメトリが正しく描画できなくなります。

[10] 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.

[11] 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

- 陽子核データライブラリを使った場合に、反応断面積のエネルギー変化を考慮せずにステップ幅を決めていたバグを修正しました。これに伴い、ステップ幅中での最大断面積を与えるエネルギーを指定するパラメータ epseudo を導入しました。また、[data max] セクションを使って一部の核種のみに対してライブラリを使うように指定した場合に 2 次粒子が発生しなくなるバグも修正しました。本改良は、高度情報科学技術研究機構の古立直也博士のご協力により実施しました。

- 他に次のバグ修正を行いました。電子の電磁場中でのバグ（真空中で反応する問題や反射境界が効かない問題）を修正。電磁場マップで z 軸方向に掛かった磁場中の挙動を修正。

バージョン 3.20 では，主に次の改良とバグ修正を行いました。（2020 年 3 月 19 日）

- 陽子 (E < 300 MeV) 及び炭素イオン (E < 10 MeV/n) に対する飛跡構造解析コード KURBUC[12] を導入しました。本改良は、同コード開発者の Hooshang Nikjoo 教授と Thiansin Liamsuwan 博士の協力により実施しました。

- 光子によるミューオン対の生成を考慮できるようにしました。[parameters] セクションで igmuppd=1 と指定することにより本機能は動作します。本機能の開発は、KEK 坂木泰仁博士にご協力いただきました。

- 特定の領域における核反応生成粒子の統計精度を向上させるため [Repeated Collisions] セクションを導入しました。詳細は 5.22 節をご参照ください。本機能の開発は、原子力機構・原子力基礎工学研究センター・核変換システム開発 Gr. の岩元大樹博士の協力により実施しました。

- [t-cross] に、a-type mesh の角度基準を決めるパラメータ iangform を新規追加しました。従来の角度基準である通過した面の法線ベクトル（iangform=0）に加え、x 軸 (iangform=1)、y 軸 (iangform=2)、z 軸 (iangform=3) が選択できます。

- [frag data] に微分断面積を群データではなく点データで与える機能を追加しました。opt=5 で利用できます。

- 物質の阻止能をユーザーが与えるデータベースから直接読み込む機能を開発しました。また、そのサンプルデータとして、様々な物質に対する陽子及び α 粒子の阻止能を PSTAR 及び ASTAR[13] で計算した結果を "/data/dedx" フォルダに格納しました。この開発は、（株）ナイスの協力により実施しました。

- 電場マップ読み込み機能を追加しました。この開発は、（株）アドバンスソフトの協力により実施しました。

- Xorshift64 方式に基づく新しい擬似乱数生成法を導入し、初期設定として利用するようにしました。この改良により、乱数周期が 2^{46} から $2^{64}-1$ に大幅に延長されます。

- ボクセルやテトラによる複雑形状を gshow できれいに描けるように修正しました。この開発は、（株）ナイスの協力により実施しました。

[12] 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

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

- DCHAIN-PHITS 用の新たな中性子放射化断面積ライブラリをいくつか追加し、それらを組み合わせたライブラリを初期設定で使用するようにしました。また、初期設定で itdecs 及び itdecn パラメータを 1 として、核反応によるターゲット減少分を考慮できるようにしました。詳細は、7.15 節をご参照ください。本機能の開発は、原子力機構・原子力基礎工学研究センター・炉物理標準コード研究 Gr. の今野力研究主幹の協力により実施しました。

- ATIMA に入力するエネルギー単位が原子質量単位あたりの MeV/u ではなく核子あたりの MeV/n だった不具合を修正しました。この修正により、荷電粒子の飛程が多少（最大で 1%程度）変化しますのでご注意ください。また、PHITS の入力ファイルで指定するエネルギーは MeV もしくは MeV/n ですので、マニュアルや Input echo の MeV/u と記載していた個所は、全て MeV/n に修正しました。

- Maxwellian 分布を持つ線源 (e-type=3,7) のエネルギーのベキを指定できるようにし、その初期値を 0.5 としました。従来は 1.0 に固定されていましたが、正しい Maxwellian 分布を再現できていませんでしたので、ご注意ください。

- 分散分析に基づいて PHITS の計算結果における系統的不確かさを評価する機能を追加しました。また、自動的に PHITS の輸送計算を実行して分散分析の結果を得られるようにするために、分析機能用スクリプトファイルとして、autorun を一新しました。詳細は 6.9.2 節をご参照ください。本開発は，原子力機構・システム計算科学センター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として RIST の三浦孝充氏の協力により実施しました。

- PHITS-FLUENT の接続計算についての資料を作成しました。“/phits/utility/FLUENT” フォルダをご確認ください。

バージョン 3.17 では，主に次の改良とバグ修正を行いました。（2019 年 10 月 29 日）

- 連続四面体形式でタリー結果を出力する際，csv 形式（コンマ区切り）で出力する機能 (foamout=2) を追加しました。

- EGS5 で [t-deposit] を使った場合に初期化されていない変数があり，場合によっては結果が極端に変になるバグを修正しました。

- 各タリーの sum over の計算方法を修正しました。

- nedisp=1 としたときのカットオフ粒子の扱いを修正しました。

- “usrdfn2.f” で計算する生物学的線量のパラメータが T.Sato et al. Radiat. Prot. Dosim. 143, 491-496 (2010) に基づいていましたが，T.Sato et al. Radiat. Res. 171, 107-117 (2009) に基づくように修正しました。

バージョン 3.16 では、sangel に関するバグとバージョン 3.15 で発生していた [t-track] で “_err” のファイルが作成されないバグを修正しました。（2019 年 9 月 26 日）

バージョン 3.15 では、主に次のバグ修正を行いました。（2019 年 9 月 12 日）

- EGS5 と [t-deposit] を使用した場合に、エネルギーカットオフの成分を過大評価するバグを修正しました。このバグはバージョン 3.13 と 3.14 で発生していました。また、バージョン 3.10 以降で発生していた EGS5 と lattice 構造を組み合わせて使用した場合のバグも修正しました。

- itall=3 により利用できるタリーの評価値及び統計誤差の推移をバッチ毎に表示する機能が、入力条件により動作しないバグを修正しました。このバグはバージョン 3.13 と 3.14 で発生していました。

- バージョン 3.14 で発生していた DCHAIN のバグを修正しました。

バージョン 3.14 では、主に次の改良とバグ修正を行いました。(2019 年 8 月 18 日)

- DCHAIN-SP を改良し，DCHAIN-PHITS としました。主な改良点は，中性子放射化断面積ライブラリ及び崩壊データライブラリを最新のデータに基づいて整備した点と， PHITS の統計誤差の伝播を考慮した誘導放射能計算を可能とした点です。これに伴い [t-dchain] にライブラリや誤差計算の有無を選択する新たなパラメータが加わりました。詳しくは 7.15 節をご参照ください。

- 飛跡構造解析モードとユーザー定義タリーを組み合わせた DNA 損傷計算機能を開発し，そのサンプルデータを "/phits/utility/usrtally/DNAdamage" フォルダに格納しました。

- PHITS バージョン 3.13 固有のバグ（[t-interact] の一部機能が動作しない，電子飛跡構造解析モードで 2 次電子が生成されない問題）を修正しました。

バージョン 3.13 では、主に次の改良とバグ修正を行いました。(2019 年 8 月 2 日)

- 陽子，パイオン，ミューオン，イオンに対するカットオフエネルギーの初期値を 1 MeV/n から 1keV/n に引き下げました。

- [surface] セクションで定義した特定の面上から線源を発生させる新たな線源タイプ (s-type=26) を導入しました。詳しくは，5.3.16 節をご参照下さい。

- Tally の評価値及び統計誤差の推移をバッチ毎に表示する機能（itall=3）を導入しました。ただし，この機能は現在のところ [t-track] と [t-point] のみ動作します。

- stdcut で定義した統計目標に達したタリーを順次，計算から除外するパラメータ istdcut を導入しました。[t-point] や [t-sed] など，タリー計算そのものに時間が掛かる場合に有用となります。また，計算開始直後に統計誤差がばらつくことがあるため，指定したバッチまでは stdcut による打ち切り機能を無効にするパラメータ istdbat を導入しました。

- 阻止能計算に ATIMA を採用した場合，水 (H_2O) に対する Ionization Potential は初期値として 75 eV が選択されるように設定されているのですが，従来は，この設定が水素の同位体 1H を明示的に指定した場合にのみ有効で，質量数を定義せずに水素元素 H を指定した場合には無効（69 eV が選択される）となっていました。そこで，H 2 O 1 や H -0.1111 O -0.8889 のように質量数を指定しなくても水素と酸素の元素比が約 2:1 で構成される物質は水と認識するよう判定基準を修正しました。この修正により，version 3.13 以降，質量数を指定せずに定義した水中での荷電粒子の飛跡長が微妙（ほとんどの場合は 1%未満）に長くなりますのでご注意下さい。また，水の Ionization Potential を自由に設定できるパラメータ ih2o に関する判定基準も同様に修正しました。

- MPI 並列 version で統計誤差打ち切り機能（stdcut）が動作しないバグ，Universe 構造に反射面が存在した場合に正しく考慮されないバグ，EGS5 を使った場合に希に前のヒストリーの影響を受けてしまうバグなど，いくつかバグを修正しました。

バージョン 3.12 では、主に次の改良とバグ修正を行いました。(2019 年 6 月 19 日)

- [source] セクションでマルチソース毎に異なるカウンター値を設定できるようにしました。また，定義可能なマルチソースの数を 200 から 500 に拡張しました。

- [t-dchain] で核変換による放射性同位元素の減少分を考慮可能としました。この変更に伴い DCHAIN も修正しました。古い version の DCHAIN では新しい [t-dchain] の出力結果を読み込めませんのでご注意下さい。

- Windows の OpenMP 版を用いた場合でも itall=1 オプション（バッチ毎に eps ファイルを出力する機能）が有効となるようにしました。

- Fluent との連成計算で密度情報が正しく受け渡しできない問題など，いくつかバグ修正を行いました。

バージョン 3.11 では、主に次のバグ修正を行いました。（2019 年 5 月 16 日）

- 四面体体系を使った輸送計算を MPI 並列で実行しようとするとエラーが起きて終了するバグを修正しました。このバグは PHITS ver.3.10 のみに存在するバグです。
- 磁場マップ読込機能で，`r-z` グリッド型 (`type = -2, -4`) の読込の際に正しく磁場が読み込まれない場合があり，これを修正しました。
- RI 線源機能で，陽電子を放出する核種で陽電子と光子放出を両方考慮した場合，消滅 γ 線が 2 重カウントされてしまうことが分かりました。これを回避するために，新しいパラメータとして `iannih` を導入し，光子線源では消滅 γ 線を考慮しないオプションを作成しました。

バージョン 3.10 では、主に次の改良とバグ修正を行いました。（2019 年 3 月 13 日）

- 熱流体解析コード（ANSYS Fluent など）との連成計算を可能としました。この目的のため，PHITS に読み込める連続四面体形状の形式に NASTRAN のバルクデータ形式を追加しました。この形式を利用することで、熱流体解析コード等で用意した連続四面体形状を PHITS でも共通して使用することができます。また、新たに連続四面体のメッシュ毎にタリー結果を出力する `mesh=tet` オプションを `[t-track]`, `[t-deposit]` , `[t-yield]`, `[t-product]`, `[t-dpa]` に導入しました。その際，OpenFoam のフィールドデータ形式で出力する機能も追加し，PHITS の計算結果を熱流体解析コードなどで直接読み込めるようにしました。この開発は，原子力機構 原子力基礎工学研究センター 燃料・材料工学 Dv のサポートのもと，伊藤忠テクノソリューションズ株式会社（CTC）の協力により実施しました。
- 上記目的の一部として `[t-deposit]` に `unit=5` を追加し，J/m^3/source 単位での出力機能を追加しました。
- 高エネルギー核データライブラリ JENDL-4.0/HE を正しく読み込めるよう改良し，いくつかサンプルを PHITS パッケージに入れました。詳しくは `\phits\recommendation\jendlHE` をご参照ください。
- xyz もしくは r-z グリッド形式で表現される複雑な磁場構造を考慮可能としました。本改良は,（株）先端力学シミュレーション研究所 (ASTOM R&D) の協力により実施しました。
- `[t-dchain]` で `mesh=xyz` が指定可能となりました。これに伴い，DCHAIN も仕様が変更されていますので，ご注意ください。なお，DCHAIN で計算した残留放射能の空間分布を可視化する機能や，その結果を線源として PHITS に取り込む機能は，まだ完成しておりません。本開発は，原子力機構・システム計算科学センター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として RIST の三浦孝充氏の協力により実施しました。
- e, μ, τ ニュートリノとその反粒子入射による反応のモデルを、入射エネルギー 150MeV まで組み込みました。ただし，標的は軌道電子, ^{1}H, ^{2}H のみ可能で、^{2}H より重い原子核との相互作用は未実装です。
- 九州大学で開発中の核反応モデル INC-ELF を改良し，標的核の集団励起による散乱や、高精度な荷電粒子放出バリア計算が可能としました。本改良は，九州大学 魚住裕介准教授及び渡辺岳氏の協力により実施しました。
- 電子飛跡構造解析モードで Auger 電子の生成を考慮することにより，その計算精度を向上させました。本改良は，原子力機構 松谷悠佑博士研究員の協力により実施しました。

- [T-deposit] で dresol や dfano だけでは表せない分解能を再現するために、ユーザー定義検出器分解能機能を使えるようにしました。その関数は “usresol.f” で定義でき、サンプルとして CdZnTe の非対称ピークを再現するために SCK-CEN の Meleshenkovskii 氏らが開発したモデルがデフォルトで組み込まれています。
- 極めて細かい構造を持つジオメトリ（ボクセルファントムや連続四面体形状など）を [t-gshow] や gshow できれいに描画するため，ピクセル状に幾何形状を出力する新たなオプション（gshow=5）を追加しました。
- 短縮形式の Lattice 構造データに対応しました。ボクセルファントムのデータなど、同じ物質が連続して並んでいる場合，その個数を負の値で指定することによりファイルサイズが大幅に縮小され，読込時間を短縮することができます。
- 負ミューオン捕獲反応の調整オプションを導入しました。
- [t-deposit] で付与エネルギーに重み付けする 3 つ目の関数を導入しました。その関数は “usrdfn3.f” で定義でき，サンプルとして医学物理計算で重要となる水等価線量を評価するモデルがデフォルトで組み込まれています。
- “batch.out” の仕様を変更し，PHITS 実行中に残りバッチ数を調整可能としました。
- [t-cross] で mesh=xyz もしくは r-z の場合，メッシュで区切られた閉曲面を 1 つの横断面として設定することにより，各メッシュ内に流入もしくは流出する粒子数（もしくはフラックス）を計算可能としました。
- [t-deposit] で output=deposit として付与エネルギー確率分布を計算する場合に，異なる Lattice 座標内にある同一セル番号への付与エネルギーを別個に扱う機能を導入しました。
- 最新版 ENDF など γ 線スペクトルデータが 1000 以上ある核データライブラリの読込を可能としました。
- ifixchg パラメータを導入し，ATIMA で電荷を指定して阻止能を計算可能としました。ただし，電荷交換反応は未実装のため，現在のところ，極めて薄いターゲット以外は物理的に正しい阻止能が計算できませんのでご注意ください。
- emin(12) を指定して emin(13) を指定しなかった場合の陽電子挙動の不具合を修正しました。Lattice 構造中で電子に対して電磁場が効かない問題を修正しました。それ以外にもいくつかマイナーなバグを修正しました。

バージョン 3.08 では、主に次の改良とバグ修正を行いました。(2018 年 8 月 20 日)

- [t-yield] 及び [t-dchain] でユーザーが独自に指定する放射化断面積を読み込む機能を追加しました。詳しくは，7.8 節をご参照ください。本改良は，革新的研究開発推進プログラム ImPACT「核変換による高レベル放射性廃棄物の大幅な低減・資源化」の一環として実施されました。
- RI 線源機能で，特性 X 線と内部転換電子を考慮可能としました。
- CAD から変換されたデータを使用するため，連続四面体形状で境界面が一致しないデータでも使用可能としました。
- ユーザー定義ソース (“usrsors.f”) 内で，インプットファイルで指定したパラメータ（c1-c99）を使えるようにしました。

- ANGEL で結果を描画する際，数値データのチェックを行わずに高速で描画するモードをデフォルトとしました。
- 円柱分布線源 (s-type=1) で r0=r1 とすると，線源位置が 0 から r1 の範囲でサンプリングされてしまう不具合など，いくつかのバグを修正。

バージョン 3.07 では、[weight window] の xyz メッシュを EGS5 を使用した場合でも指定できるように改良しました。また、半導体ソフトエラー発生率を計算するための換算係数を [multiplier] のデフォルトデータとして追加しました。他、JAM を使用した際に稀に発生するバグを修正しました。（2018 年 7 月 5 日）

バージョン 3.06 では、主に次の改良とバグ修正を行いました。（2018 年 5 月 29 日）

- igamma パラメータを負値にすることにより、励起核の運動によるドップラー効果を無視し、放出される特性ガンマ線スペクトルが単色ピークとなる機能を追加しました。
- 核子とガンマ線放出の競合を考慮できる統計崩壊モデル GEM Ver.2 を導入しました。[parameters] において ngem=2 と指定することにより利用できます。
- [transform] で、z,y,x 軸周りの回転を簡易に定義できる形式を導入しました。
- Windows PC において、MPI プロトコルを用いたメモリ分散型並列計算が可能となりました。この開発は、（株）ナイスの協力により実施いたしました。
- e-mode=2 を指定した状態で、INCL などの核反応モデルが動作した場合に発生するバグを修正しました。条件により、1MeV 以下の中性子フルエンスが数 10%程度減少していました。このバグはバージョン 3.05 で発生しておりました。

バージョン 3.05 では、主に次の改良とバグ修正を行いました。（2018 年 3 月 14 日）

- 従来 [t-heat] でしか行えなかったカーマ近似による付与エネルギー計算機能を [t-deposit] にも追加しました。その際、カーマ近似の使用の有無は、イベントジェネレータモードや EGS5 の使用状況から自動的に判断します。この改良により、付与エネルギー計算は全て [t-deposit] で行えるようになりました。
- 1 ヒストリー内でエネルギー付与する粒子のウエイトが同じ場合に、[t-deposit] の output=deposit で正しく計算できるようになりました。この改良により、s-type=9 かつ dir=-all とした場合の線源や非荷電粒子に [forced collisions] を用いた際の付与エネルギー分布が適切に評価できるようになります。
- [t-cross] において axis=z とした場合に、出力されるグラフの形式を折れ線グラフとしました。
- [t-cross] で z-type=1 とした場合に、nz=0 とすることで、タリー面を 1 面だけ設定できるようにしました。
- [t-interact] において、電子と陽電子による knock on electron 生成の成分を output=deltaray から切り分け、output=knockelec で出力するように変更しました。また、同成分を [counter] における delt から切り分け、knoe でカウントするように変更しました。
- RI 線源機能を使用して Auger 電子を発生させる場合に、RIsource.ack にデータがない RI 核種については、RIsource.rad のデータを利用するように改良しました。
- iMeVperu=1 とした場合に、[t-product] でエネルギーが正しく計算されないバグを修正しました。

バージョン 3.04 では、[t-star] でヒストリ毎の反応頻度分布を出力可能としました。また、飛跡構造解析モードで起きた反応数も計算可能としました。さらに、反応数 (interaction) を計算するタリーであることが直感的に分かるように、タリー名を [t-interact] に変更しました。ただし、プログラム上は [t-star] という名称も引き続き利用することが可能です。(2018 年 2 月 16 日)

バージョン 3.03 では、主に次の改良とバグ修正を行いました。(2018 年 2 月 1 日)

- 連続四面体から線源粒子を発生させる機能を追加しました。s-type=24 とすることにより利用できます。詳細は、5.3.15 節をご覧ください。
- [weight window] 及び [t-wwg] において xyz メッシュ形状が選択できるようになりました。
- [multiplier] において part を指定できるようにし、粒子別の係数を定義できるようにしました。また、この機能を応用して、様々な種類の被ばく線量をあらかじめデータベース化した換算係数を用いて計算できるようにしました。
- [t-deposit] で output=deposit とし、dresol を設定した場合に part=all 以外の結果が適切に計算されないバグを修正しました。
- iMeVperu=1 とした場合に、[t-track] 等で荷電粒子のレンジ計算が適切に行われないバグを修正しました。

バージョン 3.02 では、主に次の改良とバグ修正を行いました。(2017 年 12 月 1 日)

- ユーザー定義タリー [t-userdefined] で指定できる新しいパラメータ nudtvar と udtvar(*i*) を追加しました。*i* =nudtvar を上限とする udtvar(*i*) の数値をインプットファイルで指定し、それらを subroutine usrtally 内で使用できます。udtvar(*i*) は、従来の udtpara と同様の役割を果たしますが、指定できる数 (nudtvar) に制限はありません。なお、udtpara は本バージョン以降も使用可能です。
- [t-let] と [t-sed] に対して新たな unit を追加しました。
- 球殻線源(s-type=9)で dir=iso とした場合、従来は球殻の外側から線源を発生させていましたが、球殻上から線源を発生するように変更しました。この変更は、線源球殻の外側に物質がある場合のみ結果に影響を与えます。
- EGS5 を利用せず光子のみの輸送を行った場合に、[counter] で原子相互作用 (atom) のプロセスがカウントされない不具合を修正しました。
- [delta ray] セクションを使ってデルタ線を発生させる際の制限付阻止能計算のバグを修正しました。

- 荷電粒子を輸送する際の角度分散に関するバグを修正しました。emin を 1MeV/n より低く設定し、nspred を有効にした場合に発生していました。SOBP による粒子線治療を模擬した計算において影響がありました。このバグは、version 2.96 から含まれておりました(それ以前の version は問題ありません)。

バージョン 3.01 では、主に次の改良とバグ修正を行いました。(2017 年 10 月 31 日)

- 任意の ANGEL パラメータを指定できる sangel パラメータを追加しました。 本機能の開発は、原子力機構・システム計算科学センター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として RIST の三浦孝充氏に実施していただきました。

- 連続四面体データの読み込みルーチンの高速化を行いました。
- ANGEL を高速化しました。これにより、axis=xy などの 2 次元表示の際に、eps ファイルの作成時間が短縮されました。
- [t-deposit] において、stdcut の機能が使用できない不具合を修正しました。

バージョン 3.00 では、原子核・原子核間の反応断面積モデルのデフォルトを Kurotama モデルに変更し、[t-dchain] によって DCHAIN のインプットを作成する際に天然存在比による展開を考慮できるよう改良しました。また、幾つかの微小なバグを修正しました。（2017 年 10 月 4 日）

バージョン 2.97 では、主に次の改良とバグ修正を行いました。（2017 年 9 月 21 日）

- タリー出力において、原子核のエネルギーの単位を MeV から MeV/n（核子あたりの MeV）に変換するパラメータ iMeVperu を導入しました。[parameters] において iMeVperu=1 とすると、設定した全てのタリー結果が自動的に MeV/n の単位の量に変換されます。 本機能の開発は、原子力機構・システム計算科学センター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として RIST の三浦孝充氏に実施していただきました。
- [t-cross] で mesh=reg とした場合のパラメータとして、r-from, r-to を導入しました。これらはそれぞれ r-in, r-out と同様の役割を果たします。従来は、出て行く領域 (r-in) と入ってくる領域 (r-out) の in と out が逆でわかりにくかったため、より直感的にわかりやすいパラメータを追加しました。
- 従来 U($Z = 92$) までしか使用できなかった、阻止能計算モデルのオプション ndedx=3 の標的元素に関する適用範囲を Bk($Z = 97$) まで拡張しました。詳細は 5.2.5 節をご覧ください。
- PHITS のインプットファイルにおいて、最初のセクションの前に$OMP=*N* （*N* は使用する CPU コア数）を加えることで、メモリ共有型の並列計算 (OpenMP) を実行できるようにしました。また、infl: コマンドを使う場合に、インプットファイルの最初に file=（インプットファイル名）を書く必要がなくなりました。他、MacOS において、Dock によるドラッグ&ドロップを用いた場合、ターミナルを立ち上げて PHITS を実行するようにしました。ただし、Linux などで実行ファイルをコマンドラインで実行する場合は、これらの機能は使用できませんのでご注意ください。
- 原子核-原子核の反応モデル JQMD と JAMQMD の切り替えエネルギー ejamqmd のデフォルト値を 3GeV に変更しました。

バージョン 2.96 では、主に次の改良とバグ修正を行いました。（2017 年 8 月 28 日）

- 新しいセクション [ww bias] と新しいタリー [t-wwbg] を導入し、特定の方向に粒子を多く誘導するよう [weight window] にバイアスをかけてシミュレーションすることが可能となりました（5.20 節及び 7.17 節参照）。
- itall=0（初期設定値）の場合でも、各バッチ終了時にタリー結果の数値データを出力するように変更しました。この変更により、itall=1 と 0 の違いは、EPS ファイルの出力の有無のみになります。また、統計誤差に基づいて計算を打ち切る機能（stdcut）が初期設定でも有効となりました。
- icntl に依存して不要となるタリーは自動的に “off” にするようにしました。例えば、icntl=14 の場合は [t-volume] 以外のタリーは全て off になり、逆に icntl≠14 の場合は [t-volume] は off になります。また、off されたセクション内に set や infl コマンドがある場合は、warning を出力するようにしました。

- 空間分布や時間分布を出力する際、軸の単位を調整する ANGEL パラメータを導入しました。極めて小さい領域や大きい領域の結果を可視化したい場合に有効となります。詳しくは 6.7.10 節をご参照下さい。
- `[parameters]` セクションが 2 つ以上定義されている場合にエラーを出力するようにしました。
- 光核反応（巨大共鳴）断面積の計算系統式、飛程の極めて短い粒子に対する角度分散、高エネルギー重イオン核反応モデル JAMQMD2、Maxwell 分布線源スペクトル計算式などに含まれていたバグを修正しました。

バージョン 2.95 では、`[source]` セクションにおいて、エネルギー分布の有無で `s-type` を変える仕様を変更しました。線源が単色かエネルギー分布をもつかは、`e0` と `e-type` のどちらが定義されているかによって判断します。ただし、両方が定義されている場合は、従来の `s-type` による指定が優先されますので、ご注意ください。また、幾つかの微小なバグを修正しました。（2017 年 7 月 21 日）

バージョン 2.94 では、EGS5 で `[t-track]` や `[t-cross]` を用いて真空中の電子フラックスを計算する場合に、本来とは異なるエネルギービンにスコアされてしまう不具合を修正しました。なお、本バグ修正はタリーのみの問題で、粒子輸送自体には影響は与えません。また、飛跡構造解析モードに関するいくつかのバグを修正しました。（2017 年 6 月 30 日）

バージョン 2.93 では主に次の改良とバグ修正を行いました。（2017 年 6 月 16 日）

- インストールフォルダを指定する `file(1)` パラメータを導入することにより、核データライブラリファイル名 (`xsdir.jnd`) など各入力ファイル名を指定する必要がなくなりました。ただし、各ファイル名を直接指定した場合はそちらが優先されるので、基本的には従来インプットに変更の必要はありません。
- 核データを自動で設定するパラメータ `nucdata` を導入することにより、`emin` や `dmax` を指定することなく核データを利用可能としました。ただし、`emin` や `dmax` を直接指定した場合はそちらが優先されるので、基本的には従来インプットに変更の必要はありません。また、そのデフォルト値を 1 とすることにより、初期設定で中性子の挙動が正しく模擬できるようになりました。
- `negs` パラメータに新たなオプション（-1）を導入し、`negs = -1` とした場合は、従来 PHITS アルゴリズムを用いて光子のみ輸送する設定に自動調整するようにしました。これに伴って、従来は `emin` や `dmax` と `negs` パラメータを組み合わせて光子・電子・陽電子の輸送を調整していましたが、今後は、遮蔽計算などで光子のみ輸送する場合は `negs = -1` （デフォルト）、医学物理計算などで電子・陽電子の輸送も必要な場合は `negs = 1` と設定すれば、それ以外の調整は不要となります。ただし、`emin` や `dmax` を直接指定した場合はそちらが優先されるので、基本的には従来インプットに変更の必要はありません。
- `ides` パラメータのデフォルト値を 1 に変更し、初期設定で従来 PHITS アルゴリズムを使って光子輸送を行う場合は、電子・陽電子は発生しないようにしました。電子・陽電子を輸送する際は、EGS5 をお使いください。
- `igamma` パラメータの初期設定値を 2 に変更し、初期設定で核反応の脱励起から生じる光子を考慮するようにしました。
- RI 線源機能に β 線（オージェ電子含む）及び α 線放出機能を追加しました（表 67 を参照）。
- 低エネルギー電子・陽電子が引き起こす電離・励起・振動などの個々のイベントを正確に模擬する飛跡構造解析モードを追加しました（5.13 節参照）。

- δ 線放出に伴う荷電粒子の阻止能変化を、実際に放出された δ 線エネルギーの積分値ではなく制限付阻止能から自動で調整する機能 (irlet=1) を加え、そちらをデフォルトとしました。
- 相対論効果を考慮できるように 3GeV/u 以上の重イオンに対する核反応シミュレーター JAMQMD を改良し、高エネルギー重イオン核反応の再現性を向上しました。
- [t-dpa] を改良し、電子・陽電子・パイオンなどによる DPA を計算可能としました。
- 1 つの [counter] セクションで 2 つ以上のカウンターを定義した際に生じるバグを修正しました。このバグは、version 2.90 から発生していました（それ以前の version は問題ありません)。
- 自発核分裂からの中性子線源 (ispfs) で規格化が間違っていたバグを修正しました。

バージョン 2.92 では、[t-dchain] におけるデフォルト設定を output=cutoff に変更し、指定した領域で止まった粒子をタリーするようにしました。従来の output=product では、薄膜等で核反応により発生した重イオンがその領域に止まっていなくてもタリーされていたのでご注意ください。また、eps ファイルにおいて、PHITS と ANGEL のバージョン番号の出力場所を入れ替えました。他、三角柱形状線源で一様分布した際に発生するバグなど幾つかの微小なバグを修正しました。（2017 年 4 月 18 日）

バージョン 2.91 では主に次の改良とバグ修正を行いました。（2017 年 2 月 20 日）

- タリー結果を図示した際に、その値の統計的不定性を誤差棒により表示する機能を追加しました。各タリーセクションにおいて、epsout=2 とすることで動作します。ただし、axis=xy などの 2 次元表示や特殊な表示形式の場合は機能しません。
- EGS5 を使用した状態で、[t-cross] により電子や陽電子をタリーした場合にこれらの粒子のエネルギーが増加するバグを修正しました。

バージョン 2.90 では主に次の改良を行いました。（2017 年 2 月 9 日）

- 電子・陽電子の電磁場中での挙動解析が可能となりました。この改良は、（株）ナイスの協力により実施いたしました。
- Version 2.77 から導入した角度分散の大きさを考慮するパラメータ ascat2 のデフォルト値を、文献と一致するよう 0.088 から 0.038 に修正しました。nspred=2 として ascat2 をデフォルト値で計算していた方は、結果が変わりますのでご注意ください。
- xyz メッシュ空間分布線源発生機能を加えました。この機能を用いれば、空間的に複雑に分布した線源を表現することができます。詳細は、5.3.14 節をご参照ください。
- 指定した領域の体積をモンテカルロ積分法で自動計算するタリー [t-volume] を加えました。詳細は、7.18 節をご参照ください。
- バッチ終了時に CPU 時間をチェックし、計算開始からの合計 CPU 時間があらかじめ設定した値（timeout）を超えた場合に計算を打ち切る機能を加えました。
- バッチ終了時に各タリー結果の統計誤差をチェックし、その値が各タリーに対してあらかじめ設定した値 (stdcut）以下になった場合は計算を打ち切る機能を加えました。
- [t-product], [t-star], [counter] で核反応と原子反応を区別するようにしました。また、より詳細なチャンネル毎（例えば制動放射放出など）の情報も出力可能としました。
- JENDL に格納されていない元素（例えばレニウムなど）に対しても、電子・陽電子・光子のみの輸送であれば計算可能としました。

バージョン 2.89 では主に次の改良を行いました。（2017 年 1 月 11 日）

- `[material]` において、デフォルト設定では `c` をコメント文字として使用できないようにしました。炭素の元素記号 C がコメント文字として抜け落ちてしまう間違いを避けるためです。コメント文字として使用したい場合は、`[parameters]` において `icommat=1` としてください。
- `[t-deposit]` において、複数領域における deposit energy の総和を求める際に、条件に応じて重み付けを行う機能を追加しました。本機能は、特定領域への deposit energy に応じて電荷収集効率が変化するような、半導体ソフトエラー解析などに利用できます。
- `axis=eng` として求めたタリーの結果を `[source]` のエネルギー分布として設定できる機能を追加しました。`e-type=20` と指定することで利用できます。
- `[t-deposit]` でユーザー定義関数 2(`usrdfn2.f`) を使った場合に、Microdosimetric Kinetic Model に基づいて生物学的線量を導出するように変更しました。パラメータなどの詳細は、下記の文献[14] をご参照下さい。

バージョン 2.88 では主に次の改良を行いました。（2016 年 9 月 29 日）

- `[t-dchain]` でも sumtally 機能が使えるようになりました。
- ユーザー指定断面積の読み込み機能を拡張し、データが与えられていないエネルギー・角度領域に外挿を行って補間するオプションと微分断面積を与えない場合に核反応モデルにより核反応イベントを計算するオプションを追加しました。
- 計算打切エネルギーにより輸送が止められた中性子が全て崩壊してしまうバグを修正しました。このバグはバージョン 2.83 以降で発生していましたのでご注意ください。

バージョン 2.87 では主に次の改良を行いました。（2016 年 9 月 16 日）

- `[t-3dshow]` で xyz 軸を出力可能としました。
- 連続四面体形状の読込方法を改良し、計算を高速化しました。
- ジオメトリエラー出力ファイル名を “*.err” から “*_geo.out” に変更しました。
- 物質の定義で MTx（$S(\alpha,\beta)$ テーブル）を使用した際のバグ（バージョン 2.86 のみで発生）を修正しました。

バージョン 2.86 では主に次の改良を行いました。（2016 年 8 月 23 日）

- 新しいタリー `[t-wwg]` を導入しました。このタリーを使えば、Weight Window 機能が効果的に動作するパラメータ設定を自動で得ることができます。詳しくは、7.16 節をご参照下さい。
- 3 次元可視化ソフトウェア ParaView 用の出力機能を作成しました。この機能を使えば、PHITS の計算結果を ParaView 上に 3 次元的に可視化することが可能です。使い方は、utility フォルダにある ParaView をご参照下さい。また、2 次元のタリー計算結果やジオメトリをビットマップ形式で出力する機能を加えました。これらの改良は、原子力機構原子力センシング研究グループの古高和禎氏および（株）V.I.C. の協力により実施いたしました。
- 荷電粒子の阻止能を全て ATIMA で計算するモード（`ndedx = 3`）を新たに加えデフォルト設定としました。

[14] 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).

- 現在のバッチ情報を出力するファイルの名前を `batch.now` から `batch.out` に変更し、`[parameters]` セクションにおいて `file(22)` により指定できるようにしました。

- RI 線源発生機能を追加しました。この機能を使えば、放射性核種とその放射能を指定することにより、放射性核種の崩壊に伴って放出される粒子の線スペクトルを自動的に定義することができます。ただし、崩壊ガンマ線のみ扱えます。崩壊ガンマ線の線スペクトル計算には、放射性核種崩壊データベース DECDC [15] を使用します。このデータベースは ICRP107 と同等です。詳しくは表 67 をご参照ください。なお、本改良は、原子力機構原子力基礎工学研究センターの遠藤章氏にご協力いただきました。

- [material] セクションで質量数を定義せずに元素を指定した際、天然存在比に従って自動で各同位体に展開する機能を作成しました。ただし、JENDL-4.0 に含まれない核種に対しては展開されませんのでご注意下さい。本改良は、原子力機構・システム計算科学センターにご協力いただきました。

- マテリアルの核種ごとに核データライブラリー利用の上限エネルギーを設定できる [data max] セクションを追加しました。詳しくは、5.16 節をご参照下さい。本改良は、原子力機構・システム計算科学センターにご協力いただきました。

- ミューオン核反応モデルを改良しました。詳しくは、文献[16] をご参照下さい。

- 重陽子の全反応断面積を精度良く記述する新規の模型を導入しました。`[parameters]` セクションにおいて `icrdm=1` とすることで利用できます。詳細は文献[17] をご覧ください。

- パイオンの全反応断面積を改訂し、それをデフォルトで使用するようにしました。従来の幾何学的な断面積と比べて、入射エネルギー依存性を考慮しており、デルタ共鳴によるピーク構造を再現できるようになりました。従来の幾何学的断面積を使用する場合は、`[parameters]` セクションにおいて、`icxspi=0` としてください。

- 1 つのインプットファイルで複数の sumtally サブセクションを定義できるようにしました。また、sumtally に関するいくつかのバグを修正しました。

バージョン 2.85 では主に次の改良を行いました。（2016 年 5 月 16 日）

- 3GeV/u 以上の重イオン核反応で使われる JAMQMD モデルを改良し、宇宙線輸送計算などの精度及び安定性を高めました。従来モデルでは、まれにメモリ違反で強制終了されてしまう可能性がありました。

- 阻止能計算モデル ATIMA のアルゴリズムを改良し、高速化しました。この改良により、精度の高い ATIMA を用いた PHITS シミュレーションが、SPAR を用いた場合とほぼ同程度の計算時間で実行できるようになりました。本改良は、原子力機構・システム計算科学センター・原子力コード高速化作業の一環として、高度情報科学技術研究機構（RIST）の和田暁雄氏に実施していただきました。

- 飛跡を計算する最小&最大エネルギーを指定するパラメータ `esmin` と `esmax` の単位を MeV から MeV/n に変更しました。

- EGS5 を使って高エネルギー光子（約 10MeV 以上）を輸送する際、ジオメトリを細かく区切ると光子の飛程が長くなってしまうバグを修正しました。

---

[15] A. Endo, Y. Yamaguchi and K.F. Eckerman, Nuclear decay data for dosimetry calculation - Revised data of ICRP Publication 38, JAERI 1347 (2005).

[16] S. Abe and T. Sato, Implementation of muon interaction models in PHITS, J. Nucl. Sci. Technol. (2016) [http://www.tandfonline.com/doi/abs/10.1080/00223131.2016.1210043]

[17] K. Minomo, K. Washiyama, and K. Ogata, J. Nucl. Sci. Tech. DOI:10.1080/00223131.2016.1213672

- 標的中に ^{1}H が存在する際、負ミューオンが全て ^{1}H に捕獲されてしまうバグを修正しました。

バージョン 2.84 では、幾つかの微小なバグを修正しました。(2016 年 3 月 16 日)

バージョン 2.83 では主に次の改良を行いました。(2016 年 3 月 3 日)

- 平均寿命 886.7 秒で中性子が陽子、電子、反電子ニュートリノに崩壊する反応を考慮できるようにしました。ただし、デフォルトでは時間カットオフ(1 秒)で粒子輸送がストップしてしまうため、中性子の崩壊が問題になるような巨大な体系では、中性子のみならず、その崩壊により生成される陽子や電子などの tmax を大きく設定して下さい。
- EGS5 を用いて付与エネルギーを計算する場合に、ドップラー効果によるエネルギーの微妙なゆらぎを光子によるエネルギー付与とカウントしてしまうバグを修正しました。このゆらぎが問題となる低エネルギー光子(主に 100keV 以下)によるエネルギー付与計算や、[t-deposit] で part = photon としていた場合に結果が変化しますので、ご注意下さい。
- [t-point] タリーよりも後ろに mesh = reg を含むタリーが書かれていると、そのタリーの結果が壊れてしまうバグを修正しました。
- infl を含むインプットファイルにおいて sumtally 機能が使えないバグを修正しました。

バージョン 2.82 では主に次の改良を行いました。(2015 年 12 月 16 日)

- ある点やリング状の線分上における中性子・光子フルエンスを効率的に計算するポイントタリー [t-point] を導入しました(7.3 節及び \utility\tpoint 参照)。
- [t-yield] タリーに、弾性散乱により反跳された標的核情報を出力する elastic オプションを加えました。
- [t-star] タリーに、標的核が核変換を起こした場合の star density を出力するオプション output = transmut を加えました。
- [counter] セクションに、核分裂のオプション fiss を加えました。これにより、核分裂を経由して生成する粒子、特に核分裂の世代(回数)毎の情報を選択的にタリー可能となりました。
- [source] セクションに、文献[18] で計算される自発核分裂からの中性子線源を模擬するオプションを加えました。この改良は、株式会社ナイスの Liem Peng Hong 氏のご協力を得て行いました(5.3.2 節参照)。
- [source] セクションに、三角柱内に分布する線源を表現するオプションを加えました(5.3.13 節参照)。
- [source] セクションで、任意の時間分布を指定可能としました(5.3.21 節参照)。
- [parameters] セクションに、中性子増倍効果を制御する NONU パラメータを追加しました。この機能により、臨界解析コード等により別途評価した同体系における中性子増倍効率を加味した中性子線源を用いて、実質的な中性子透過問題の評価が行えるようになりました。
- [t-track] タリーで mesh = r-z の場合に、円面の角度 θ も定義可能としました。この機能により、扇形柱体における粒子フルエンスをタリー可能となりました。

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

- EGS5 に関連するいくつかのバグ（[t-track] を使ったときに電子のエネルギーが増えてしまうバグ、物質として水を指定したときにホルミウム（元素記号 Ho）の値を使って密度効果などを補正してしまうバグなど）を修正しました。
- 核共鳴蛍光散乱において入射光子の偏光効果を考慮可能としました。
- [t-dpa] を使った再開始計算を可能としました。
- Sumtally 機能を拡張し、[t-dchain] 以外の全てのタリーに対応しました。（バージョン 2.88 より、全てのタリーで本機能が使えるようになりました。）
- [t-deposit] タリーで output=deposit として part を指定した場合、各粒子の寄与を正しく計算可能としました。
- ミューオン核反応、光核反応、JQMD-2.0 に関する軽微なバグを修正しました。
- utility フォルダに、連続四面体 (TetraGEOM), ポイントタリー (tpoint)、ユーザー定義タリー (usrtally) の使用方法の解説を加えました。

バージョン 2.81 では主に次の改良を行いました。（2015 年 10 月 15 日）

- makefile で依存関係を正しく記述するようにしました。この改良により、make -j オプションが使えるようになり、コンパイルが速くなりました。なお、makefile を使ってコンパイルした際に作られる実行ファイル名が変更されていますのでご注意下さい。この改良は、原子力機構 原子力センシング研究グループの古高和禎氏の協力の下、実施いたしました。
- EGS5 を使用したときに使える物質制限をなくしました。ただし、数 100 以上の物質を定義した場合は、メモリ不足により計算が実行できない場合があります。また、EGS5 の仕様により、1 つの物質に含まれる元素数の最大値は 20 となります。
- EGS5 で [t-track] を使ったときに希に発生するバグを修正しました。
- EGS5 で [t-deposit] タリー、output = deposit のときに全吸収ピークエネルギーが多少低くなってしまうバグを修正しました。
- [t-dchain] で継続行を正しく認識するようにしましたまた、[t-dchain] で指定できる領域の最大数を 100 から 500 に拡張しました
- [t-deposit] にて mesh=reg,output=deposit で計算した際、デルタ線を生成した粒子のエネルギー付与量計算の誤りを修正しました。
- [t-heat] において mesh=r-z とした場合に起こるバグを修正しました。

バージョン 2.80 では主に次の改良を行いました。（2015 年 9 月 2 日）

- 連続四面体形状（ポリゴンの一種）の読み込み機能を組み込みました（5.6.5.4 節参照）本改良は、韓国 Hanyang 大学の HUREL 研究所の協力の下、実施しました。
- ミューオンによる制動放射及び対生成を再現できるようにしました。また、ミューオン捕獲反応におけるバグを修正しました。これにより、ミューオンが引き起こすほぼ全ての反応を精度よく再現可能となりました。
- 核共鳴散乱（NRF）計算機能を組み込みました。これにより、数 MeV 程度の光子照射による原子核の励起や核異性体の生成が再現できるようになりました。核共鳴蛍光散乱モデルは [parameters] セクションで ipnint=2 とすることにより起動します。

- Sumtally 機能を拡充し、[t-dpa]，[t-dchain] を除く全てのタリー結果を足し合わせることが可能となりました。(バージョン 2.88 より、全てのタリーで本機能が使えるようになりました。)
- ユーザー指定断面積の読み込み機能を追加しました（5.17 節参照）。
- 荷電粒子のエネルギー分散計算方法を修正しました。この改良により、陽子や重イオンによるブラッグピーク付近の吸収線量がより精度よく再現可能となりました。
- 飛程に関するパラメータ deltc，deltm を物質の密度で割ることにより、気体中の計算時間を短縮しました。
- Dump 線源を使った繋ぎ計算時の統計誤差計算方法を改良しました（5.3.17 節参照）。
- pnimul パラメータを導入し、光核反応を優先して引き起こす機能を追加しました。
- [t-yield] の誤差計算方法のバグを修正しました
- EGS5 に関する改良点
  - ipegs パラメータを導入し、PEGS5 のみ実行したり、PEGS5 は実行せずに PHI TS のみ実行したりすることを可能としました。
  - imsegs パラメータを導入し、EGS5 を使った場合、物質が変わるたびに多重散乱を詳細に模擬するようにしました。これにより、chard パラメータを変更することなく、薄膜による電子の散乱を模擬可能としました（PHITS オリジナルオプション）。
  - EGS5 を使った場合に、電子の飛程が短くなるバグ（v2.77 のみ）を修正しました。
  - EGS5 で使用するメモリの一部を動的配列化することにより、EGS5 を使った場合の物質制限数をなくしました。本改良は、原子力機構・システム計算科学センター・原子力コード高速化作業の一環として高度情報科学技術研究機構 (RIST) の足立将晶氏に実施していただきました。

バージョン 2.77 では主に次の改良を行いました。(2015 年 5 月 19 日)

- EGS5 を使用した際、axis=eng としたときに、タリー結果が不自然な飛び飛びの値を取るという不具合を修正しました。
- ミューオン捕獲反応を記述するモデルを改良しました。
- 重陽子や ^{4}He などの軽イオンが標的となる核反応を、逆運動学によって計算するようにしました。すなわち、これらの反応が起こった際、重イオンが入射粒子の場合でも、デフォルトの設定では INCL が核反応モデルとして使用されます。

バージョン 2.76 では主に次の改良を行いました。(2015 年 3 月 23 日)

- ミューオンが仮想光子を介して引き起こす核反応モデルを組み込みました。また、負ミューオンが物質内で止まったときに起こるミューオン捕獲反応を考慮できるようになり、この反応で生成されるミューオニックアトムからの特性 X 線の放出、及びその後に起こる核吸収反応も再現できるようになりました。
- nspred = 2 として荷電粒子の角度分散を考慮する際、その分散の大きさを調整するパラメータを導入しました。
- Intel Fortran コンパイラの最新版 (2015) で生じるバグを修正しました。

バージョン 2.75 では、複数のタリーセクションが書かれたインプットファイルで sumtally 機能が動作しないバグを修正しました。また、`e-mode=2` を選択した場合に発生するバグを修正しました。（2015 年 2 月 9 日）

バージョン 2.74 では主に次の改良を行いました。（2015 年 1 月 30 日）

- PHITS パッケージに含まれる DCHAIN のバージョンを DCHAIN2001 (dchain264.exe) から DCHAIN2014(dchain274.exe) に変更しました。DCHAIN2001 と比べた DCHAIN2014 の主な改良点は以下の通りです。
    (1) 入力ファイル形式の変更
    (2) 放射化計算断面積ライブラリの中性子エネルギー群数を 175 から 1968 に変更
    (3) PHITS 入力形式での [source] セクション出力オプションの追加
    (4) ANGEL 入力形式での残留放射能出力オプションの追加
- EGS5 部分もスレッド並列化しました。これにより、`negs=1` としたときにスレッド並列で実行できなくなる制限がなくなりました。また、これに関連して、EGS5 部分のバグをいくつか修正しました。本並列化の開発は、原子力機構・システム計算科学センター・原子力コード高速化作業の一環として高度情報科学技術研究機構 (RIST) の足立将晶氏に実施していただきました。
- 複数のタリー結果を足し合わせる新しい機能 “sumtally” を追加しました。（バージョン 2.88 より、全てのタリーで本機能が使えるようになりました。）詳しくは、6.8 節をご参照下さい。本機能の開発は、原子力機構・システム計算科学センター・スーパーコンピュータ利用に係るプログラミング支援作業の一環として RIST の三浦孝充氏に実施していただきました。
- Kurotama モデルで、5GeV/u 以上の断面積を出力できるようにしました。詳しくは文献[19] をご参照下さい。
- γ 脱励起に関するデータベース `trxcrd.dat` をソースファイルに組み込みました。この改良により、`e-mode`≥1 や`|igamma|`≥1 のときに `file(14)` を指定する必要がなくなりました。
- JAM 及び JAMQMD などに関連するバグをいくつか修正しました。

バージョン 2.73 では、核反応モデルによる計算結果として di-neutron 等の異常な原子核が生成されるバグを修正しました。また、Windows OS については、メモリ共有型並列計算用の実行ファイルとして 64bit 版をインストールするようにしました。シングルコアによる計算は 32bit 版と 64bit 版の両方の Windows で動作しますが、メモリ共有型並列計算は 32bit 版では動作しなくなりますのでご注意ください。（2014 年 11 月 5 日）

バージョン 2.72 では、`igamma=2` を選択した際に発生するバグと GEM の計算結果として di-neutron（2 つの中性子で構成される原子核）が生成されるバグを修正しました。また、[source] セクションにおいて `a-type` により角分布を定義する際、角度を degree 単位で与えていた場合のバグを修正しました。cos の値に変換して内挿を実行すべきところを degree 単位のまま行っており、指定したビン幅で一様な分布を取るべきところがバイアスのかかった分布となっておりましたのでご注意ください。加えて、[source] の `a-type` サブセクションにおいて、`na` や `nn` を負の値で与えられないようにしました。 （2014 年 10 月 21 日）

バージョン 2.71 では、EGS5 を使用した際の電子対消滅の取り扱いに関するバグを修正しました。（2014 年 9 月 26 日）

バージョン 2.70 では主に次の改良を行いました。（2014 年 8 月 30 日）

[19] L. Sihver *et al.*, Nucl. Instr. & Meth. B 334, 34-39 (2014).

- 電磁カスケード計算コード EGS5 [20] を組み込み、光子・電子・陽電子の輸送に使用することが可能となりました。EGS5 は [parameters] セクションで negs=1 とすることにより起動します。その際、file(20) で EGS5 用データライブラリ格納フォルダを指定する必要があります。ただし、現在のところ、EGS5 を使ってメモリ共有型並列計算を行うことはできませんのでご注意ください。また、EGS5 を使った場合、定義できる物質数が 100 に制限されます。（バージョン 2.80 よりこの制限はなくなりました。）詳しくは、5.2.13 節をご参照ください。 この開発は、KEK の平山英夫氏と波戸芳仁氏と共同で実施しました。

- 非共鳴領域の光核反応モデルを JAM に組み込むことにより、1TeV までの光核反応を再現可能としました。

- ミューオンの仮想光子放出断面積モデル（Minorikawa *et al.* Nuove Cimento 1981）と光核反応モデルを組み合わせることにより、1TeV までのミューオン核反応を再現可能としました。このモデルは、[parameters] セクションで imuint=1 とすることにより起動します。

- 核データライブラリに格納された荷電粒子放出断面積（(n, p), (n, α) など）を読み込むようにイベントジェネレータモード ver.2 を改良しました。この改良により、20MeV 以下の中性子核反応から放出される荷電粒子スペクトルが精度よく計算できるようになりました。このモードは、[parameters] セクションで e-mode=2 とすることにより起動します。

- 重イオン核反応モデル JQMD に相対論効果や初期状態安定アルゴリズムを組み込んだ JQMD-2.0 モデルを開発しました。JQMD-2.0 モデルは、[parameters] セクションで irqmd = 1 とすることにより起動します。この開発は、フランス CEA の Davide Mancusi 氏と共同で実施しました。

- [t-deposit] で output=deposit としてイベント毎の付与エネルギーを計算する際、実測を再現するように意図的にエネルギー分解能を持たせる機能を追加しました。

バージョン 2.67 では主に次の改良を行いました。（2014 年 5 月 22 日）

- 領域エラーチェック機能を追加しました。ジオメトリを 2 次元表示するタリーで自動的に実行され、2 重定義や未定義の領域がある場合にその領域に色を付けて出力します。詳しくは 10 節を参照してください。

- 20MeV 以下の中性子輸送にイベントジェネレータモードを適用した際、二次中性子が 2 個以上放出される反応で二次中性子のエネルギー・角度分布が核データからずれるのを阻止する拡張機能を導入しました。e-mode=2 で使用できます。この改良により 20MeV 以下の中性子に対するイベント解析の精度が向上しました。詳しくは 5.2.26 節をご参照ください。

- file(6)(D=phits.out) に出力される PHITS のサマリー情報を指定するパラメータ infout を用意しました。これにより必要とする情報のみを出力させることができるようになります。

- コンソール画面に現在計算中のバッチ番号を出力するようにしました。また、断面積ファイルが見つからないなど、PHITS の実行を停止してしまうような重要なエラーメッセージをコンソール画面にも出力するようにしました。

- s-type=18 とすることで、円錐形状の線源領域を指定できるようになりました。

- dumpall オプションと [t-cross], [t-time], [t-product] における dump 機能が再開始計算を実行した場合でも利用できるようになりました。各タリーで dump 機能を使用した際に出力していた.cfg の内容を file=で指定するファイルに出力し、dump data 自体は "_dmp" が付いたファイルに書き出すことにしました。

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

- `param.inc` の中で指定している *PHITS* が使えるメモリの最大値 (`mdas`) を 120,000,000（=1GB 相当）に上げ、lattice 格子数の最大値 latmax を 25,000,000 に上げました。この変更により ICRP ファントムなど詳細な人体ボクセルファントムが再コンパイルなしで扱えるようになりました。

バージョン 2.66 では主に次の改良を行いました。（2014 年 2 月 21 日）

- DWBA（歪曲波ボルン近似）計算で求めた離散スペクトルを考慮する機能を追加しました。特定の標的原子核における陽子、重陽子入射反応を対象として、放出中性子や陽子のエネルギースペクトルに、DWBA によって計算した離散スペクトルを加えます。
- 光子入射によるパイオン生成反応として Δ 共鳴と N^* 共鳴過程を組み込むことにより、1GeV までの光核反応を計算できるようにしました。（バージョン 2.70 より 1TeV まで計算できるようになりました。）
- Gy 単位によるタリー出力オプションを `[t-heat]` タリーにも拡張しました。また、質量密度が 0 の場合に結果が `NaN` となってしまうバグを修正しました。
- [source] セクションにおいて、`e-type = 2,3,5,6,7,12,15,16` を設定して関数系でエネルギー分布を与える場合に、その分割数 nm が負の場合に発生するバグを修正しました。同様に、`a-type = 5,6,15,16` で角度分布を関数系で与える場合に、分割数 nn を負とした際のバグを修正しました。

バージョン 2.65 では、`[t-deposit]` タリーを改良し、Gy の単位で dose 結果を出力できるようにしました。また、[material] と [cell] セクションにおいて、密度を質量密度で定義していた場合に発生するバグを修正しました。原子数密度への変換を行う際に間違いがあり、中性子が過剰な元素を扱う場合にその影響が出ていました。最大で 0.6%程度の影響がありました。 （2014 年 1 月 30 日）

バージョン 2.64 では、光核反応や EBITEM、その他の幾つかについてバグ修正を行いました。また、`[t-heat]` を使って以下の核種に対する中性子からの付与エネルギーを計算した場合に、結果が `NaN` となってしまうバグがありましたので、それらを修正しています。カーマ近似を使わない場合の結果に影響はありません。（2013 年 11 月 19 日）

```
As075 Ba130 Ba132 Ba134 Ba135 Ba136 Ba137 Ba140 Br079 Br081 Cd106 Cd108
Cd110 Cd111 Cd112 Cd113 Cd114 Cd116 Ce141 Ce142 Ce143 Ce144 Cf250 Fe059
Ga069 Ga071 Hf174 Hf176 Hf177 Hf178 Hf179 Hf180 Hf181 Hf182 I_127 I_129
I_130 I_131 I_135 In113 In115 Kr078 Kr080 Kr082 Kr083 Kr084 Kr085 La138
La139 La140 Mo092 Mo094 Mo095 Mo096 Mo097 Mo098 Mo099 Mo100 Nb094 Nb095
Ni059 Pr141 Pr143 Rb085 Rb086 Rb087 Rh103 Rh105 Ru096 Ru098 Ru099 Ru100
Ru101 Ru102 Ru103 Ru104 Ru105 Ru106 Sb121 Sb123 Sb124 Sb125 Sb126 Se074
Se076 Se077 Se078 Se079 Se080 Se082 Sr084 Sr086 Sr087 Sr088 Sr089 Sr090
Tc099 Te120 Te122 Te123 Te124 Te125 Te126 Te127m Te128 Te129m Te130 Te132
Xe124 Xe126 Xe128 Xe129 Xe130 Xe131 Xe132 Xe133 Xe134 Xe135 Y_089 Y_090
Y_091 Yb168 Yb170 Yb171 Yb172 Yb173 Yb174 Yb176 Zr093 Zr095
```

バージョン 2.60 では主に次の改良を行いました。（2013 年 8 月 22 日）

- 核反応後に生成する残留核の脱励起を ENSDF (Evaluated Nuclear Structure Data File) データベースに基づいて計算するモデル (EBITEM: ENSDF-Based Isomeric Transition and isomEr production Model) を導入しました。この改良により、不連続なピークを持つ即発 γ 線のエネルギースペクトルを精度よく再現できるようになりました。また、このモデルを用いることで準安定核（アイソマー）の生成率を計算できるようになりました（詳しくは 5.2.7 節を参照してください)。

- 光子入射による準重陽子崩壊過程を組み込むことにより、140MeV までの光核反応を計算できるようにしました。(バージョン 2.70 より 1TeV まで計算できるようになりました。) また、^{6}Li, ^{12}C, ^{14}N, ^{16}O に対して、巨大共鳴が起きたときの蒸発過程にアイソスピン依存性を考慮するようにしました。この改良により、これらの核種から α 粒子の放出が抑えられ、より多くの中性子や陽子が生成されるようになりました。
- 電磁混合場における放射線挙動（電子除く）を模擬できるようにしました。詳しくは 5.11 節を参照してください。
- `[source]` セクションにおいてエネルギー分布を定義する際、エネルギー微分値、すなわち単位が [/MeV] で与えられた線源スペクトルをそのまま利用できるようになりました。また、エネルギー分布としてポイントワイズ、すなわち各点各点で与えられる不連続なものも定義できるようになりました。詳細は 5.3.19 節をご覧ください。
- いくつかのアルゴリズムを最適化して計算時間を短縮しました。特にメッシュ数の多いタリー計算で計算時間が短くなりました。また、タリー及び *ANGEL* におけるメモリ使用方法を改善しました。本改良は、原子力機構・システム計算科学センター・原子力コード高速化作業の一環として (株) 富士通システムズ・イーストの大日向大地氏に実施していただきました。
- 下記に示すバグ修正及び軽微な改良を行いました。
  - PHITS と INCL4.6 の参考文献を修正
  - `[t-dchain]` でタリー領域数の制限をなくし、Lattice に対応
  - JENDL-4.0 の更新に併せていくつかの核データを更新
  - 電子の最大エネルギー制限を 1GeV から 10GeV に変更
  - JAM、SMM でまれに発生するイベントによるバグを修正
  - δ 線生成に関するバグを修正
  - 電子の Lost Particle が起きたときのバグを修正
  - 異常終了したときの結果からも再開始計算を可能とした
  - 物質の密度を乗じる新たな multiplier function を追加
  - 7 桁のセル番号を使えるようにした
  - `istdev = 2` で `xyz-mesh` タリーを使用した場合に計算時間が膨大になることを回避
  - 陽子の断面積ライブラリ読み込み時のバグを修正
- また、スレッド並列化未対応のため、CG ジオメトリに対するサポートを終了しました。今後はマクロボディを活用して GG でジオメトリを作成してください。なお、CG ジオメトリで書いた PHITS のインプットファイルは、スレッド並列を使わない限り従来通り動作します。

バージョン 2.52 では主に次の改良を行いました。(2012 年 12 月 27 日)

- 電子、陽電子、および光子の輸送について、新規のアルゴリズムを導入しました。電子と陽電子の阻止能をそれらの計算打切エネルギーに応じて変化させ、高エネルギー電子の挙動が計算打切エネルギーに依存しないようにしました。加えて、光子や電子の輸送計算でもイベントジェネレータとなるように改良しました。
- また、DCHAIN 用のインプットファイルを作成することができる `[t-dchain]` タリーを新しく実装しました。DCHAIN は放射線による物質の放射化の時間変化を調べることができるコードで、このタリーを用いることで PHITS と DCHAIN の接続が容易になります。詳細は 7.15 節をご覧ください。

- 新たなマクロボディーとして、楕円柱 REC (Right Ellliptical Cylinder)、カットされた円錐形 TRC (Truncated Right-angle Cone)、楕円体 ELL (Ellipsoid)、くさび形 WED (Wedge) を追加しました。

バージョン 2.50 では次の様々な機能を追加しました。（2012 年 9 月 25 日）

- タリー出力の統計誤差が正しく計算できるようになりました。また、統計が十分でなかった場合などに対応するため、古い計算結果から次の初期乱数とタリー出力を読み込んで再開始計算を行う機能を実装しました。詳細は 5.2.2 節をご覧ください。本改良は、原子力機構・システム計算科学センター・原子力コード高速化作業の一環として (株) 富士通システムズ・イーストの大日向大地氏に実施していただきました。

- メモリ共有型並列計算に対応させるため、ソースを大幅に書き換えました。ただし、メモリ共有型並列には、まだいくつかの制限がありますのでご注意下さい (12.2 節参照)。また、それに伴い、古いコンパイラー (f77, g77 など) ではコンパイルできなくなりました。詳しくは 11 節を参照してください。本研究の成果は次世代生命体統合シミュレーションソフトウェアの研究開発プロジェクト、理化学研究所戦略的研究展開事業、理化学研究所基礎科学特別研究員制度の支援によって得られたものです。また、開発において「京」コンピュータ試験利用および理研情報基盤センター RICC システムを利用させていただきました。

- 巨大共鳴反応断面積を評価済み核データである光子入射反応データ (JENDL-PD/2004) を使用するように変更しました。ただし、光核反応の中で現在の *PHITS* が取り扱うことのできる反応は巨大共鳴反応だけです。入射光子エネルギーが巨大共鳴領域より大きい場合（約 20MeV 以上）、*PHITS* の計算は過小評価します。ご注意ください。

- 蒸発モデルの GEM を拡張し、統計マルチフラグメンテーションモデル (SMM) を加えました。これにより、元の核から大幅に (60-90 %程度) 質量が減るような核の生成を正確に評価できるようになりました。

- 核子、パイオン、軽イオン入射反応を精密に記述する INCL(Intra-Nuclear Cascade of Liège) 模型を核反応模型として組み込みました。本バージョン以降、これらの粒子が関与する反応では初期設定で INCL が選択されます。本模型の PHITS への組込は、(独) 日本原子力研究開発機構と CEA/Saclay の共同研究の一環として、Liège 大学の Joseph Cugnon 氏、CEA/Saclay の Davide Mancusi 氏、Alain Boudard 氏、Jean-Christophe David 氏、Sylvie Leray 氏らの協力のもと行いました。

- 最新の反応断面積模型である KUROTAMA 模型を組み込みました。天然に存在する安定核はもちろん不安定核種が関与する反応についても、非常に幅広いエネルギー領域に対して適用可能となっています。この組み込みは、(独) 理化学研究所の小濱洋央氏、(大) 高知大学の飯田圭氏、(学) 愛知淑徳学園の親松和浩氏らとの共同開発によるものです。

- 核子入射反応において軽イオン生成過程を取り入れた INC-ELF(Intra-Nuclear Cascade with Emission of Light Fragment) 模型を核反応模型として組み込みました。この組み込みは、(独) 日本原子力研究開発機構と (大) 九州大学の間の共同研究の一環として魚住研究グループに実施していただきました。

- ユーザーが任意の物理量を導出できるよう設定できるユーザー定義タリー `[t-userdefined]` を加えました。これにより、既存のタリーでは難しかったシミュレーション結果に対する詳細な分析が可能となります。ただし、ソースファイルのコンパイルが必要となります。詳しい使用方法は 7.19 節をご覧ください。

- ^{35}Cl など幾つかの核種に関して中性子の Kerma factor を更新しました。また、光子-原子および電子-原子データライブラリーをそれぞれ JENDL-4.0 とリバモア評価済電子データライブラリー (EEDL) を基にして新たに開発し、利用できるようにしました。

バージョン 2.30 では、材料損傷の指標である “原子あたりのはじき出し数 (Displacement Per Atom, DPA)” 導出の計算モデルにおいて、輸送荷電粒子のクーロン弾性散乱の寄与を含むように拡張しました。これにより従来よりも DPA の再現性が向上しました。また、[multiplier] セクションを追加し、任意のエネルギー依存の係数を [t-track] タリーの結果に掛けることが可能となりました。（2011 年 8 月 18 日）

バージョン 2.28 では、`dumpall` オプションと `[t-cross]`, `[t-time]`, `[t-product]` における dump 機能が MPI による並列計算でも利用できるようになりました。使用する並列 PE (Processor Element) 数 −1 個のファイルを作成し、PE 毎にファイルを変えて各結果を書き出します。読み込みも同様で、各 PE に対応したファイルの中身をそれぞれが読み込みます。

バージョン 2.26 では、荷電粒子が物質中を通過する際に発生する δ 線を 2 次粒子として実際に輸送させることができるようになりました。[delta ray] セクションを利用して領域毎にしきい値エネルギーを指定することにより、そのエネルギー以上の δ 線を発生させます。

1.2 開発体制

PHITS の開発は、日本原子力研究開発機構が中心となって、高度情報科学技術研究機構、高エネルギー加速器研究機構、九州大学等と共同で進められています。

現在の開発メンバーは以下の通りです。
(財) 高度情報科学技術研究機構 (RIST)
仁井田浩二

(国研) 日本原子力研究開発機構 (JAEA)
佐藤達彦、岩元洋介、橋本慎太郎、小川達彦、古田琢哉、安部晋一郎、甲斐健師、松田規宏、松谷悠佑、Hunter Ratliff, Lan Yao, Pi-En Tsai[21]

(共) 高エネルギー加速器研究機構 (KEK)
岩瀬広

(国) 九州大学 (Kyushu University)
執行信寛

ウィーン工科大学, オーストリア
Lembit Sihver

また、これまでに以下の方々が PHITS の開発に寄与されました。

原子力機構
高田弘、明午伸一郎、勅使河原誠、前川藤夫、原田正英、池田裕二郎、坂本幸夫、中島宏、深堀智生、奥村啓介、甲斐哲也、野田秀作

[21] 現所属: SLAC, USA

(国) 東京工業大学 (TITech)
　千葉敏

東北大学工学部
　中村尚司

Chalmers University, Sweden
　Davide Mancusi

1.3 PHITS の参考文献

バージョンに関わらず、PHITS をご使用になられた場合は次の文献を引用してください。

- 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).

この文献はオープンアクセスとなっており、次の URL からダウンロードできます。

`https://www.tandfonline.com/doi/full/10.1080/00223131.2017.1419890`

この他に、PHITS がもつ特徴についてまとめた文献には次のものがあります。

- 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).

2　インストール及び実行方法

PHITS は、Windows（7 以降）、Mac（OS X v10.6 以降）及び Linux 上で動作する Fortran プログラムです。各環境にあわせて用意されたインストーラを使用することで、簡単に実行環境を準備できます。

PHITS をインストールするにあたっては、物理メモリが 1GB（推奨は 2GB）、ハードディスクの空き容量が 6GB のコンピュータをご用意ください。また、Windows と Mac の OS に関しては、管理者 (Administrator) 権限をもつアカウントで使用する必要があります。

以下の 2.1, 2.2, 2.3 節では、それぞれ Windows, Mac, Linux OS における PHITS のインストールと実行方法についてご紹介します。

2.1　Windows OS におけるインストール及び実行方法

2.1.1　インストール方法

(1) 古いバージョンの PHITS をインストールしている場合は、そのフォルダ名を変更（同じフォルダにインストールしない）

(2) PHITS パッケージにある “setup-jpn.vbs” をダブルクリック

(3) インストールフォルダを指定 (“c:/” を奨励します)

(4) インストール用のパスワードを入力

(5) インストール完了

(6) インストールが適切に完了したかどうかを確認するため、“/phits/lecture/basic/lec01/lec01.inp” のファイルを指定し、右クリックして “送る” →“PHITS” を選択

(7) “xz_track_all.eps” が作成されたことを確認

（注意）インストールフォルダや PHITS のインプットファイルがあるフォルダの名称にスペースや漢字があるとエラーになる場合がありますので、そのような文字が使われていないフォルダでご利用ください。

（注意）パスワードを間違えた場合は、作成された “phits” フォルダを一度削除して再インストールしてください。

インストーラは、下記の内容を実施します。

(1) “phits.zip” を指定フォルダに解凍する。

(2) PHITS 実行形式を含むフォルダ “/phits/bin/” に “PATH” を通す。

(3) “/phits/bin/” フォルダにある “/phits.bat/” と “angel.bat” 及び “/phits/dchain-sp/bin/” フォルダにある “dchain.bat” 内の実行ファイル名をインストールフォルダに合わせて変更し、そのショートカットを “sendto” フォルダに作成する。[22]

(4) “/phits/data/” フォルダにある核データリストファイル “xsdir.jnd” の 1 行目を `datapath=`インストールフォルダ `+/ phits/ XS` に書き換える。

[22] Mac や Linux のインストーラと異なり，インストールフォルダに合わせたインプットファイルの変更（例えば `file(1)` など）は行いません。デフォルトのフォルダ (“c:/”）以外を選択した場合，ほとんど全てのサンプルインプットが動かなくなりますのでご注意ください。

2.1.2　実行方法

PHITS を実行する場合は、“送る” による方法とコマンドプロンプトを用いる方法があります。

2.1.2.1　“送る” による実行方法

Windows の “送る” 機能を用いて PHITS を実行できます。PHITS の入力ファイルを指定して右クリックをすると、“送る” → “PHITS” が表示されますので、これを選択することで PHITS が実行されます。この方法では、“/phits/bin/” フォルダにある PHITS 用のバッチファイル “phits.bat” のショートカットを “送る” に登録しています。このため、“phits.bat” を編集することで、その変更を “送る” による PHITS の実行内容に反映させることができます。

メモリ共有型並列で実行する場合は、PHITS のインプットファイルにおいて、最初のセクションの前に `$OMP=`N（N は使用する CPU コア数）を加えてください。その際、$N = 1$ の場合は並列計算を使用しません。また、$N = 0$ の場合は計算機が持つ全てのコアを使用します。なお、バージョン 2.73 より、メモリ共有型並列計算は 64bit 版 Windows のみで動作するようにしていますのでご注意ください。

また，`$MPI=`M と指定すれば並列数 M のメモリ分散型並列で実行可能ですが，そのためには，事前に MPICH2 のインストールが必要になります。詳しくは 12.1.1 節をご参照下さい。

2.1.2.2　コマンドプロンプトからの実行方法

Windows の CUI (Character user interface) である “コマンドプロンプト” から、コマンド入力により PHITS を実行できます。Windows の “スタートボタン” をクリックし、検索ボックスに “cmd” と入力して表示される “cmd.exe” を選択すると、図 1 に示すようなコマンドプロンプトのウインドウが立ち上がります。

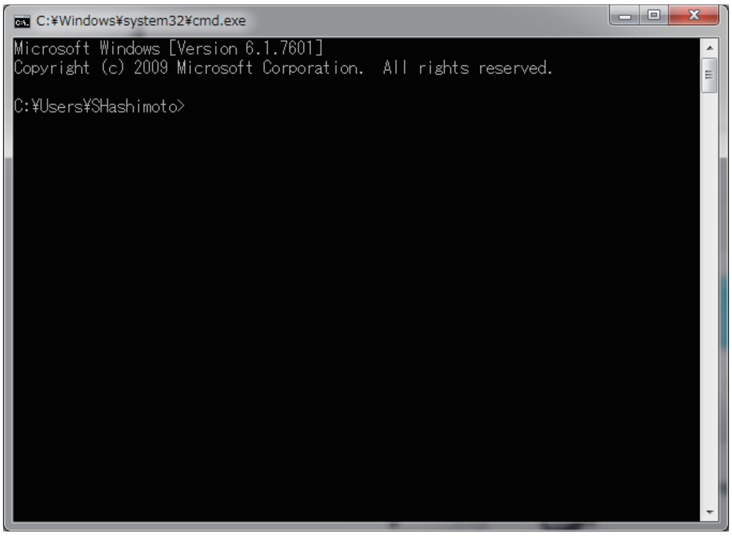


図 1: コマンドプロンプトのウインドウ

PHITS の実行方法は、“cd” コマンドを使ってインプットファイルのあるフォルダに移動した後、コマンドプロンプト上で下記のように入力します。

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

ここで、“phits.inp” は PHITS のインプットファイル名です（例：lec01.inp）。コマンドプロンプトにおいて ↑ キーを押すと、過去のコマンド履歴が出ますので、同じインプットファイルを何度も実行する場合に便利です。

2.1.3 推奨ソフトウェア

PHITS を実行するために必要なソフトウェアは特にありません。ただし、PHITS の入力ファイルを作るためには、行番号を表示可能なテキストエディタがインストールされていることが望ましいです（エラーが生じたときに、原因となる入力ファイルの行番号が表示されるため）。また、画像出力ファイル（EPS 形式）を見るためには、Ghostscript 及び GSview をインストールする必要があります。Windows 用のフリーのテキストエディタは、

- TeraPad 1.08（日本語）(http://www5f.biglobe.ne.jp/~t-susumu/)
- Crimson Editor（英語）(http://www.crimsoneditor.com/)

などがあります。Ghostscript 及び GSview（バージョン 5 を推奨）のインストールに関しては、下記のホームページをご参照ください。

- Ghostscript (http://www.ghostscript.com/)
- GSview (http://pages.cs.wisc.edu/~ghost/gsview/index.htm)

2.2 Mac OS におけるインストール及び実行方法

2.2.1 インストール方法

(1) PHITS パッケージに入っている “/mac/” フォルダの中の “phits_installer” をダブルクリックします。

(2) インストールモードの選択において、“Automatic” を選択します（図 2 参照）。（“Manual” は “Automatic” が上手く動作しない場合に選択してください。（詳細は PHITS パッケージ中の “/mac/README-jpn.pdf” をご参照ください。）

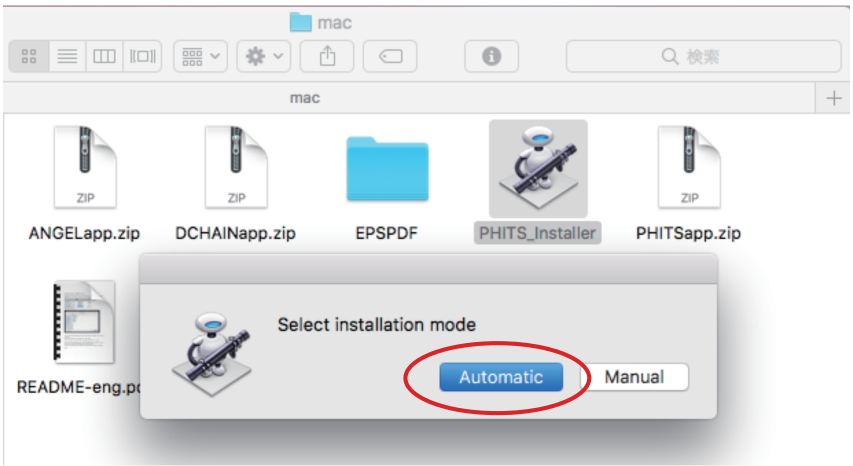


図 2: インストールモードの選択

(3) インストール先のフォルダを指定します。通常、アカウント名（例：iwamoto）と同名のフォルダが選択されるので、“選択” を押します（図 3 参照）。

(4) インストール先に “/phits/” という名前のフォルダが作られます。このフォルダの中には、PHITS 本体とソース、講習会の資料、例題などが入っています。

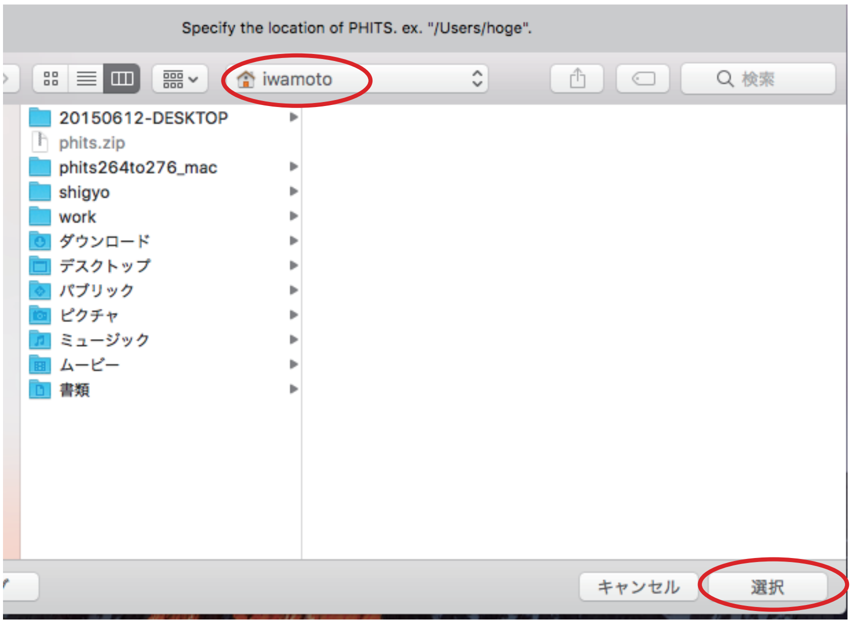


図 3: インストールフォルダの選択

(注 1) インストール時に入力するパスワードを間違えると、デスクトップにすぐにインストール完了の表示が出て、Dock に PHITS のアイコンが現れますが、全てのファイルがシステムにコピーされず、PHITS を実行できない状態になります。その場合は、手順 (1) に戻り、正しいパスワードを入力してインストールを進めてください。

(注 2) インストール後に “/phits/” フォルダを別のフォルダへ移動させると PHITS は動作しなくなります。その場合はもう一度インストールし直してください。

(注 3) インストール先に “/phits/” という名前のフォルダが存在する場合は、古いフォルダは “/phits[今日の日付].[現在の時刻]/” に名称が変更されます。

(注 4) PHITS パッケージをコピーするフォルダやインストールフォルダの名称にスペースや漢字があるとエラーになる場合がありますので、このような文字が使われていないフォルダにインストールしてください。

2.2.2 実行方法

PHITS を実行する場合は、ドラッグ&ドロップによる方法とターミナルを用いる方法があります。

2.2.2.1 ドラッグ&ドロップによる実行方法

ドラッグ&ドロップで実行する場合は、インプットファイルを Dock にある青い PHITS アイコンへドラッグ&ドロップします（図 4 参照）。新たにターミナルウィンドウが立ち上がり、計算状況が出力されます。計算結果の出力ファイルは、インプットファイルがあるフォルダに作られます。また、再度同じ名前のインプットで計算を実行したい場合は、計算状況が出力されたターミナルで ↑ キーを押してからリターンキーを押せば実行できます。

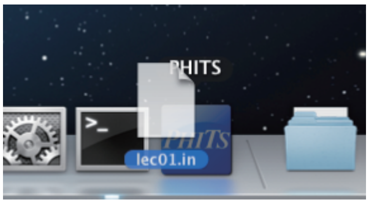


図 4: Dock にある PHITS アイコンへのドラッグ&ドロップ

メモリ共有型並列で実行する場合は、PHITS のインプットファイルにおいて、最初のセクションの前に `$OMP=`N（N は使用する CPU コア数）を加えてください。その際、$N = 1$ の場合は並列計算を使用しません。また、$N = 0$ の場合は計算機が持つ全てのコアを使用します。

また、ANGEL 及び DCHAIN を動かす際は、PHITS のタリーで出力されたそれぞれのインプットファイルを赤い ANGEL アイコンもしくは緑の DCHAIN アイコンにドラッグ&ドロップします。また、PHITS アイコンには、PHITS、ANGEL、DCHAIN のインプットファイルを自動識別する機能がついておりますので、ANGEL や DCHAIN のインプットファイルを PHITS アイコンにドラッグ&ドロップして ANGEL や DCHAIN を起動することも可能です。

2.2.2.2　ターミナルからの実行方法

“PHITS_Installer” によりインストールが完了した後、ターミナルからコマンド入力により PHITS を実行できます。Dock にある Finder アイコンをクリックして “アプリケーション” → “ユーティリティ” →“ターミナル” を選択して、ターミナルを起動してください。

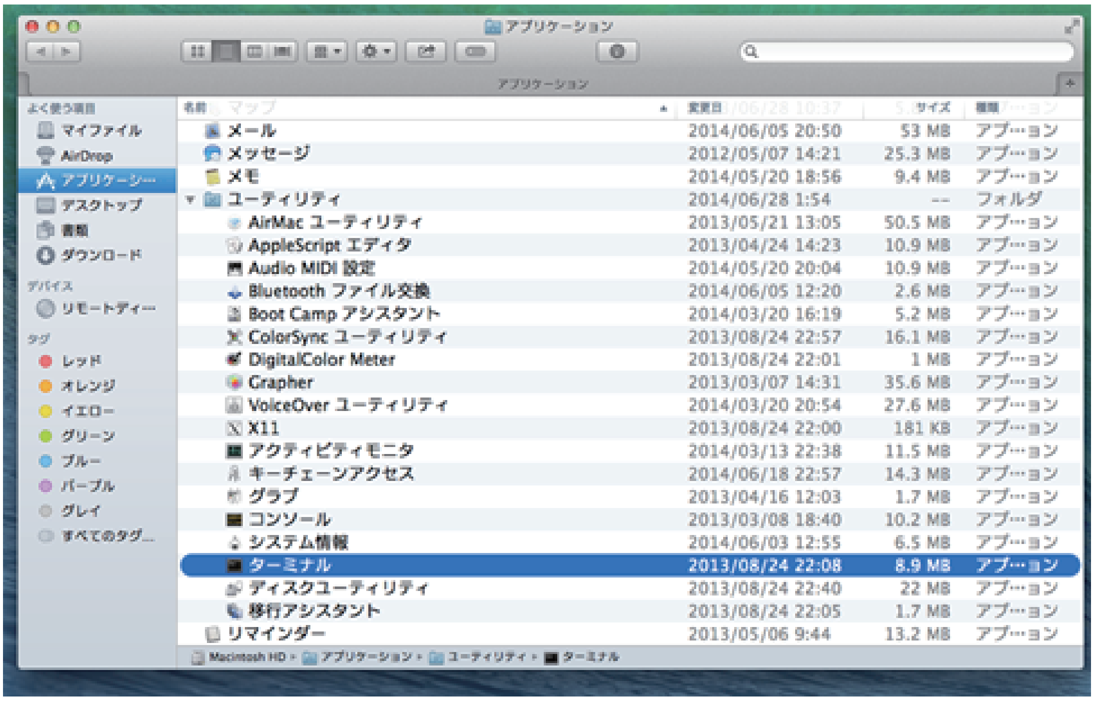


図 5: ターミナルを選択する画面

PHITS の実行方法は、“cd” コマンドを使ってインプットファイルのあるフォルダに移動した後、ターミナルで下記のように入力します。

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

ここで、“phits.inp” は PHITS のインプットファイル名です（例：lec01.inp）。ターミナルで↑キーを押せば過去のコマンド履歴が出ますので、同じインプットファイルを何度も実行する場合に便利です。

PHITS と同様、ANGEL や DCHAIN もターミナルから実行できます。ANGEL の実行方法は、ターミナルで

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

と入力します。ここで “angel.inp” は、ANGEL のインプットファイル名です（例：track_xz.out）。DCHAIN の実行方法は、ターミナルで

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

と入力します。ここで “dchain.out” は、DCHAIN の入力ファイル名（PHITS の [t-dchain] で指定したファイル名）です。

2.2.2.3 “PHITS_Installer” の “Automatic” と “Manual” が共に動作しないときの対応

ターミナルから PHITS を実行できるようにします。ターミナルを使って PHITS を初めて実行する場合は、PHITS の実行ファイルがあるフォルダに PATH を通す必要があります。PATH を通すためには、ターミナルで下記のコマンドを入力する必要があります。

```
echo export PATH=/PATH-TO-PHITS/phits/bin:${PATH} >> ~/.bash_profile
source ~/.bash_profile
```

ここで、“PATH-TO-PHITS” は各自のインストール先のフォルダ名に変更してください（例:/Users/iwamoto）。フォルダ名が分からない場合は、ターミナルを立ち上げた時点で、

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

を入力してください。ここで、“XXX” は PHITS のバージョンを表します。表示される結果から “/phits/bin/” を削除したものが “PATH-TO-PHITS” に対応します。なお、この PATH の設定は、初めて PHITS を実行する時のみ必要となり、それ以降は不要です。

そして、“/phits/bin/phits.sh” の 8 行目の PHITSver=252 をインストールしようとしているバージョンに合わせます。

以上により、PHITS をターミナルで実行できるようになります。実行方法については、前節をご覧ください。実行までの流れを図 6 に示します。

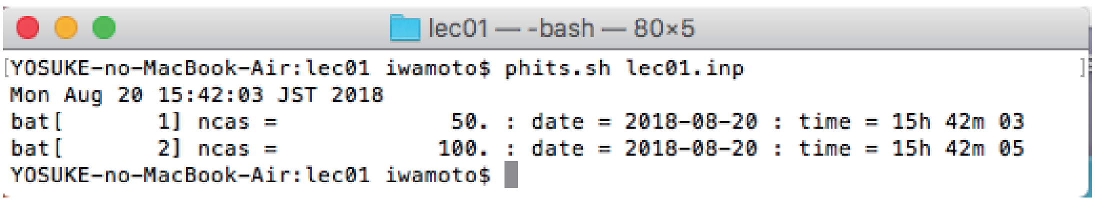


図 6: ターミナルを使った PHITS 実行の流れ

なお、ターミナルから DHCAIN を使う場合は、PHITS と同様に DCHAIN の実行ファイルがあるフォルダに PATH を通す必要があります。PATH の通し方は、PHITS の場合と同様に

```
echo export PATH=/PATH-TO-PHITS/phits/dchain-sp/bin:${PATH} >> ~/.bash_profile
source ~/.bash_profile
```

となります。ここで、“PATH-TO-PHITS” は各自のインストール先のフォルダ名に変更してください（例：/Users/iwamoto）。DCHAIN の実行方法についても前節をご参照ください。

2.3 Linux OS におけるインストール及び実行方法

2.3.1 インストール方法

まず、PHITS パッケージから “phits.zip” をインストールしたいフォルダにコピーします。また、PHITS パッケージの linux フォルダから Linux 用のインストーラ “PHITS-Installer_lin.sh” を同じフォルダにコピーしてください。インストーラに実行権限を付加するため、ターミナルで、このフォルダに移動して以下のコマンドを実行してください。

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

次にインストーラーを実行します。以下の様に実行してください。

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

ZIP ファイルの解凍、各ファイルに書かれた PATH の修正、実行ファイル PATH の設定の順でインストール作業が行われます。ZIP ファイルの解凍の際にパスワードの入力が求められますので、インストール用のパスワードを入力してください。Bash の使用を想定して PATH の設定を “.profile” に追加しているので、他のシェルを使用している方は適宜変更してください。インストールの途中でエラーが表示されないようであれば、無事インストールは完了です。

PATH の設定を有効にするために、以下のコマンドを実行してください。

```
source ~/.profile
```

次回ログイン以降は、ログイン時に自動的に “.profile” が読み込まれるはずですので、上のコマンドをログイン毎に行う必要はありません。

2.3.2 実行方法

Linux 環境ではターミナルで PHITS を実行します。ターミナルを起動し、“cd” コマンドを使ってインプットファイルのあるフォルダに移動した後、ターミナルで下記のように入力します。

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

ここで、“phits.inp” は PHITS のインプットファイル名です（例：lec01.inp）。ターミナルで ↑ キーを押せば過去のコマンド履歴が出ますので、同じインプットファイルを何度も実行する場合に便利です。

メモリ共有型並列で実行する場合は、PHITS のインプットファイルにおいて、最初のセクションの前に `$OMP=`N（N は使用する CPU コア数）を加えてください。その際、$N = 1$ の場合は並列計算を使用しません。また、$N = 0$ の場合は計算機が持つ全てのコアを使用します。

また，`$MPI=`M と指定すれば並列数 M のメモリ分散型並列で実行可能ですが，そのためには，事前に OpenMPI のインストールが必要になります。OpenMPI のインストールは、OpenMPI のホームページ (`https://www.open-mpi.org/`) をご参考の上、環境にあわせて行ってください。

PHITS と同様、ANGEL や DCHAIN もターミナルから実行できます。ANGEL の実行方法は、ターミナルで

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

と入力します。ここで “angel.inp” は、ANGEL のインプットファイル名です（例：track_xz.out）。DCHAIN の実行方法は、ターミナルで

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

と入力します。ここで “dchain.out” は、DCHAIN の入力ファイル名（PHITS の [t-dchain] で指定したファイル名）です。

(注) 実行時に許可がありません等のエラーがでる場合は、実行ファイルに実行権限が付加されていないことが原因と思われるので、実行ファイルが格納されているフォルダ “phits/bin/” 等に移動し、以下のコマンドにより実行権限の付加を行ってください。

```
chmod +x XXX
```

ここで、“XXX” は実行ファイル名です。

2.3.2.1 シェルを使用せず直接実行ファイルを指定して実行する方法

インプットファイルのあるフォルダに移動した後、以下の様にコマンドを入力することで、シェルファイル “phits.sh” を利用せずに直接実行ファイルを指定して実行することができます。

```
phitsXXX.exe < input.inp
```

ここで、“phitsXXX.exe” は PHITS 実行ファイル、“input.inp” はインプットファイルを表します。ただ、この実行方法で infl を用いる場合には注意が必要で、“input.inp” の 1 行目を

```
file = input.inp
```

として自身のファイル名を記述してください。infl の使い方については、4.3 節をご覧ください。

また、メモリ分散型並列では標準入力からの読み込みをしない仕様になっており、代わりに実行ディレクトリにある “phits.in” のファイルから入力ファイル名を読み込みます。この “phits.in” は固定です。このファイルの 1 行目に

```
file = input.inp
```

のように入力ファイル名を記述します。これは、メモリ分散型並列だけの制約です。

2.3.3 メモリ共有型 (OpenMP) 並列計算に失敗した場合

メモリ共有型並列の実行のために、“libiomp5.so” のライブラリが必要とされる場合があります。この場合には以下の手順に従い、設定を行ってください。まず、“libomp-dev” のパッケージを各ディストリビューションの手順に従いインストールして下さい。例えば Ubuntu の場合は、

```
sudo apt-get install libomp-dev
```

でインストールできます。これにより、“/usr/lib/x86_64-linux-gnu/libomp.so.5” のライブラリがインストールされます。次に、このディレクトリに移動

```
cd /usr/lib/x86_64-linux-gnu/
```

し、“libiomp5.so” にシンボリックリンクを作ります。

```
sudo ln -s libomp.so.5 libiomp5.so
```

これにより、“libiomp5.so” のライブラリが認識されるようになり、PHITS によるメモリ共有型並列計算が実行できるようになります。

3　再開始計算と中断方法

3.1　再開始計算

PHITS を実行して得られたタリー結果の統計量が足りない場合に、その結果を読み込んで、続き計算（再開始計算）を行うことが可能です。`[parameters]` セクションにおいて、`istdev < 0` とすることで、再開始計算を実行できます。その際の統計誤差計算モードは、過去のタリー結果を踏襲します。再開始計算は、各タリー別に行いますので、過去のインプットファイルに新たにタリーを加えた再開始計算も可能です。再開始計算の流れは、以下のようになります。

(1) 各タリーの `resfile` (デフォルトは `file` で指定した出力ファイル名と同じ) で指定したファイルの有無をチェックする。

(2) ファイルが存在しない場合は、そのタリーは新規タリーとして扱う。全てのファイルが存在しない場合は、`istdev = -2` の場合はヒストリー分散、`istdev= -1` の場合はバッチ分散モードとして新規計算を開始する。

(3) ファイルが存在する場合は、そのファイルから統計誤差計算モード (`istdev`)、これまでの全ウェイト数 (`resc2`) や全ヒストリー数 (`resc3`)、バッチ当たりのヒストリー数 (`maxcas`、バッチ分散モードのみ必要)、次の初期乱数シード (`rijklst`)、タリーパラメータ、結果及び相対誤差などを読み込む。

(4) 過去のタリーと入力ファイルで指定したタリーの一致性の確認する (一致しない場合はエラーメッセージを出力して実行停止。ただし、全てのパラメータをチェックするわけではないので注意)。

(5) 過去のタリーが複数ある場合、それらの間での統計誤差モードや `maxcas` (バッチ分散モードのみ) の一致性を確認する。一致した場合は、その値を再開始計算で利用する。一致しない場合は、エラーメッセージを出力して実行停止する。

(6) 初期乱数シードを一番目に読み込んだ過去タリーの `rijklst` に変更。(`rijklst` が過去タリー間で異なる場合は、警告を出力するが計算は可能)

(7) 再開始計算を実行し、その結果を過去のタリー結果に加えて `file` で指定したファイルに出力する。`resfile` を指定しない場合、出力ファイル (`file`) と過去タリーファイル (`resfile`) が同じであるため、過去の結果に新しい結果を上書きして出力する。

再開始計算を行う際の注意点としては、[1] 統計誤差計算モードの違うタリー同士で再開始計算をしないこと、[2] バッチ分散の場合は必ず `maxcas` を統一すること、[3] ジオメトリや入射粒子に関しては新旧ファイルの一致性を確認しないので、ユーザーの責任で必ず一致させること、などがあります。

3.2　計算の中断

プログラムを実行すると、カレントディレクトリに “batch.out” [23] というファイルが作られます。そのファイルにはひとつのバッチが終了する毎に、（メモリ分散型並列の場合にはバッチ数 × (PE −1) 毎に、）そのバッチの計算時間など、簡単な情報が出力されます。メモリ分散型並列の場合には、各 PE の状態が含まれます。異常終了が起こった PE はこれでチェックできます。

“batch.out” の 1 行目は、

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

[23] Version 2.85 以前は batch.now というファイル名でした。

となっています。この最初の数字はその時点での残りバッチ数です。この数字を減らして batch.out を保存することにより、任意のバッチで計算を終了させることができます。例えば、“3” にするとその時点から 3 バッチ分計算を実行した後で終了しますし、“0” にするとその時のバッチで計算が終了します。この機能により、途中経過を見ながら計算を中断することができます。

ただし、デフォルト設定では、計算途中のタリー結果の画像ファイル（eps 形式）は出力されません。各タリーにおいて `epsout` を設定し（6.7.16 節を参照）、`[parameters]` セクションにおいて `itall=1` とすることで、途中結果を画像ファイルで確認することができます。

バージョン 2.86 以降、`[parameters]` セクションにおいて `file(22)` を指定することにより、“batch.out” のファイル名を変更できるようになりました。これにより、同一ディレクトリにおいて複数のインプットファイルに対して PHITS を実行することが可能となりました。

“batch.out” に出力される `rijk` は、各バッチ毎 (各 IP number 毎) に与えられた初期乱数の値です。特定のバッチで異常終了が起こった時など、任意のバッチの計算を再現したい場合にこの値を利用できます。

3.3 タリー結果と統計誤差の推移を表示する機能

`[parameters]` セクションで `itall=3` とし、各タリーセクションで anatally サブセクションを設定することにより、そのタリーの結果と統計誤差がヒストリー数（或いはバッチ数）によってどのように推移するかを表示させることができます。計算を始めて統計量が少ない時は、タリー結果や統計誤差が大きく変化する場合があるため、統計誤差が小さい結果を得たからといってもその値が信頼できるかどうか判断できません。そこで、ヒストリー数の増加に対する各値の推移状況を調べることで、適切な結果が得られているかどうかを確認するのが本機能の目的です。

タリーセクションで指定したファイル名の拡張子の前に “_StD” を付けたファイルを作成し、各バッチ終了時に出力されるタリー結果とその統計誤差を出力します。バッチ終了時に更新されて、その時点までのタリー結果と統計誤差の推移を確認することができます。

ただし、この機能は現在のバージョンでは `[t-track]` と `[t-point]` のみ動作します。

各タリーセクション内に下のような anatally サブセクションを設定することで、タリー結果と統計誤差を出力させるメッシュを指定することができます。なお、6.9 節で、複数のタリー結果の解析機能を使用する際にも anatally サブセクションを設定しますが、ここで説明している `itall=3` の場合とは使用できるパラメータや考え方が違うのでご注意ください。

例 1: 出力させるメッシュを指定する場合

```
1:  anatally start
2:  ix = 89 91
3:  iz = 61 81 101
4:  iy = 1
5:  ipart = 1
6:  anatally end
```

Anatally サブセクションは、`anatally start` と `anatally end` で挟んだ領域で設定します。もし、これらの 2 行のみでメッシュの指定が無い場合は、全てのメッシュ点の結果を出力します。`ix,iz,iy` はそれぞれ `x-type`，`z-type`，`y-type` で定義した各変数のメッシュ点を指定するためのパラメータであり、この例の場合、x について 89 と 91 番目のメッシュ点を指定しています。指定範囲は各メッシュの全ての組み合わせとなっており、上の例では x について 2 点と z について 3 点指定していることから、合計 2*3=6 個のメッシュ座標についてタリー結果を出力させます。`ipart=1` は `part=`で定義した粒子の内、最初の粒子のみ出力させることを意味しており、例えば `part = proton neutoron` としていた場合、proton に関するタリー結果のみを出力させることになります。

メッシュを指定する変数には、次の 10 種類があります。`ireg`: 領域（セル）、`ix`，`iy`，`iz`，`ir`，`ie`，`it`，`ia`: `x-`，`y-`，`z-`，`r-`，`e-`，`t-`，`a-type` サブセクション、`ipart`: `part` パラメータ、`imul`: `multiplier` パラメータ。省略した変数については、全てのメッシュ点を出力させます。明示的に `all` を指定することもできます。

4 入力ファイルの書式

新しい入力ファイルは、[******] で始まる次の様な各セクション単位でデータを読み込みます。セクションの順番は自由です。[******] の先頭の空白は 4 個まで無視しますが、それ以上はセクションの始まりとは認識しませんので注意して下さい。

4.1 セクションの種類

セクションには、次の様なものがあります。

表 1: セクションの種類 (1)

name	説明
[title]	計算のタイトルを定義します。
[parameters]	ヒストリー数など、計算の動作を決めるパラメータを定義します。
[source]	入力ソース粒子の情報を定義します。
[material]	体系を構成する物質を定義します。
[surface]	体系を構成する面を定義します。
[cell]	体系を構成する cell を定義します。
[transform]	体系を構成する面などの座標変換を定義します。
[temperature]	cell の温度を定義します。
[mat time change]	物質の時間変化を定義します。
[magnetic field]	体系を構成する領域の磁場を定義します。
[electro magnetic field]	体系を構成する領域の電磁混合場を定義します。
[delta ray]	δ 線を発生させます。
[track structure]	飛跡構造解析モードを使用するための設定を定義します。
[super mirror]	低エネルギー中性子のスーパーミラーを定義します。
[elastic option]	低エネルギー中性子の弾性衝突のオプションを定義します。
[data max]	核種毎に核データライブラリを利用する上限エネルギーを定義します。
[frag data]	ユーザー指定断面積読み込み機能について定義します。
[importance]	領域の importance を定義します。
[weight window]	領域の weight window を定義します。
[ww bias]	[weight window] のパラメータにかけるバイアスを定義します。
[forced collisions]	強制衝突の領域とファクターを定義します。
[Repeated collisions]	核反応の繰返し衝突の領域とファクターを定義します。
[volume]	体系を構成する領域の体積を定義します。
[multiplier]	multiplier を定義します。
[mat name color]	物質表示の名前、色、大きさを指定します。
[reg name]	領域表示の名前、大きさを指定します。
[counter]	counter を定義します。
[timer]	粒子の固有時間をリセット、ストップを定義します。

表 2: セクションの種類 (2)

name	説明
[t-track]	粒子の飛跡長 (track length) やフルエンスを導出するタリー。
[t-cross]	粒子の面横断回数やフルエンスを導出するタリー。
[t-point]	ある点や線上のフルエンスを導出するタリー。
[t-deposit]	物質におけるエネルギー付与を導出するタリー。
[t-deposit2]	2 つの領域でのエネルギー付与の相関を出力するタリー。
[t-heat]	物質におけるエネルギー付与を導出するタリー。（非推奨[24]）
[t-yield]	残留核の生成量を導出するタリー。
[t-product]	線源や核反応による生成粒子を導出するタリー。
[t-dpa]	原子あたりのはじき出し数 (DPA) を導出するタリー。
[t-let]	LET の関数として飛跡長や線量を導出するタリー。
[t-sed]	微小領域におけるエネルギー付与分布を導出するタリー。
[t-time]	Energy cut off と escape 粒子の個数を導出するタリー。
[t-interact]（従来の [t-star]）	反応数を導出するタリー。
[t-dchain]	DCHAIN 用入力ファイルを作成するタリー。
[t-wwg]	[weight window] のパラメータを出力するタリー。
[t-wwbg]	[ww bias] のパラメータを出力するタリー。
[t-volume]	体積自動計算機能のためのタリー。
[t-userdefined]	ユーザー定義による任意の物理量を導出するタリー。
[t-gshow]	ジオメトリ（仮想空間）を 2 次元で表示するタリー。
[t-rshow]	ジオメトリを 2 次元で物理量による色分けをして表示するタリー。
[t-3dshow]	ジオメトリを 3 次元で表示するタリー。
[end]	入力ファイルの終了を示します。

これらのセクション単位でデータを読み込みます。従って、入力ファイルの先頭から最初のセクションまで、また [end] 以下の部分は読み飛ばします。

4.2 読み込みコントロール

大文字、小文字、空白文字 (␣)

ファイル名を除いて、大文字小文字は区別しません。行頭、行末の空白文字は無視します。セクション名は間に空白文字を入れてもかまいません。ただし、[******] の先頭の空白文字は 4 個まで無視しますが、それ以上はセクションの始まりとは認識しませんので注意して下さい。

タブ

タブは、8 個のブランクに置き換えます。

継続行

一行に書くことができる文字数は半角文字で 200 字までです。それ以降の文字は読み込まれません。一行に収まらないデータは、行末に “¥” をつけると次行を含めて一行と扱います。複数行が使用でき

[24] バージョン 3.04 までは [t-deposit] によるカーマ近似計算ができなかったため、この計算が可能な [t-heat] との使い分けを行っていました。

ます。ただし、[cell] と [surface] の継続行は “¥” を使用せず、先頭に 5 個以上の空白文字を入れてください。

行連結

短いパラメータ文が続くとき、複数行を “;” でつなぐことにより一行に表示することができます。例えば、

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

のよう書くことができます。ただし、メッシュの記述のようにサブセクションの書式が決まっている場合は使用できません。

コメント文字

コメント文字として#, %, !, $の 4 文字が使用できます。これらの文字以降行末まで無視されます。ただし、[surface], [cell] セクションの場合は、コメント文字として使用できるのは$のみですので注意してください。他に、各行の最初に “c␣” （␣ は空白文字）と書くと、その行はコメント行となり、読み飛ばされます。ただし、左から 5 カラムまでに書いてください。また、この機能により、[material] セクションで炭素の自然同位体を定義するときに C を用いると、コメント行として扱われる場合があります[25]。この場合は、6000、で指定するようにしてください。

空白行

空白行は、読み飛ばします。コメント文字で始まる行も読み飛ばします。

セクションの読み飛ばし

ひとつのセクションを読み飛ばしたいとき、[******] off のようにセクション名の後に “off ” をつけると、次のセクションまでを読み飛ばします。直後に書かれた infl:や set:が意図せず読み飛ばされる場合がありますのでご注意ください。

セクション途中からの読み飛ばし

セクションの途中で、行頭に qp: を挿入すると、この行以降次のセクションまでを読み飛ばします。

全ての読み飛ばし

入力ファイルのどこでも、q: を行頭に入れると、それ以降全てを読み飛ばします。これは、[end] と同じです。

4.3 ファイルの挿入

入力ファイルのどの場所でも他のファイルを取り込めます。インクルードファイルの書式は、

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

です。{ } 内がファイル名、[ ] 内が行数指定です。n_1 行から n_2 行をインクルードします。行数指定は省略できます。省略した場合は、そのファイル全てをインクルードします。また、

[25] バージョン 2.89 より、デフォルトでは [material] セクションにおいて c をコメント文字として使用できなくなりました。コメント文字として使用したい場合は、[parameters] セクションにおいて icommat=1 としてください。ただし、icommat=1 を含む [parameters] セクションが c を使用している [material] セクションより前に書かれている必要があります。

[n_1-]
[$-n_2$]

という表現も可能です。上が n_1 行から最後まで、下が先頭から n_2 行までをインクルードします。インクルードは、9 回までネストできます。インクルードファイルを読み終えると、一層上のファイルに戻ります。

Windows 系の環境でコマンドラインインタプリタ (コマンドプロンプト) により PHITS を実行している場合に、infl を使用する際は注意が必要です。このときは、インプットファイルの 1 行目を

```
file = input.dat
```

としてください。ここで、input.dat はインプットファイルです。コマンドラインインタプリタを用いた実行方法については、2.1.2.2 節をご参照ください。

4.4 ユーザー定義定数

入力ファイルの数字の入力箇所で、ユーザーが定義した定数が利用できます。定数のセットの書式は以下の通りです。

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

ただし、c*i*[] において、*i* と [の間に空白文字を入れることができないのでご注意ください。この定数セットは、どの場所でも可能です。定数名は、c1 から c99 まで、何度でもセットし直せます。セットした以降、次にセットされるまで、その値を保持します。定義の中で他の定数を用いる場合 (上の 3 番目の例)、その時点での値が参照され、それ以降、定義の中で用いられている定数 (上の 3 番目の例では c1) をリセットしても定数の値は変化しません。pi は、デフォルトで定義されている定数です。

Sumtally サブセクション内で定義した定数は無視されるのでご注意ください。

ユーザー定義プログラム (“usrsors.f, usrtally.f, usrdfn1.f” など) の中でこれらの変数を使いたい場合は、

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

と宣言した上で、cval として利用できます。例えば、c1 パラメータは cval(1) に相当します。なお、複数回同じパラメータを定義した場合、インプットファイルの最後に定義した値がユーザー定義プログラム内で有効となります。

4.5 数式の利用

入力ファイルで数字の入力箇所で、数式の利用が可能です。数式の書式は、Fortran 形式です。利用できる内部関数は、

例として、

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

表 3: Intrinsic Function.

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

この例のように、`param =` の後に 1 個の数字が期待されるときは、数式の途中に空白が許されますが、幾つかの数字の並びが期待される場合は、空白が数字の区切りを示しますので空白は使えません。このときは、空白を省いて記述するか、`{ c1 * 2 / pi }` のように { } で括ってください。

4.6 粒子の表式

`[source]` セクションや、タリーセクションで用いる 粒子の表式を下の表にまとめます。粒子を定義するときは、symbol か kf-code を用います。symbol のないものは、kf-code だけ利用可能です。

粒子タイプ 11 の other particles は、以下のように kf コードで指定します。また、これらの粒子の崩壊チャンネルは以下のようなものが含まれています。

JQMD コードの導入に伴い、原子核の入射、輸送が可能となりました。原子核種の指定の仕方は、208Pb, 56Fe の形式です。Pb, Fe と指定すれば、同位体全てを意味します。もちろん入射粒子の指定には使えません。`ityp=19` の `nucleus` は、全ての原子核から `ityp=1,15,16,17,18`、すなわち `proton, deuteron, triton, 3He, alpha` を除いた全てを意味します。kf コードで原子核を指定するには、kf = Z * 1000000 + A とします。

以前は、光子の名前を “gamma”としてきましたが、“photon”と変更しましたので注意して下さい。

表 4: List of the 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	22	γ
15	deuteron	1000002	deuteron
16	triton	1000003	triton
17	3he	2000003	$^3\mathrm{He}$
18	alpha	2000004	α
19	nucleus	Z*1000000+A	nucleus

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	−	+ − 221	$\eta\ \overline{\eta}$
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	$\Omega^-\ \overline{\Omega^-}$

表 5: 崩壊チャンネルと寿命

			分岐率	寿命 (sec)
n	$\rightarrow$	$p + e^- + \bar{\nu}_\mathrm{e}$	100%	8.867×10^2
π^0	$\rightarrow$	$\gamma + \gamma$	100%	0
π^+	$\rightarrow$	$\mu^+ + \nu_\mu$	100%	2.6029×10^{-8}
π^-	$\rightarrow$	$\mu^- + \bar{\nu}_\mu$	100%	2.6029×10^{-8}
μ^+	$\rightarrow$	$e^+ + \nu_e + \bar{\nu}_\mu$	100%	2.19703×10^{-6}
μ^-	$\rightarrow$	$e^- + \bar{\nu}_e + \nu_\mu$	100%	2.19703×10^{-6}
K^0	$\rightarrow$	$\pi^+ + \pi^-$	68.61%	8.922×10^{-11}
	$\rightarrow$	$\pi^0 + \pi^0$	31.39%	
	$\rightarrow$	$\gamma + \gamma$	other	
K^+	$\rightarrow$	$\mu^+ + \nu_\mu$	63.51%	1.2371×10^{-8}
	$\rightarrow$	$\pi^+ + \pi^0$	other	
K^-	$\rightarrow$	$\mu^- + \bar{\nu}_\mu$	63.51%	1.2371×10^{-8}
	$\rightarrow$	$\pi^- + \pi^0$	other	
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	

4.7 配列の大きさの変更

インクルードファイル “param.inc“の中に、ユーザーが場合により変更しなければならない配列の大きさが記述されています。特に重要なのが mdas です。これにおいて、幾何形状、タリー、 核データ関係、バンクの配列に必要な配列の大きさを指定します。インプットエコーを見て現在の使用量をチェックしてみてください。バンクの配列の大きさは、[parameters] で指定します。mdas の余った配列は、バンクが足りなくなった場合に自動的に使われます。

以下に現在のデフォルト値が記述された “param.inc“を示します。

ファイル 1: param.inc

```
************************************************************************
*                                                                      *
*     'param.inc'                                                      *
*                                                                      *
************************************************************************

      parameter ( mdas   = 80000000 )
      parameter ( kvlmax =     3000 )
      parameter ( kvmmax =  1000000 )
      parameter ( itlmax =      200 )
      parameter ( inevt  =       70 )
      parameter ( isrc   =      500 )
      parameter ( nbchmax=    10000 )
      parameter ( multmax=      500 )

      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 *
*                                                                      *
*----------------------------------------------------------------------*
```

5　セクション書式

5.1　[Title] セクション

このセクションでは、計算のタイトルを定義します。タイトルは、標準出力の先頭に印字されます。書式は次の様なものです。何行でもかまいません。ただし、このセクションでは空白行は無視されます。

```
[ T i t l e ]
This is a test calculation of PHITS.
Any number of title lines are allowed.
..........
```

5.2 [Parameters] セクション

このセクションでは、PHITS の動作を決定するパラメータを指定します。書式は次の様なものです。

```
[ P a r a m e t e r s ]
para1 = number | file.name
para2 = number | file.name
..........
```

パラメータの順番は自由、省略すれば、デフォルト値が読み込まれます。
以下に各々のパラメータの意味と値を説明します。(D=) は、デフォルト値です。

5.2.1 計算モード

表 6: icntl パラメータ

パラメータ	値	説明
icntl	(D=0)	基本動作オプション。
	= 0	通常の粒子輸送計算。
	= 1	断面積・カーマファクター出力モード（utility\icntl1 参照）
	= 3	input echo の書き出し。メモリー空間、ライブラリー、ファイルのチェック用。
	= 5	全てを真空 (void) 領域とし、粒子の反応や散乱、電離等が起こらない輸送計算を実行する。geometry check や体積・面積計算の際に使用する。
	= 6	ソースのチェック。[t-product] で線源粒子をタリーできる。
	= 7	[t-gshow] タリーの実行。
	= 8	xyz メッシュタリーの gshow オプションのあるものの幾何形状表示。
	= 9	[t-rshow] タリーの実行。
	= 10	reg メッシュタリーの rshow オプションのあるものの幾何形状表示。
	= 11	[t-3dshow]（3 次元幾何形状表示タリー）の実行。
	= 12	dumpall ファイルから読み込んで再計算。file(15) で出力ファイルを指定。
	= 13	タリー結果の統合機能（Sumtally 機能）の利用。
	= 14	体積自動計算機能の利用。
	= 15	[t-wwbg] タリーの実行。
	= 16	分析機能スクリプトの利用。
	= 17	[anatally] セクションの実行。

icntl=1 は、従来、核反応計算モードとして使っていましたが、現在は、断面積や Kerma factor などを出力する目的でのみ利用可能です。詳しくは “/phits/utility/icntl1” フォルダをご参照ください。

icntl=12 を指定すると、dumpall=1 で計算した時にダンプしたデータを基に再計算します。輸送に関しては、最初に計算した全ての情報を再現します。再計算の時には、最初に計算したインプットファイルが必要になります。タリーを除いて同じものを使わないと矛盾が生じます。maxcas，maxbch は、ダンプファイルに指定されているものを使いますので変更できません。最初に計算したときと異なるタリーを入れて再計算したいときなど有効です。ただし、dumpall=1 で計算するときは、巨大なデータファイルが作成されることがありますから注意してください。dumpall=1 でダンプする情報は、5.2.25 で解説します。

icntl=13 を指定することにより、Sumtally 機能を利用することができます。この機能の詳細は 6.8 をご覧ください。

icntl=14 を指定することで、[t-volume] による体積自動計算機能を利用することができます。この機能の利用方法は 7.18 をご覧ください。

icntl=15 を指定することで、[ww bias] のパラメータを自動生成する [t-wwbg] が機能します。この機能の利用方法は 7.17 をご覧ください。

分析機能スクリプトを利用する場合は、icntl=16 と指定する必要があります。

5.2.2 ヒストリー数、バンク配列の大きさ

表 7: ヒストリー数、バンク配列の大きさに関するパラメータ

パラメータ	値	説明
maxcas	(D=10)	1 バッチのヒストリー数。上限は 2147483647。
maxbch	(D=10)	バッチ数。上限は 2147483647。
maxbnk	(D=10000)	一時保存できる粒子の数。
timeout	(D=-1.0)	PHITS 計算の打ち切り時間 (sec)。
istdev	(D=0)	新規 / 再開始計算及び統計誤差導出方法のコントロール。
	=-2	再開始計算 (ただし、過去のタリー結果が全くない場合は、istdev=2 として新規計算を開始。
	=-1	再開始計算 (ただし、過去のタリー結果が全くない場合は、istdev=1 として新規計算を開始。
	=0	新規計算。統計誤差は、メモリ共有型並列で italsh=1 の場合はバッチ分散を、それ以外はヒストリー分散を自動的に選択します。
	=1	新規計算。統計誤差は、各バッチ間の分散の平方根として導出。
	=2	新規計算。統計誤差は、各ヒストリー間の分散の平方根として導出。
italsh	(D = 0)	OpenMP（メモリ共有型並列計算）を利用した場合にスレッド間のタリー変数の共有/非共有を切り替えるオプション。タリー変数を共有すれば、少ない物理メモリでも並列計算が実行できるため、大容量のタリー領域を必要とする並列計算の実行で有用です。一方で、メモリ競合を防止するプロセスのために計算速度が低下する可能性があります。少ないヒストリー数で計算速度等の確認をした上でお使いください。また、メモリ共有型並列で italsh=1 の場合、タリー変数をスレッド毎に用意しないため、統計誤差はバッチ分散 (istdev=1) でしか計算できません。
	= 0	タリー変数をスレッド間で共有しない。
	= 1	タリー変数をスレッド間で共有する。
ireschk	(D=0)	再開始計算時のタリーパラメーター一致性確認の有無。
	=0	読み込むタリー出力の設定が現在のタリー設定と一致しているか確認する。
	=1	上記確認を行わない。極めて長いタリー設定をした場合、タリー出力に全ての設定が書き込まれずに、新旧ファイル一致性の確認でエラーが出てしまう場合があります。その場合、ireschk=1 とし、一致性を確認しないモードで再開始計算を実行してください。

メモリ分散型並列計算では、バッチ単位で並列化しています。このため、入力のバッチ数 maxbch が $N_{\mathrm{PE}}-1$ の整数倍となるように指定してください。ここで、N_{PE} は並列計算で使用する実行 PE(Processing Element) 数です。上記の -1 は、並列計算のコントロールのために 1PE 使うためです。もし整数倍でないときは、maxbch が整数倍になるように変更し、加えて全ヒストリー数が入力した数と同程度となるように maxcas を自動で調整します。この自動調整が実行された場合は、出力のインプットエコーの最後にコメントが書き出されます。

timeout を用いることにより、PHITS 計算の打ち切り時間を指定できます。timeout が正の場合に機能します。バッチ終了時の CPU 時間が timeout を超えた場合に、そのバッチで計算を終了します。バッチの途中で timeout を超えた場合は、そのバッチが終了するまで計算が継続します。メモリ分散型並列の場合は、全ての CPU 時間の和で判定します。ただし、ご使用の計算機によっては CPU 時間が正しく取得できず、timeout を適切に使用できない場合がありますのでご注意ください。

バージョン 2.50 より、タリー出力の統計誤差が正しく計算できるようになりました。また、統計が十分でなかった場合などに対応するため、古い計算結果から次の初期乱数とタリー出力を読み込んで再開始計算を行う機能を実装しました。

統計誤差計算は、各バッチ間のタリー量の分散の平方根として標準偏差を導出するバッチ分散モードと、各ヒストリー間のタリー量の分散の平方根として標準偏差を導出するヒストリー分散モードがあります。どちらも、次式で標準偏差 σ を計算し、その値とタリー量の平均値 $\overline{X}$ の比を相対誤差として出力します。

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

ここで N は標本数 (istdev=1 のときは全バッチ数、istdev=2 のときは全ヒストリー数) であり、x_i, w_i は各標本のタリー量及びソースウェイト、$\overline{w}$ はソースウェイトの平均値です。メモリ共有型並列計算の場合は、原理的に各ヒストリーの分散を計算することが不可能ですので、バッチ分散モードのみ選択可能です。バッチ分散の場合は、同じ全ヒストリー数でも、maxcas と maxbch の組み合わせによって標準偏差が変化しますのでご注意ください。基本的には、maxbch を大きくした方が計算の信頼性は高くなりますので、正確に統計誤差を計算したい場合は、maxbch を少なくとも 10 以上としてください。ただし、PHITS は、各バッチ終了時にいろいろな処理をしますので、maxbch を大きくしすぎると計算時間が長くなりますのでご注意ください。一方、ヒストリー分散モードの場合は、標準偏差は全ヒストリー数のみに依存し、maxcas と maxbch の組み合わせには依存しません。したがって、メモリ共有型並列を行わない場合は、ヒストリー分散モードを選択することをお薦めします。しかしながら、xyz メッシュタリーを細かく設定する等、メモリを多く使用するタリーを設定している場合には、ヒストリー分散モードでの計算時間が莫大となることがありますのでご注意ください。なお、各ヒストリーの分散を導出するタリー ([t-deposit] タリーの output = deposit や [t-deposit2] など) の場合は、統計誤差として標準誤差を出力しており、istdev の値に関係なく、寄与したヒストリー数の平方根から計算します。すなわち、寄与したヒストリー数が K の場合、その相対誤差は $1/\sqrt{K}$ となります。

相対誤差は、通常、タリー量の右側の r.err カラムに出力されます。ただし、axis = xy, rz など２次元プロットの場合は、各タリーで指定した file 名の拡張子の手前に_err と付けた別ファイルに出力します。例えば、file=tally.out と指定した場合、相対誤差ファイルは tally_err.out となります。誤差ファイルの書式は通常のタリー出力と同じですので、ANGEL で処理することにより誤差の 2 次元プロットを作成することができます。

PHITS で評価しているのは標準偏差や標準誤差なので、必ずしも真値がこの範囲内に収まるわけではないのでご注意ください。平均値を中心として $\pm1\sigma$ の範囲は約 68%の信頼区間を表しており、この範囲に約 68%の確率で計算結果が含まれることを意味します。また、核反応モデルに起因する系統的な不定性は含まれていません。

istdev < 0 とした場合は、再開始計算となります。その場合、統計誤差計算モードは、過去のタリー結果を踏襲します。再開始計算については、3.1 をご覧ください。

表 8: 乱数に関するパラメータ

パラメータ	値	説明
nrandgen	(D=1)	擬似乱数生成法オプション。
	=0	線形合同法を使用する。乱数の周期は 2^{46}。詳細は下記文献[26]を参照。
	=1	xorshift64 を使用する。乱数の周期は $2^{64}-1$。詳細は下記文献[27]を参照。
irskip	(D=0)	乱数をコントロールするオプション。
	irskip>0	irskip 回ヒストリーをスキップして計算を開始 (デバッグ用)
	irskip<0	irskip 回ヒストリーの乱数をスキップして計算を開始 (手動並列用)
rseed	(D=0.0)	初期乱数オプション
	rseed=0	デフォルトの初期乱数 6.647299061401E+12 を使用する。
	rseed≠0	rseed を初期乱数とする。nrandgen=0 の時は絶対値の整数値を使用する。
itimrand	(D=0)	時間依存の初期乱数オプション。
	=0	rseed により決まる値を使用する。
	=1	計算の開始時間情報より決まる初期乱数を使用する。
bitrseed		初期乱数のビットデータ（0 か 1 で表される 64 桁の整数列）。デフォルトは 6.647299061401E+12 のビットデータ。

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

5.2.3 計算打切エネルギー、モデル切り替えエネルギー

表 9: 計算打切エネルギーに関するパラメータ

パラメータ	値	説明
emin(i)	(D=1.0e-3) (D=1.0e-11) (D=1.0e-3) (D=1.0) (D=1.0e+9) (D=1.0e-3) (D=1.0e-3)	(i=1) 陽子の計算打切エネルギー [MeV]。 (i=2) 中性子の計算打切エネルギー [MeV]。 (i=3-10) パイオン、ミューオン、ケイオンの計算打切エネルギー [MeV]。 (i=11) その他の粒子 (ityp=11) の計算打切エネルギー [MeV]。 (i=12,13) 電子 (i=12)、陽電子 (i=13) の計算打切エネルギー [MeV]。 (i=14) 光子の計算打切エネルギー [MeV]。 (i=15-19) 重陽子 (i=15)、3 重陽子 (i=16)、^{3}He(i=17)、^{4}He(i=18)、原子核 (i=19) の計算打切エネルギー [MeV/n]。ここで、MeV/n は核子あたりの MeV。 粒子番号 i は表 4 を参照。
dmax(i)	(D=emin(i)) (D=20.0) (D=1.0e+3)	(i=1,3-13,15-19) 粒子番号 i の粒子に関するライブラリーを利用する際の上限エネルギー [MeV]。(ただし、i=15-19 の場合は [MeV/n]。) (i=2) 中性子に関するライブラリーを利用する際の上限エネルギー [MeV]。 (i=14) 光子に関するライブラリーを利用する際の上限エネルギー [MeV]。 PHITS のパッケージに含まれているライブラリーは i=1,2,12,13,14 のみなので（ただし、i=1 については一部の核種のみ)、これら以外の粒子についてはご自身でライブラリーを用意して使用してください。
esmin esmax	(D=0.001) (D=300000)	電子・陽電子を除く荷電粒子の range 計算の最小エネルギー [MeV/n]。 電子・陽電子を除く荷電粒子の range 計算の最大エネルギー [MeV/n]。
cmin(i)	(D=1.0)	粒子番号 i の粒子の反応計算打切エネルギー [MeV]。(ただし、i=15-19 の場合は [MeV/n]。) cmin(i) 未満のエネルギーで起こる反応や散乱（δ 線生成も含みます）は無視する。 ただし、中性子、電子・陽電子、光子のデフォルト値は emin(i) です。
etsmin etsmax	(D=1e-6) (D=1e-2)	飛跡構造解析を実施する電子（もしくは陽電子）のエネルギーの下限値 [MeV]。 飛跡構造解析を実施する電子（もしくは陽電子）のエネルギーの上限値 [MeV]。
tsmax	(D=1e-3)	陽子やイオンに対する飛跡構造解析を行う最大エネルギー [MeV/n]。ただし、断面積ライブラリがない場合は、ATIMA など通常の阻止能モデルを使って電離エネルギー損失を計算します。現在のところ、1 keV – 300 MeV の陽子と 1 keV/n – 10 MeV/n の炭素イオンに対する断面積データライブラリが PHITS に組み込まれています。各自でコンパイルした PHITS で陽子やイオンに対する飛跡構造解析を行いたい場合は、11.4 節をご参照ください。

emin，esmin，cmin，etsmin によって各々のエネルギー領域に制限を与える際は、その値未満となります。したがって例えば、emin(1) で陽子の計算打切エネルギーを指定した場合、丁度その値のエネルギー (デフォルトだと 0.001 MeV) をもつ陽子は計算打切とはなりません。一方、dmax，esmax，etsmax の場合は、その値以下が対象となります。

輸送粒子のエネルギーが emin より低くなると、エネルギーカットオフとなり、輸送計算が打ち切られま

す。その際、残りの運動エネルギーはその場所に付与され、中性子を除く表 5 の粒子は崩壊します。また、陽電子は対消滅を起こします。

emin(2) や dmax(2) を指定しない場合、これらの値は nucdata オプションによって自動的に調整されます。例えば、nucdata=0 の場合、emin(2)=dmax(2)=1.0e-3 となります。また、emin(12,13) や dmax(12,13) を指定しない場合、これらの値は negs オプションによって自動的に調整されます。例えば、negs=1 の場合、emin(12,13)=0.1、dmax(12,13)=1.0e+3 となります。詳細は、表 10 をご覧ください。

粒子のエネルギーを E とした場合に、emin< E <dmax の範囲にあるエネルギーをもつ粒子の反応や散乱がライブラリーにより計算されます。emin>dmax とすれば、ライブラリーを用いた計算をしません。中性子と光子のライブラリーの上限は、それぞれ 20 MeV と 100 GeV です。また、EGS5 を利用しない場合の電子と陽電子のライブラリーの上限は 10 GeV となります。

dmax の値がライブラリの上限エネルギーよりも高い値に設定された場合は、上限エネルギーの断面積の値を dmax まで使います。例えば、通常の JENDL-4.0 を使う場合、dmax=200 と設定すると、20 MeV から 200 MeV までは 20 MeV の断面積が使用されます。また、ライブラリによって上限が違う場合は [data max] セクションで個別に定義してください。

エネルギー E の荷電粒子について物質中の飛程 (range) を計算する際、esmin< E <esmax の範囲でデータテーブルを作成して計算します。より小さい、もしくは、より大きなエネルギーを取り扱いたい場合には、esmin や esmax を設定してください。通常はデフォルト値のままで計算を行います。

荷電粒子の計算打ち切りエネルギー emin を esmin よりも小さくすることはできません。その場合、emin は自動的に esmin の値に修正されます。

[Frag Data] で考慮したい最小の入射エネルギーに応じて、cmin(i) を設定してください。

etsmin は、原理的には 1 meV まで設定可能ですが、計算時間が膨大になりますので、1 eV 程度に設定することをお勧めします。etsmax は、1 keV よりも大きく設定する必要があります。原理的には極めて大きい値も設定可能ですが計算時間が膨大になりますので、現実的には 100 keV 以下に設定することをお勧めします。飛跡構造解析の場合は、必ず EGS5 モードを用いて emin(12-13) を 1 keV に設定する必要があります。

表 10: 光子・電子・陽電子・中性子の輸送に関するパラメータ

パラメータ	値	説明
negs	(D=-1)	光子・電子・陽電子の輸送に関するオプション。
	=-1	PHITS オリジナルモデルを用いて光子のみ輸送する。デフォルト値である emin(14)=0.001, dmax(14)=1000.0 が使用されるが、これらのパラメータを指定した場合は、それらの値が優先される。
	= 0	電子・陽電子・光子の輸送を行わない。emin(14)=dmax(14)=1.0e+9 に自動調整される。
	= 1	EGS5 を用いて光子・電子・陽電子を輸送する。emin(12,13)=0.1, dmax(12-13)=1000.0 に自動調整されるが、これらのパラメータを直接指定した場合は、それらの値が優先される。このオプションを使用する場合は、file(1) もしくは file(20) の設定が必要になる。
nucdata	(D=1)	中性子に関するライブラリーの使用についてのオプション。
	= 0	中性子に関するライブラリーを使用しない。emin(2)=dmax(2)=1.0e-3 に自動調整し、中性子の輸送計算をライブラリーを使用せずに実行する。ただし、emin(2) や dmax(2) を直接指定した場合は、それらの値が優先される。
	= 1	中性子断面積データライブラリー JENDL-4.0 に合わせたデフォルト値 (emin(2)=1.0e-11, dmax(2)=20.0) を使用する。ただし、emin(2) や dmax(2) を直接指定した場合は、それらの値が優先される。このオプションを使用する場合は、file(1) もしくは file(7) の設定が必要になる。
ieleh	(D=0)	電子、陽電子の輸送オプション。
	= 0	エネルギー e が dmax(12) 以上の場合、減速や反応を行わない。
	= 1	エネルギー e が dmax(12) 以上の場合、ウエイト値 wgt を wgt= e/dmax(12) に変更し、dmax(12) の値をエネルギーとして輸送計算を実行する。

表 11: モデル切り替えエネルギーに関するパラメータ

パラメータ	値	説明
ejamnu	(D=20.)	核子の核反応モデルに関する Bertini（もしくは JQMD）と JAM への切り替えエネルギー [MeV]。
ejampi	(D=20.)	パイオンの核反応モデルに関する Bertini と JAM の切り替えエネルギー [MeV]。
eisobar	(D=0.0)	isobar=1 の時の isobar モデルの上限エネルギー [MeV]。
isobar	(D=0)	isobar モデルのオプション
	= 0	isobar モデルを用いない
	= 1	isobar モデルを用いる
eqmdnu	(D=20.)	核子の核反応モデルに関する Bertini と JQMD の切り替えエネルギー [MeV]。
eqmdmin	(D=10.)	JQMD 適用の下限エネルギー [MeV/n]。
ejamqmd	(D=3000.)	原子核の核反応モデルに関する JQMD と JAMQMD の切り替えエネルギー [MeV/n]。
inclg	(D=1)	核子、パイオン、軽イオン ($d,t,^3$He,α) の核反応モデルとして INCL を適用する場合のオプション。
	=0	INCL を使用しない。核子の反応については、ejamnu と eqmdnu の値に応じて、JAM、JQMD、Bertini のモデルが適用される。パイオンの反応については、ejampi の値に応じて、JAM か Bertini が適用される。軽イオンの反応については JQMD が適用される。
	=1	核子、パイオン、軽イオンが関与する核反応で INCL を使用する。
	=2	核子とパイオンが関与する核反応で INCL を使用する。
einclmin	(D=1.0)	INCL が適用される下限のエネルギー [MeV/n]。
einclmax	(D=3000.0)	INCL が適用される上限のエネルギー [MeV/n]。
incelf	(D=0)	核子が関与する核反応モデルとして INC-ELF を使用する。
	=0	INC-ELF を使用しない。
	=1	INC-ELF を使用する。
eielfmin	(D=1.0)	INC-ELF が適用される下限のエネルギー [MeV]。
eielfmax	(D=3500.)	INC-ELF が適用される上限のエネルギー [MeV]。
irqmd	(D=0)	核反応モデルとして JQMD か JQMD-2.0 を使用する。
	=0	JQMD を使用する。
	=1	JQMD-2.0 を使用する。
iscinful	(D=0)	0.1–150 MeV の中性子と炭素の核反応計算に SCINFUL-QMD のデータベースを適用する場合のオプション。
	=0	SCINFUL-QMD のデータベースを使用しない（核反応モデルを使う）。
	=1	SCINFUL-QMD のデータベースを使う。この場合 [data max] セクションで炭素のライブラリ上限エネルギーを 0.1 MeV に設定する必要があります。詳しくは utility\usrtally\scinful-qmd をご参照ください。
Kerma	(D=0)	Kerma 計算に関するオプション。
	=0	このオプションを使用しない。
	=1	中性子や陽子の核データを用いた計算において、陽子以外の荷電粒子の生成を無視します。
epseudo	(D=10.0)	荷電粒子核データライブラリを使う際、ステップ幅中での最大断面積を与えるエネルギーを指定するパラメータ。このエネルギーを小さくすると、低エネルギー核反応の計算精度が多少上がりますが、計算時間が飛躍的に長くなります。

d, t, α 原子核を輸送している場合、`eqmdmin` 以下のエネルギーでは JQMD による原子核反応をさせません。低エネルギーでの JQMD の適用には限界がありますし、通常の物質中ですと低エネルギーでは飛程が短いので核反応を考慮しなくても影響は小さいです。

高エネルギーの原子核反応は、デフォルトで 3.0 GeV/n で、JQMD から JAMQMD モデルへと切り替わります。この切り替えエネルギーを `ejamqmd` で変えられます。核子入射反応でも、`eqmdnu`, `ejamnu`, とこの `ejamqmd` の値を調整すれば、JAMQMD モデルで計算することも可能です。

INCL (Intra-Nuclear Cascade of Liège) は、核子、パイオン、軽イオン入射反応を記述する核反応モデルです。バージョン 2.50 から、明示的にモデルの選択を行わない場合、これらの入射粒子の反応に対しては基本的にこのモデルが使用されるようになりました。もし、本モデルを用いて得られた結果を発表する場合は、必ず下記文献[28]を引用してください。

INC-ELF (Intra-Nuclear Cascade with Emission of Light Fragment) は軽イオン生成機構をもつ核反応モデルです。核子 (陽子、中性子) が入射反応の時に使用できます。本モデルを用いて得られた結果を発表する際は、必ず下記文献[29] を引用するようにしてください。

JQMD, JQMD-2.0 は核反応を記述するモデルで、特に重イオン入射反応を記述するのに優れています。バージョン 2.70 以降は、それまで使われていた JQMD に加えて、JQMD-2.0 も選択できるようになりました。JQMD-2.0 [30] は JQMD を改良し、反応の記述をより合理的にしたモデルで、周辺衝突反応の扱いが特に改善されています。ただし核反応の計算にかかる時間が 2 倍程度長くなります。

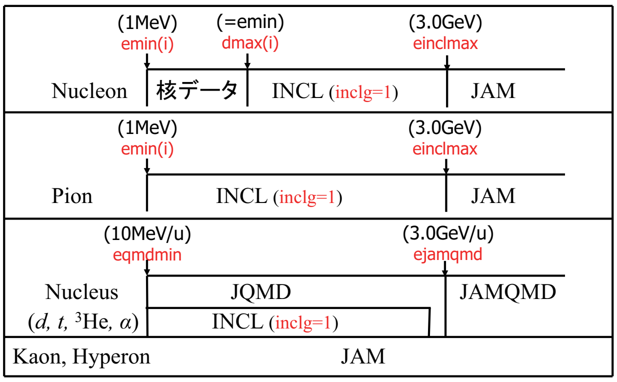


図 7: 核反応モデルの切り替えエネルギー

[28] A. Boudard, J. Cugnon, J.-C. David, S. Leray, and D. Mancusi, Phys. Rev C87, 014606 (2013).
[29] Y. Sawada, Y. Uozumi, S. Nogamine, T. Yamada, Y. Iwamoto, T. Sato, and K. Niita, Nucl. Instr. & Meth. B 291, 38-44 (2012).
[30] T. Ogawa, T. Sato, S. Hashimoto, D. Satoh, S. Tsuda, and K. Niita, Phys. Rev C92, 024614 (2015).

5.2.4 時間カット、ウエイトカット、ウエイトウインドウ

表 12: 時間カット、ウエイトカット、ウエイトウインドウに関するパラメータ

パラメータ	値	説明
tmax(i)	(D=1.e+9)	粒子番号 i(i=1-20) の粒子の計算打切時間 [nsec]。粒子番号は表 4 を参照。
wc1(i) wc2(i) swtm(i)	(D=-0.5) (D=wc1/2) (D=1.0)	i-th 粒子の minimum weight i-th 粒子の cutoff weight i-th 粒子の minimum source weight
wupn	(D=5)	ウェイトウィンドウの上限値 = [Weight Window] セクションで指定した下限値 × wupn. (wupn ≥ 2)
wsurvn	(0.6*wupn)	サバイバルウエイト値 (1 < wsurvn < wupn)
mxspln	(D=5)	スプリット数の最大値、サバイバルの最大倍数 (mxspln > 1)
mwhere	(D=0)	ウエイトウインドウのアクションの場所 -1: 核反応時、0: 両方、1: 境界横断時
iwwbias	(D=0) = 0 = 1	[ww bias] に関するオプション。 [ww bias] の値を用いない。 [ww bias] で与えたバイアスの値の逆数が [weight window] で指定したウエイトの下限値にかかる。
istdcut	(D=0) = 0 = 1	個別タリー打ち切り機能の切り替え。 統計誤差が stdcut 以下になっても計算終了時まで全てのタリー計算を続行します。なお，istdcut でタリーを打ち切った後に再開始計算を行うと，乱数の重複が起きてしまう可能性があります。再開始計算を行う場合は，前回のシミュレーションで最後まで実行されたタリーをインプットファイルの最初に書くようにしてください。 統計誤差が stdcut 以下になったタリーから計算を打ち切ります。
istdbat	(D=0)	計算初期に stdcut を無効にするパラメータ。バッチ番号が istdbat 以下の場合，stdcut による計算打ち切りを行いません。OpenMP を利用した場合，バッチ分散から統計誤差を推定するため (istdev = 1)，計算初期に統計誤差が不安定になります。そのような場合は，誤差が安定してくるバッチ数を istdbat に指定してください。

時間のカットオフは粒子毎に tmax(i)= で指定します。単位は nsec です。カットオフ時間を上回った粒子は輸送を終了し、殺されます。高エネルギーの場合ほとんど意味がありませんが、低エネルギーの輸送で利用できます。

ウエイトカットオフは、インポータンス、強制衝突、implicit capture、ウエイトウインドウを利用し、粒子のウエイトが変化する場合に、指定のウエイトカットオフ値になった粒子にロシアンルーレットを施すものです。ウエイトウインドウを指定した粒子については作用しません。

粒子のウエイトが、$WC2$ とソースの発生場所のインポータンスと現在のインポータンスの比 R の積、$WC2\times R$ より小さくなった時、現在の粒子ウエイト WGT の関数、$WGT/(WC1\times R)$ の確率で存続させ、その時粒子のウエイトを $WGT = WC1\times R$ とします。それ以外は、その粒子を殺します。$WC1, WC2$ の入力が負の数で与えられた場合は、$|WC1|\times SWTM$ また、$|WC2|\times SWTM$ が $WC1, WC2$ として設定されます。インポータンスが指定されていない粒子や、領域は、インポータンスは全て 1 にセットされます。

ウエイトウインドウのパラメータについては、通常はデフォルト値で省略できます。

iwwbias=1 とした場合、input echo の [weight window] には [ww bias] で定義した値を考慮した結果

が書き出され、[ww bias] には off が付きます。もし、[ww bias] が定義されていない場合は、input echo において全ての値が 1 となった [ww bias] が書き出されます。

5.2.5 阻止能モデル

表 13: 阻止能モデルに関するパラメータ

パラメータ	値	説明
ndedx	(D=3)	電子・陽電子を除く荷電粒子の dE/dx オプション
	= 0	重イオンは SPAR[31]、他は NMTC オリジナル
	= 1	重イオン、陽子は ATIMA[32]、他は NMTC オリジナル
	= 2	重イオン、陽子、パイオン、ミューオンは SPAR、他は NMTC オリジナル
	= 3	全ての荷電粒子に対して ATIMA を利用する
mdbatima	(D=500)	ATIMA 用データベースの最大値
dbcutoff	(D=0.0)	ATIMA 用データベース作成の下限エネルギー (MeV/n)
ih2o	(D=-1)	ATIMA の時の水 (H2O のみ) の Ionization Potential オプション
	< 0	デフォルト値、75 eV
	> 0	水の Ionization Potential (eV)
ifixchg	(D=0)	ATIMA で阻止能を計算するときの電荷数
	= 0	有効電荷。高エネルギーの場合は完全電離（フルストリップ），エネルギーが低くなると経験式に従って徐々に電荷が下がる。
	= 1	固定値。基本的には完全電離を仮定するが，izst パラメータで線源粒子の電荷を指定した場合，核反応が起きるまではその値となる。実際は物質中ですぐに電子がはぎ取られて完全電離状態となるが，その過程は模擬できないため，薄膜やガス中での計算を想定。
irlet	(D=1)	δ 線生成機能を使ったときの荷電粒子の dE/dx 計算手法。
	= 0	従来手法（実際に生成した δ 線のエネルギーを非制限阻止能から差し引く方法）。
	= 1	制限付阻止能。

ndedx は、バージョン 2.00 以前は ndedx=0 を、それより後からバージョン 2.85 までは ndedx=2 をデフォルトとしておりましたので、注意してください。

バージョン 2.85 より、阻止能計算モデル ATIMA のアルゴリズムを改良し、高速化しました。従来のアルゴリズムでは、輸送計算の途中で新たな粒子が発生したり、別の物質に到達したりする度に、毎回 ATIMA による阻止能計算を実行し、粒子の飛程を求めていました。そこで、計算された粒子・物質の組みに対する情報をデータベースとして記憶しておき、次回の同じ組の呼び出しではデータベースを参照することで、計算負荷の高い ATIMA の呼び出し回数を減らすように変更しました。この改良により、精度の高い ATIMA を用いた PHITS シミュレーションが、SPAR を用いた場合とほぼ同程度の計算時間で実行できるようになりました。

バージョン 2.97 より、Np から Bk($93 \leq Z \leq 97$) の元素を含む物質についても、ndedx=3 を使用できるようになりました[33]。その際、核的阻止能の計算に必要となるイオン化ポテンシャルおよび密度補正のパラメータを文献[34]より追加し、電子的阻止能の計算は SPAR で代用しました。

[31] T.M. Armstrong and K.C. Chandler, ORNL Report, ORNL-4869 (1973).
[32] ATIMA モデルについては、次のウェブサイトをご覧ください。https://web-docs.gsi.de/~weick/atima/
[33] バージョン 2.96 以前では、U より大きな核種を含む物質を扱う場合、ATIMA は使用できません。その際は ndedx=2 としてください。
[34] R.M.Sternheimer, Atomic data and nuclear data tables, 30, 261-271 (1984) の表 1。

5.2.6 弾性散乱や全断面積、反応断面積モデル

表 14: 弾性散乱や全断面積、反応断面積モデルに関するパラメータ

パラメータ	値	説明
ielas	(D=2)	弾性散乱オプション
	= 0	弾性散乱を考慮しない
	= 1	中性子の弾性散乱を考慮する
	= 2	中性子と陽子の弾性散乱を考慮する
ielms	(D=100)	弾性散乱角度分布分点数
inmed	(D=1)	Bertini モデルの核子核子断面積オプション
	= 0	free (nmtclk25.dat)
	= 1	Cugnon old (nmtclk95.dat)
	= 2	Cugnon new (nmtclk30.dat)
icxsni	(D=0)	核子・原子核間の反応断面積、弾性散乱、全断面積オプション
	= 0	Pearlstein-Niita の式
	= 1	KUROTAMA モデル
	= 2	Sato の式
icrhi	(D=2)	原子核・原子核反応の反応断面積オプション
	= 0	Shen の式
	= 1	NASA の式
	= 2	KUROTAMA モデル
icrdm	(D=0)	重陽子の反応断面積オプション
	= 0	icrhi と同じ
	= 1	MWO の式
icxspi	(D=1)	パイオンの反応断面積オプション
	= 0	幾何学的な断面積
	= 1	Hashimoto の式

inmed オプションは、inmed=1 の inmedium 断面積がデフォルトになっています。

KUROTAMA モデルは、幅広いエネルギー範囲において様々な原子核間同士の反応断面積を再現する最新のモデルです。詳細はこの文献[35] をご覧ください。また、本モデルを使用して得られた結果を使用して論文発表等を行う場合は、必ず該当文献をご引用ください。

オリジナルの NASA の式[36] は、軽イオンに対して低エネルギー補正項 X_m が導入されていますが、PHITS では $X_m = 1$ として計算しています。

MWO の式は、重陽子の全反応断面積を精度良く記述するモデルです。主に、入射エネルギーが 1 GeV 以下の領域および標的核が炭素以上の大きさをもつ場合に、高い信頼性をもちます。詳細は文献[37] をご覧ください。また、本モデルを使用して得られた結果を使用して論文発表等を行う場合は、必ず該当文献をご引用ください。

バージョン 2.86 より、Hashimoto の式がパイオンの全反応断面積のモデルとしてデフォルトで使用されます。この式は、従来の幾何学的な断面積と比べて、入射エネルギー依存性を考慮しており、デルタ共鳴によるピーク構造を再現します。

[35] 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).
[36] R.K. Tripathi, F.A. Cucinotta, and J.W. Wilson, Nuclear Instruments and Methods in Physics Research B 155 (1999) 349-356
[37] K. Minomo, K. Washiyama, and K. Ogata, J. Nucl. Sci. Technol. 54, 127-130 (2017).

5.2.7 崩壊や脱励起過程モデル

表 15: 崩壊や脱励起過程モデルに関するパラメータ

パラメータ	値	説明
nevap	(D=3)	蒸発モデルのオプション
	= 0	蒸発モデルを用いない
	= 3	GEM モデルを用いる
ngem	(D=1)	GEM モデルのオプション。
	= 0	デフォルトバージョン (Ver.1) を使う。
	= 1	Ver.1 を使う。
	= 2	Ver.2 を使う（核子放出・ガンマ線放出の競合、殻補正に KTUY のデータ[38]を使う）。
ifission	(D=1)	核分裂反応モデルのオプション。
	= 0	デフォルトバージョン (Ver.1) を使う。
	= 1	Ver.1 を使う。
	= 2	統計崩壊開始時に原子番号が 89 未満の核は、Ver.1 に代わって Iwamoto モデル[39]を使用する。
igamma	(D=2)	残留核の γ 崩壊オプション。
	= 0	γ 崩壊を考慮しない。
	= 1	γ 崩壊を考慮する。
	= 2	γ 崩壊を EBITEM モデルを用いて考慮する。
	= 3	γ 崩壊とアイソマー生成を EBITEM モデルを用いて考慮する。 バージョン 2.73 以前では、`1-3` の場合に `file(14)=trxcrd.dat` を [parameters] セクションで設定する必要があります。
	=-1,-2,-3	1,2,3 と同様ですが励起核の運動に起因するドップラー効果を無視します。
ismm	(D=0)	統計マルチフラグメンテーションモデル (SMM) のオプション。
	= 0	統計マルチフラグメンテーションモデルを用いない。
	= 1	統計マルチフラグメンテーションモデルを用いる。JQMD の場合、JQMD から GEM への切り替え時間がデフォルトの 150 fm/c から 75 fm/c に変わります。

EBITEM モデルを用いてアイソマー生成を考慮すると、`[t-yield]` において `axis=chart, dchain` とした場合に、生成されたアイソマーの情報が出力されます。本モデルに関する詳細は文献[40] をご覧ください。

`igamma<0` は反応の物理過程を歪めるため、必要な場合（ガンマ線放出の準位構造を調べるなど）以外は使わないでください。

統計マルチフラグメンテーションモデルは、鉛や水銀などの重い核、または 100 MeV/n 前後での核反応で核種生成の再現性を向上させます。ただし核反応の計算にかかる時間が場合によっては数倍長くなります。本モデルに関する詳細は文献[41] [42] をご覧ください。

---

[38] 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.
[39] Hiroki Iwamoto, and Shin-ichiro Meigo, Journal of Nuclear Science and Technology, 56 (2019) 160-171.
[40] T. Ogawa, S. Hashimoto, T. Sato, and K. Niita, Nuclear Instruments and Methods in Physics Research B 325 (2014) 35-42.
[41] T. Ogawa, T. Sato, S. Hashimoto, and K. Niita, Nuclear Instruments and Methods in Physics Research A 723 (2013) 36-46.
[42] J.P. Bondorf, A.S. Botvina, A.S. Iljinov, I.N. Mishustin, and K. Sneppen, Physics Reports 257 (1995) 133-221.

5.2.8 イベントジェネレーターとカーマ近似

表 16: イベントジェネレーターとカーマ近似に関するパラメータ

パラメータ	値	説明
e-mode	(D=0)	Event generator mode に関するオプション。20 MeV 以下の中性子のみに影響。
	= 0	単純サンプリングモード (イベントジェネレータモ—ド無効)。
	= 1	Event generator mode Ver. 1. (単純サンプリング + 蒸発モデル).
	= 2	Event generator mode Ver. 2 (Ver.1 ＋荷電粒子生成量高精度化＋中性子エネルギー分布の高精度化). バージョン 2.73 以前では、1,2 の場合に file(14)=trxcrd.dat を [parameters] セクションで設定する必要があります。
em-emode	(D=20)	e-mode の上限エネルギー [MeV]。核データを用いた e-mode の計算のとき、このモードを適用するエネルギーの上限。
ikerman	(D=0)	[t-deposit] で付与エネルギーを計算する際、中性子に対してカーマ近似の使用の有無を決めるオプション。
	=0	自動調整。イベントジェネレータモードを使用すると (e-mode≥1)、中性子反応により放出される荷電粒子を明示的に計算するため、カーマ近似は利用しなくなります。
	=1	常にカーマ近似を利用しない。
ikermap	(D=0)	[t-deposit] で付与エネルギーを計算する際、光子に対してカーマ近似の使用の有無を決めるオプション。
	=0	自動調整。EGS5 モードを使用すると (negs=1)、2 次電子を明示的に計算するため、カーマ近似は利用しなくなります。また、EGS5 モードを利用しない場合でも、電子のカットオフエネルギー emin(12) 及び emin(13) を 10 MeV 以下に設定すると利用しなくなります。
	=1	常にカーマ近似を利用しない（電子による付与エネルギーを計算する）。

イベントジェネレータモードは、20 MeV 以下の中性子による検出器波高、残留核の反跳、微細線量など、イベントごとの量を計算する場合必要な機能です。単純サンプリングモード (e-mode=0) は核データの二次中性子エネルギー分布を正確に再現しますが、 エネルギー・運動量保存則は満たしません。e-mode オプションの詳細は 5.2.26 をご覧ください。

5.2.9 光子輸送の adjoint mode

表 17: Adjoint mode のパラメータ

パラメータ	値	説明
iadjoint	(D=0)	Adjoint mode の輸送オプション。
	= 0	Adjoint mode を用いない（通常の輸送計算）。
	= 1	Adjoint mode を用いる。

iadjoint=1 を設定することで、光子を adjoint mode で輸送します。Adjoint mode の計算は、通常の輸送計算 (forward mode) より、線源領域がタリー領域より大きい場合に効率的です。Adjoint mode では、forward mode の線源とタリー領域を切り替えた計算をしており、タリー領域で発生させた粒子を時間的に逆に解き、線源領域まで輸送させます。

PHITS の adjoint mode では光子の輸送のみ模擬できます（他の粒子の輸送は不可）。この mode の光子は物質と逆反応し、その反応は 3.0 MeV 以下のエネルギー領域で連続的にモデル化されています。詳細は下記の文献[43]を参照してください。

Adjoint mode（iadjoint=1）を使用する場合は、次のように各パラメータを定義する必要があります:

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

Adjoint mode 計算において、[Source] で定義する線源領域は、通常の輸送計算におけるタリー領域と一致する必要があります。また、[Source] では proj=photon を定義し、均一的なソースエネルギー分布を定義してください。エネルギー分布の最小値と最大値はタリーで定義するエネルギー範囲をカバーするようにしてください。

[T-Adjoint] タリーの利用で adjoint mode 計算の結果が取得できます。[T-Adjoint] の結果は、通常の輸送計算における [T-Track] の結果に相当します。[T-Adjoint] のタリー領域は通常の輸送計算での [Source] の領域と一致する必要があります。

Adjoint mode では、[T-Adjoint] での e-type メッシュの最小と最大のエネルギー値は [Source] セクションで定義するエネルギー分布の最小値と最大値と一致するようにしてください。[T-Adjoint] の e-smin と e-smax パラメータで実際の光子線源の最小と最大エネルギーを定義することになります。

Adjoint mode の実行方法や例題については、\phits\utility\adjoint\ フォルダにあるサンプルインプットをご参照ください。

[43] A. Malins *et al.*, “Continuous energy adjoint transport for photons in PHITS ” EPJ Web Conf., 153, 06001 (2017). DOI: 10.1051/epj-conf/201715306001

5.2.10 核反応モデルオプション

表 18: 核反応モデルオプションに関するパラメータ

パラメータ	値	説明
andit	(D=0)	Bertini モデルの Δ 角分布オプション
	= 0	50%等方、50%前方
	= 1	全て等方分布
	= 2	全て前方分布
iidfs	(D=0)	中性子入射核分裂反応のオプション
	= 0	中性子の核データを使用する
	= 1	文献値に基づき中性子多重度 ν とエネルギー分布を再現する
idwba	(D=0)	DWBA スペクトルのオプション
	= 0	DWBA 計算で求めた離散スペクトルを考慮しない
	= 1	DWBA 計算で求めた離散スペクトルを考慮する
npidk	(D=0)	計算打切エネルギーに達した負パイオンの崩壊の取り扱い
	= 0	吸収反応を強制的に考慮する
	= 1	崩壊させる

iidfs=1 を指定した場合、下に示す 18 核種の核分裂で放出される中性子の多重度 ν とエネルギー分布を文献[44] にあるデータに基づき再現します。
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.

idwba は、特定の標的原子核における陽子、重陽子入射反応を対象として、放出中性子や陽子のエネルギースペクトルに、DWBA（歪曲波ボルン近似）によって計算した離散スペクトルを追加するオプションです。具体的には、以下の入射エネルギーと反応が対象となります。

30–400 MeV における ^{7}Li$(p,n)^{7}$Be 反応
10–50 MeV における ^{9}Be$(p,n)^{9}$B 反応
10–50 MeV における ^{6,7}Li$(d,n)^{7,8}$Be および ^{6,7}Li$(d,p)^{7,8}$Li 反応
5–25 MeV における ^{9}Be$(d,n)^{10}$B および ^{9}Be$(d,p)^{10}$Be 反応
10–50 MeV における ^{12,13}C$(d,n)^{13,14}$N および ^{12,13}C$(d,p)^{13,14}$C 反応

崩壊チャンネルを持つ粒子が、計算打切エネルギーに達した時には、全て崩壊させます。その中で負パイオンについては、npidk=0 としていると、まず強制的に核反応を試み吸収させます。それでも吸収されなかった場合は、やはり崩壊させます。

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

5.2.11 低エネルギー中性子に関するオプション

以下のパラメータは、20 MeV 以下の中性子の計算に用います。

表 19: 低エネルギー中性子に関するパラメータ

パラメータ	値	説明
emcnf	(D=0.0)	中性子捕獲のしきい値 (MeV)
		これ以上で implicit capture、以下で analog capture
dnb	(D=-1)	核分裂からの delayed neutron 数
	=-1	natural sampling
	= 0	no delayed neutron
	> 0	neutron number
nonu	(D=1)	通常の fission として取り扱う
	=0	fission を起こさない。単なる中性子捕獲反応として扱う。
		光子が生成される。
isaba	(D=0)	$S(\alpha,\beta)$ に関するオプション。
	= 0	このオプションを使用しない。
	= 1	$S(\alpha,\beta)$ を使用した計算においてデータの内挿を行う。

nonu=0 とすることで、計算体系内に設定された核分裂性核種に、輸送中の中性子が核分裂反応を誘起した際、2 次中性子による中性子増倍効果を発現させず、即ち、中性子を発生させずに、単に中性子捕獲反応として輸送計算を継続することができます。用途としては、臨界解析コード等により別途評価した同体系における中性子実効増倍率を加味した中性子線源等を用いて、実質的な中性子透過問題の評価が行えます。これは、MCNP コードにおける NONU カード相当に相当します。

isaba=1 は、薄膜等の散乱回数が少ない場合を計算する際に指定してください。

5.2.12 光子・電子輸送の PHITS オリジナルモデル

以下のパラメータは、negs≠ 1 のときのみ有効となります。

表 20: 光子・電子輸送オリジナルモデルに関するパラメータ

パラメータ	値	説明
emcpf	(D=100)	photon の詳細モデルの上限 (MeV)
nocoh	(D=0)	photon の coherent scattering のオプション
	= 0	coherent scattering を起こす
	= 1	coherent scattering を起こさない
ibad	(D=0)	brems. の角分布のオプション
	= 0	full brems. tabular angular distribution
	= 1	simple brems. angular distribution approximation
bnum	(D=1)	brems. photon のオプション
	= 0	brems. photon を生成しない
	> 0	analog brems. photon 数
xnum	(D=1)	x-ray photon のオプション
	= 0	x-ray photon を生成しない
	> 0	analog x-ray photon 数
numb	(D=0)	brems. process のオプション
	= 0	nominal brems. production
	> 0	produce brems. on each substep

5.2.13 EGS5 用パラメータ

表 21: EGS5 用パラメータ 1

パラメータ	値	説明
ipegs	(D=0)	PEGS5（EGS5 用断面積生成プログラム）コントロールオプション。 pegs5.inp をユーザー自身で変更したいときに使用します。なお、ipegs が 1 以上の場合、iegsout は自動的に 1 となり PEGS5 関連ファイルを残すようにします。（ただし、iegsout = 2 としている場合は変更しません。） また、negs≠ 1 のときは無効となります。
	= -1	PEGS5.inp の作成○、PEGS5 の実行○、PHITS の実行×
	= 0	PEGS5.inp の作成○、PEGS5 の実行○、PHITS の実行○
	= 1	PEGS5.inp の作成×、PEGS5 の実行○、PHITS の実行○
	= 2	PEGS5.inp の作成×、PEGS5 の実行×、PHITS の実行○
imsegs	(D=1)	多重散乱（Multiple Scattering）の扱いに関するオプション。
	= 0	EGS5 オリジナル手法。ソース発生直後のみ starting scattering strength を使って多重散乱を詳細に模擬し、その後は、maximum scattering strength を使用する。
	= 1	PHITS-EGS5 独自の手法。物質が変わるたびに starting scattering strength を使用して多重散乱を詳細に模擬する。これにより、多少計算時間が長くなるが、chard パラメータを変更することなく薄膜による散乱をある程度正確に模擬できるようになる。
iegsout	(D=0)	EGS5 用出力ファイルオプション
	= 0	EGS5 用の出力ファイルを残さない。ただし、計算を途中で打ち切った場合は、ファイルが残る場合があります。
	= 1	再読込するファイル（pegs5.inp, pegs5.dat, pegs5.msfit）のみ残す
	= 2	全て残す。各ファイルに出力される情報の詳細は、EGS5 のマニュアル[45]をご参照ください。

PEGS は EGS5 を使用する前に動作させる Fortran のサブプログラムです。EGS5 の計算で必要となる物質毎の断面積データを生成します。

negs=1 の場合、電子･陽電子･光子の輸送エネルギー下限･上限に対するデフォルト値が emin(12,13)=0.1, emin(14)=0.001, dmax(12-14)=1000.0 となります。ただし、これらの値を [parameters] セクションで直接指定した場合は、指定した値が優先されます。また、多くの EGS5 用パラメータのデフォルト値がオリジナル EGS5 の値と異なっていることにご注意ください。

バージョン 2.76 より EGS5 を使用している場合でも光核反応を考慮することができるようになりました。ipnint=1 とした場合は、以前のバージョンと結果が変わりますのでご注意ください。

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

表 22: EGS5 用パラメータ 2

パラメータ	値	説明
iegsrand	(D=-1)	EGS5 で使う乱数初期値。0 以上を指定した場合、EGS5 用の乱数を使うため、MPI 並列計算や再開始計算ができなくなることにご注意ください。
	< 0	EGS5 用の乱数は使わない（PHITS で使う乱数をそのまま利用する）
	= 0	EGS5 のデフォルト乱数初期値（314159265）を使う
	> 0	指定した値を EGS5 乱数初期値として使う
iedgfl	(D=1)	K 及び L 殻特性 X 線に関するオプション
	= 0	K 及び L 殻特性 X 線を扱わない
	= 1	K 及び L 殻特性 X 線を扱う
iauger	(D=1)	K 及び L 殻オージェ電子に関するオプション
	= 0	K 及び L 殻オージェ電子を扱わない
	= 1	K 及び L 殻オージェ電子を扱う
iraylr	(D=1)	レイリー散乱（Rayleigh scattering）に関するオプション
	= 0	レイリー散乱を扱わない
	= 1	レイリー散乱を扱う
lpolar	(D=0)	光子散乱での直線偏光（linearly polarized photon scattering）に関するオプション（未対応）
	= 0	直線偏光を扱わない
	= 1	直線偏光を扱う
iunrst	(D=0)	PEGS5 による阻止能出力オプション。ただし、計算結果が変わってしまうので、0 以外の値を指定する場合は ipegs=-1 として PEGS5 のみ実行すること。
	= 1	unrestricted collision only
	= 2	unrestricted collision and radiative
	= 3	unrestricted collision, restricted radiative
	= 4	restricted collision, unrestricted radiative
	= 5	unrestricted radiative only
	= 6	restricted radiative only
	= 7	restricted collision only
chard	(D=0.1)	EGS5 のステップサイズをコントロールするパラメータ。imsegs=0 として薄膜のある体系を計算する場合、その薄膜の厚さ程度に設定する必要があります。オリジナルの EGS5 では物質毎に設定できますが、PHITS では全ての物質に対して同じ値を設定します。 また、負値を入力した場合は、その絶対値を各物質の密度 (g/cm^3) で割った値を EGS5 で使用する chard パラメータに設定します。密度が薄く巨大な体系（km オーダー）では、負値を設定することにより、計算時間を大幅に短縮できます。

表 23: EGS5 用パラメータ 3

パラメータ	値	説明
epstfl	(D=0)	ICRU90 に格納された衝突断面積に関する密度補正係数を利用するオプション。
	= 0	ICRU90 を使わない。
	= 1	ICRU90 を使う。この場合、[material] セクションで定義した物質から、液体の水、グラファイト、空気を自動的に判別して、それらの物質に対して ICRU90 に対応した密度補正係数を使います。自動判別する基準は以下の通りです。 液体の水：H 及び O で構成され密度が 0.9–1.1 g/cm^3 の物質 C で構成され密度が 1.55–2.40 g/cm^3 の物質 C, N, O, 及び Ar で構成され密度が 0.03 g/cm^3 の物質。このしきい値密度は gasegs パラメータで調整可能です。
gasegs	(D=0.03)	この密度以下の物質は、PEGS で自動的に気体とみなされます。
incohr	(D=1)	コンプトン散乱角の非干渉散乱関数（incoherent scattering function、S/Z rejection）に関するオプション
	= 0	非干渉散乱関数を使わない（自由電子との散乱）
	= 1	非干渉散乱関数を使う（束縛電子との散乱）（ibound が自動的に 1 に固定されます）
iprofr	(D=1)	コンプトン散乱におけるドップラー効果に関するオプション
	= 0	ドップラー効果を考慮しない
	= 1	ドップラー効果を考慮する（incohr と ibound が自動的に 1 に固定されます）
impacr	(D=1)	EII（Electron impact ionization）に関するオプション
	= 0	EII を考慮しない
	= 1	EII を考慮する
ieispl	(D=0)	EII により発生する光子の扱いに関するオプション
	= 0	分割しない（no splitting）
	= 1	分割する（splitting）
neispl	(D=0)	ieispl=1 のときに分割する光子の数
ibrdst	(D=1)	制動放射の発生角度に関するオプション
	= 0	固定値
	= 1	サンプリング
iprdst	(D=1)	電子対生成の発生角度に関するオプション
	= 0	固定値
	= 1	サンプリング
iphter	(D=1)	光電子の角度分布に関するオプション
	= 0	固定値（入射光子の方向）
	= 1	サンプリング
ibound	(D=1)	コンプトン断面積に関するオプション
	= 0	自由コンプトン断面積 (free Compton cross section)
	= 1	束縛全コンプトン断面積 (bound total Compton cross section)
iaprim	(D=1)	制動放射断面積に関するオプション
	= 0	Motz *et al.* の経験式を使う
	= 1	ICRU-37 で与えられた放射阻止能（radiative stopping power) に規格化する
	= 2	規格化しない

5.2.14 光・ミューオン反応モデル

表 24: 光・ミューオン反応モデルに関するパラメータ

パラメータ	値	説明
ipnint	(D=0)	光核反応のオプション。 (バージョン 2.30 以前、ipngdr というパラメータ名であったもの。)
	= 0	光核反応を考慮しない。
	= 1	光核反応を考慮する。ただし核共鳴蛍光散乱を除く。
	= 2	核共鳴蛍光散乱を含む全ての光核反応を考慮する。 （偏光については欄外参照。）
pnimul	(D=1.0)	光核反応のサンプリング確率に乗じる倍率。
igmuppd	(D=0)	光子によるミューオン対の生成に関するオプション。
	= 0	考慮しない。
	= 1	考慮する。
gmumul	(D=1.0)	光子によるミューオン対生成のサンプリング確率に乗じる倍率。
imucap	(D=1)	負ミューオンが止まったときのミューオン捕獲反応に関するオプション。ミューオニックアトム生成後の特性 X 線の放出と核吸収反応を考慮する。
	= 0	考慮しない。
	= 1	考慮する。特性 X 線の放出の計算で用いるパラメータは、デフォルト値を使用する。
	= 2	考慮する。特性 X 線の放出の計算で用いるパラメータは、file(28) で指定したユーザー定義ファイルから読み込む。
imuint	(D=1)	仮想光子を介して高エネルギーミューオンが引き起こす核反応に関するオプション。
	= 0	考慮しない。
	= 1	考慮する。
imubrm	(D=1)	ミューオンによる制動放射に関するオプション。
	= 0	考慮しない。
	= 1	考慮する。
imuppd	(D=1)	ミューオンによる電子・陽電子対生成に関するオプション。
	= 0	考慮しない。
	= 1	考慮する。
emumin	(D=200.0)	仮想光子を介して核反応を引き起こすミューオンの最低エネルギー (MeV)。imuint = 1 の場合のみ有効。
emumax	(D=1.0e+6)	仮想光子を介して核反応を引き起こすミューオンの最大エネルギー (MeV)。imuint = 1 の場合のみ有効。

ipnint=2 とした際に考慮される核共鳴蛍光散乱については、入射光子が偏極している場合、[source] セクションで sx,sy,sz を用いて偏光方向を指定できます。その偏極方向を考慮して、散乱光子の角度分布が決定されます。

emumin を変更した場合、ミューオン核反応断面積が全エネルギー領域で変化しますのでご注意ください。

5.2.15　ニュートリノ反応モデル

表 25: ニュートリノ反応モデルに関するパラメータ

パラメータ	値	説明
ntrnore	(D=0)	ニュートリノ反応のオプション。
	= 0	ニュートリノの反応を考慮しない。
	= 1	ニュートリノの反応を考慮する。

ntrnore = 1 によって考慮されるのは、$^1\mathrm{H}(\overline{\nu}_e, e^+)n$ 反応、$^2\mathrm{H}(\overline{\nu}_e, \overline{\nu}'_e)np$ 反応、$^2\mathrm{H}(\nu_e, e^-)2p$ 反応、$^2\mathrm{H}(\nu_e, \nu'_e)np$、電子との弾性散乱反応です。入射エネルギーが 20 MeV 以上では一部の反応で断面積の外挿を行います。ただし、入射エネルギーが 150 MeV 以上の反応チャンネル（例 : μ ニュートリノの荷電カレント反応）は、考慮されませんのでご注意ください。表 26 に、考慮しているニュートリノ反応の有無を標的粒子毎にまとめました。考慮している場合は、その上限エネルギーを示しています。

表 26: 各標的粒子に対するニュートリノ反応の上限エネルギー。これより高いエネルギーでは外挿した値を使用します。× は未実装、-は反応が存在しないことを表します。

荷電カレント反応

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

中性カレント反応

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

5.2.16 クーロン散乱や重力場・電磁場に関するオプション

表 27: クーロン散乱や重力場・電磁場に関するパラメータ

パラメータ	値	説明
`nspred`	(D=0)	電子・陽電子を除く荷電粒子のクーロン散乱 (angle straggling) オプション[46]。
	= 0	クーロン散乱を考慮しない
	= 1	クーロン散乱を考慮する。NMTC オリジナル
	= 2	（推奨）クーロン散乱を考慮する。Moliere 理論に基づく Lynch の式[47]
`ascat1`	(D=13.6)	Lynch の式（下記参照）の係数 A。`nspred = 2` の場合のみ有効。
`ascat2`	(D=0.038)	Lynch の式（下記参照）の係数 B。`nspred = 2` の場合のみ有効。
`nedisp`	(D=0)	電子・陽電子を除く荷電粒子の減速過程におけるエネルギー分散 (energy straggling) のオプション。
	= 0	エネルギー分散を考慮しない。
	= 1	（推奨）エネルギー分散を考慮する。Landau Vavilov[48]
`gravx`	(D=0)	重力場の方向ベクトルの x 成分
`gravy`	(D=0)	重力場の方向ベクトルの y 成分
`gravz`	(D=0)	重力場の方向ベクトルの z 成分
`usrmgt`	(D=1)	[magnetic field] で time を指定した時のユーザーサブルーティンオプション
	= 1	usrmgt1.f を使用、現在、Wobbler magnet がインストール
	= 2	usrmgt2.f を使用、現在、中性子用パルスマグネットがインストール
`usrelst`	(D=1)	[elastic option] でのユーザーサブルーティンオプション
	= 1	usrelst1.f を使用、現在、中性子用 Bragg 散乱用プログラム
	= 2	usrelst2.f を使用、現在、中性子用サンプルプログラム
`imagnf`	(D=0)	磁場のオプション
	= 0	磁場を考慮しない
	= 1	磁場を考慮する。タリーで粒子軌跡を滑らかに表示したい場合は、`deltm` を軌跡を表示するステップ長に変更してください。
`ielctf`	(D=0)	電磁混合場のオプション
	= 0	電磁混合場を考慮しない
	= 1	電磁混合場を考慮する。タリーで粒子軌跡を滑らかに表示したい場合は、`deltm` を軌跡を表示するステップ長に変更してください。

`gravx`, `gravy`, `gravz` は、重力場の方向ベクトル。重力を設定したときは、1 eV 以下の中性子に作用します。`gravx=1`, `gravy=0`, `gravz=0` を指定したときは、x 負方向に重力が作用します。

[46] 厚さが 1cm 以下の薄膜等による散乱計算を行う場合は、`delt0` をその厚さの 1/10 に設定してください。

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

[48] 本モデルについては、次のウェブサイトをご覧ください。`http://www.dnp.fmph.uniba.sk/cernlib/asdoc/geantold/H2GEANTPHYS332.html`

5.2.17 出力オプション (1)

表 28: 出力オプションに関するパラメータ 1

パラメータ	値	説明
infout	(D=7)	file(6) に出力する情報を指定するオプション 次の I, II, III, IV, V, VI については表外の説明を参照
	= 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
nrecover	(D=0)	Lost particle が発生して recover に成功した場合に、 warning message を出力する回数。
ierrout	(D=0)	エラーや警告が表示された際、そのメッセージを出力したソースコードのプログラム名、サブルーチン名、行番号を出力するオプション。
	= 0	上記情報を出力しない。
	= 1	上記情報を出力する。例えば、`---L:22332/R:param/F:read02.f` と出力された場合、`read02.f` の 22332 行目、サブルーチン（もしくはファンクション）param でメッセージが出力されたことを意味します。

パラメーター `infout` は、PHITS のサマリーファイルに出力する情報を指定するオプションです。情報は次の 6 カテゴリに分類されており、`infout` によって出力させるカテゴリを指定できます。

I 基本情報
ロゴ（PHITS 著者の情報を除く）、計算プロセス (ncol) 情報、発生線源情報、幾何形状エラー情報、乱数情報、計算時間情報

II インプットエコー

III メモリー情報、バッチ情報

IV 粒子情報

V 詳細情報（分散低減情報、領域別散乱情報、物質別散乱情報）

VI PHITS 著者の情報

これらの情報は、下記の順番で書き出されます。

(1) ロゴ（カテゴリ I. ただし、著者情報はカテゴリ VI）
PHITS のバージョン、PHITS 著者、JOB タイトル、スタート時間

(2) インプットエコー（カテゴリ II）
ユーザーのインプットファイルのエコー

(3) メモリー情報（カテゴリ III）
どのパートにどのくらいのメモリーが使われているかの情報

(4) バッチ情報（カテゴリ III）
バッチ毎のヒストリー数、計算時間等の情報

(5) 計算プロセス (ncol) 情報（カテゴリ I）
領域横断、散乱、カットオフ等、輸送のプロセス別の回数

(6) 分散低減情報（カテゴリ V）
ウエイトを変化させる importance, weight window, forced collision 等における情報

(7) 領域別散乱情報（カテゴリ V）
領域ごとの散乱の情報

(8) 物質別散乱情報（カテゴリ V）
物質ごとの散乱の情報

(9) 粒子情報（カテゴリ IV）
輸送粒子、生成粒子、崩壊粒子、カットオフ粒子

(10) 発生線源情報（カテゴリ I）
発生粒子数とそのウエイト

(11) 幾何形状エラー情報（カテゴリ I）
ロスト粒子の数と情報

(12) 乱数情報（カテゴリ I）
初期乱数と次の計算の初期乱数

(13) 計算時間情報（カテゴリ I）
総計算時間と各プロセスのコール回数に関する情報
各項目の意味は以下の通りです。（ただし、`icput=0` の場合は total cpu time 以外の計算時間が出力されません。またその場合、transport と set data の項目自体出力されません。）

- total cpu time: 総計算時間
- transport: 総計算時間から以下の 5 項目に要した時間を除いた時間
- set data: 入力パラメータなどのデータ設定に要した時間
- analysis: 何らかのデータ処理に要した時間および実行回数
- nevap: 蒸発モデルによる蒸発過程の計算に要した時間および実行回数
- dexgam: 原子核の脱励起の計算に要した時間および実行回数
- nreac: 原子・原子核反応の直接過程・前平衡過程計算およびライブラリーによる計算に要した時間および実行回数
- dklos: 粒子の崩壊の計算回数
- hydro: 標的が 1H, 入射粒子が原子核で無い場合の計算回数
- n-data: 中性子ライブラリーによる計算回数
- h-data: 陽子ライブラリーによる計算回数
- p-data: 光子ライブラリーによる計算回数

- e-data: 電子ライブラリーによる計算回数
- p-egs5: EGS5 を用いた光子の計算回数
- e-egs5: EGS5 を用いた電子の計算回数
- photonucl: 光核反応の計算回数
- muon: ミューオン核反応の計算回数
- elast: 弾性散乱の計算回数
- ncasc: 核反応モデルによる直接過程・前平衡過程の計算回数
- bertini: Bertini モデルの計算回数
- isobar: isobar モデルの計算回数
- JAM: JAM モデルの計算回数
- QMD: JQMD モデル、JQMD-2.0 の計算回数
- JAMQMD: JAMQMD モデルの計算回数
- INCL: INCL モデルの計算回数
- INCELF: INC-ELF モデルの計算回数

また、`infout=8` とした場合は核反応モデル毎のサマリーが出力され、各入射粒子の反応が何回モデル計算されたか (count) とその計算が成功した回数 (real) 及びその成功率 (%) が書きだされます。

5.2.18 出力オプション (2)

表 29: 出力オプションに関するパラメータ 2

パラメータ	値	説明
incut	(D=0)	計算打切エネルギー以下の中性子情報の出力オプション
	= 0	出力なし
	= 1	ファイル名 file(12) の ncut ファイルに出力
	= 2	file(12) に時間の情報も出力
igcut	(D=0)	計算打切エネルギー以下の γ 線、電子情報の出力オプション
	= 0	出力なし
	= 1	γ 線情報をファイル名 file(13) の gcut ファイルに出力
	= 2	file(13) に γ 線の時間の情報も出力
	= 3	file(13) に γ 線、電子、陽電子の情報を出力
ipcut	(D=0)	計算打切エネルギー以下の陽子情報の出力オプション
	= 0	出力なし
	= 1	ファイル名 file(10) の pcut ファイルに出力
	= 2	file(10) に時間の情報も出力
inpara	(D=0)	メモリ分散型並列計算の時、ncut ファイル名のオプション
	= 0	/wk/uname/ファイル名 file(12)
	= 1	/wk/uname/ファイル名 file(12) の後に PE 番号を付加
	= 3	ファイル名 file(12) の後に PE 番号を付加
igpara	(D=0)	メモリ分散型並列計算の時、gcut ファイル名のオプション
	= 0	/wk/uname/ファイル名 file(13)
	= 1	/wk/uname/ファイル名 file(13) の後に PE 番号を付加
	= 3	ファイル名 file(13) の後に PE 番号を付加
ippara	(D=0)	メモリ分散型並列計算の時、pcut ファイル名のオプション
	= 0	/wk/uname/ファイル名 file(10) そのまま
	= 1	/wk/uname/ファイル名 file(10) の後に PE 番号を付加
	= 3	ファイル名 file(10) の後に PE 番号を付加
dumpall	(D=0)	全ての情報のダンプオプション
	= 0	出力なし
	= 1	ファイル名 file(15) の dumpall ファイルにバイナリ形式でデータを出力
	= -1	ファイル名 file(15) の dumpall ファイルにアスキー形式でデータを出力
idpara	(D=3)	メモリ分散型並列計算の時、dumpall ファイル名のオプション
	= 0	/wk/uname/ファイル名 file(15)
	= 1	/wk/uname/ファイル名 file(15) の後に PE 番号を付加
	= 3	ファイル名 file(15) の後に PE 番号を付加

incut, igcut のデフォルトをゼロに変えましたので注意してください。

dumpall=1, -1 を指定すると、輸送に関して全ての情報をファイルに書き出します。このファイルを基に、icntl=12 を指定して再計算ができます。最初に計算したときと異なるタリーを入れて再計算したいときなど有効です。ただし、dumpall の機能を利用するときは、巨大なデータファイルが作成されることがありますから注意してください。この dumpall オプションは、GG のときのみ有効です。実際にダンプする情報は、5.2.25 のところで解説します。

“/wk/uname/” にある “/wk/” と “/uname/” はそれぞれデフォルトのディレクトリ名と環境変数 LOGNAME から読み込まれるユーザー名です。メモリ分散型並列計算時は PE (Processor Element) 毎に別のファイルを用

意して書き出します。inpara，igpara，ippara，idpara=0，1 の場合は、各 PE がもっているハードディスク中の “/wk/uname/” のディレクトリにファイルを作成します。inpara，igpara，ippara，idpara =1，3 の場合は、ファイル名の最後に各 PE 番号を付けて総 PE 数 −1 個のファイルを作成します。各 PE は対応するファイルにのみその PE で得られたデータを書き出します。

5.2.19 出力オプション (3)

表 30: 出力オプションに関するパラメータ 3

パラメータ	値	説明
itall	(D=0)	タリーの出力をバッチ毎に行うオプション。
	= -1	出力なし。
	= 0	数値データのみ出力する。
	= 1	数値データおよび画像ファイル（*.eps など）を出力する[49]。
	= 2	各バッチ終了時に数値データを異なるファイルに出力する。ファイル名は、指定のファイル名にバッチ番号を付けたもの。
	= 3	タリー結果とその統計誤差がヒストリー数（或いはバッチ数）によって、どのように推移するかを表示させる。タリーセクションで指定したファイル名の拡張子の前に “_StD” を付けたファイルに出力する。ただし、この機能は [t-track] と [t-point] でのみ動作します。
	= 4	各バッチの結果を独立して集計し、異なるファイルに出力する。ファイル名は、指定のファイル名にバッチ番号を付けたもの。ただし、このオプションはバッチ分散モード（istdev=1）では動作しません。また、この機能は [t-track] と [t-deposit] でのみ動作します。
iMeVperu	(D=0)	タリー出力におけるエネルギーの単位に関するオプション。
	=0	MeV として出力する。
	=1	MeV/n（核子あたりの量）として出力する。ただし、原子核以外については MeV として出力します。
itstep	(D=1)	magnetic field などの粒子の運動量が変化する場でのタリーを取るオプション
	= 0	境界、核反応までタリーを取らない
	= 1	ステップごとにタリーを取る
imout	(D=0)	[material] セクションでの物質表示オプション
	= 0	mat[12]，208Pb.33c タイプ
	= 1	mat[12]，Pb-208.33c タイプ
	= 2	m12，82208.33c MCNP 入力タイプ
jmout	(D=0)	[material] セクションでの物質密度表示オプション
	= 0	入力のまま
	= 1	粒子密度に変換する
	= 2	質量比 (%) に変換する
kmout	(D=0)	核データ物質情報表示オプション
	= 0	表示無し
	= 1	input echo に表示する
		バージョン 2.86 より、核種毎の dmax も出力されるようになりました。

iMeVperu=1とした場合、[t-track], [t-cross], [t-point], [t-product], [t-time], [t-interact]において、原子核のエネルギーの単位を核子あたりの量 [MeV/n] に変換して、各物理量を出力します。例えば、[t-track] において unit=2 とした場合、[1/cm^2/MeV/source] ではなく [1/cm^2/(MeV/n)/source] を単位とする結果が出力されます。

磁場など粒子の運動量を変化させる場がある時、[t-track] で表示する磁場中の軌跡は deltm をステップ幅として表現します。もし、粒子の軌跡を滑らかに表示したい場合は、deltm の値を小さくしてください。

5.2.20 出力オプション (4)

[cell] セクションにおいて同一の物質番号で密度の違う領域を複数定義した場合、デフォルト設定 (matadd=1) では、最初のもの以外の密度の物質に対して新しい物質番号を付与します。matadd=0 とすることで同じ物質番号を指定できるようになりますが、[Mat Name Color] セクションで指定する物質番号には影響がないのでご注意ください。すなわち、このセクションでは自動的に付与された新しい番号を指定する必要があります。この番号は file(6) (D=phits.out) の最初に Warning として出力されますので、その値をご参照ください。

計算時間短縮のため、cpu 時間のカウントをデフォルトで無しにしました。各過程の計算時間を知りたい時は、icput=1 として下さい。

input echo の入力パラメータの表示は、デフォルトで入力のあったものだけですが、ipara=1 とすれば全てのパラメータのデフォルト値がエコーされます。

5.2.21 ボクセル及び連続四面体形状のオプション

ivoxel=2 を指定すると、ボクセルデータをバイナリ形式でファイル名 file(18) に出力して計算を終了します (バージョン 2.30 以前では、終了せずに引き続き計算を行います)。その後、このデータファイルがある状態で ivoxel=1 として再度 PHITS を実行させると、通常一時的に行うバイナリデータの作成を省略して、以後の計算を行います。PHITS では、読み取ったインプットデータを一時的にバイナリ形式で書き出し、それを読み込んでから粒子輸送計算を行います。したがって、読み込むデータ量が大きくなると、それに比例して書き出しの際の計算時間がかかることになります。ivoxel を利用すると、バイナリデータの書き出しを省略するため、何度も PHITS を実行する場合に計算時間が短縮されます。ただし、このパラメータが使えるのは、fill 以降が単純な universe 番号の並びになっている場合のみで、座標変換などが含まれる場合（例えば 3(1 0 1) など）はエラーとなってしまいますのでご注意下さい。

itetvol=1 を指定することで、連続四面体形状 (LAT=3) を使用している場合に、連続四面体形状内に現れる Universe の体積を自動で計算して使用します。ただし、単純に四面体の体積を足し上げるため、Universe と Fill の入れ子構造で、連続四面体形状の一部を切り出して使用している場合などは、正しい体積の値とならないので、注意が必要です。itetvol=1 が指定されている場合は上記の計算値が Volume セクション等での指定よりも優先されます。

itetra=2 を指定すると、連続四面体形状を読み込んで輸送計算に必要な情報データを生成した後に、その情報をバイナリ形式のデータファイル Tetra.bin （ファイル名固定）に出力して計算を終了します。その後、このデータファイルがある状態で itetra=1 として再度 PHITS を実行させると、データ処理を省略して必要情報をファイルから読み込みます。四面体数の多い連続四面体データを使用する際には、データ処理に計算時間がかかるため、同じ連続四面体形状を使用して何度も PHITS を実行する場合には、計算時間の短縮が図れます。

連続四面体形状を使用する計算では、計算スピードを向上させるため、形状を収めた直方体を自動で細分化し、個々の領域に入る四面体の数を減らす二種類のアルゴリズムが組み込まれています。一つは、連

[49] ただし、vtk ファイルに関しては itall=0,1 の両方で出力されます。
[50] 使用している核データライブラリに指定した天然元素がない場合、エラーメッセージを出力します。

表 31: 出力オプションに関するパラメータ 4

パラメータ	値	説明
matadd	(D=1)	[cell] セクションにおいて同一の物質番号を用いて密度が異なる領域を定義した場合のオプション。
	= 0	同じ物質番号で扱う。
	= 1	新しい物質番号を付与する。
natural	(D=1)	[material] セクションにおいて質量数を指定せずに書いた元素に対し、天然存在比による展開を行うかどうかのオプション。核データライブラリの使用時のみ関連し、核反応モデル計算の際はこのパラメータの値に関係なく常に展開します。
	= 0	展開しない[50]。
	= 1	展開する。ただし、file(6) には展開せずに書き出す。
	= 2	展開する。ただし、file(6) に展開した結果を書き出す。
iggcm	(D=0)	GG の warning 等のメッセージをエコーする
	= 0	エコーしない
	= 1	インプットエコー部分に書き出す
icput	(D=0)	CPU 時間のカウントオプション
	= 0	カウント無し
	= 1	カウント有り
ipara	(D=0)	パラメータ表示オプション
	= 0	入力のみの表示
	= 1	全てを表示
nwsors	(D=0)	nwsors 個のソース情報を file(6) に書き出す
gsline	(D=2)	各タリーの gslat を一括で設定するパラメータ。基本的には gsline=gslat となりますが、各タリーで gslat を明示的に指定した場合、そちらのパラメータの方が優先されます。
	= 0	境界線を描画しない。
	= 1	境界線を描画する。（ただしボクセルファントムはきれいに描画されません。）
	= 2	同じセルの場合は境界線を描画しない。
	= 3	同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。 また、gsline=2,3 の場合、gshow≥ 3 としても、セル番号や Lat 番号は表示されなくなります。

続四面体の外側から連続四面体内へと粒子が入射される場合にどの四面体の面を通過するかの判定に使用するアルゴリズムで、領域を細分化して一つの領域に含まれる外部面の数を ntetsurf 以下に限ることで、輸送計算を高速化します。もう一つは、連続四面体内に線源が存在する場合に、放射線が発生する位置がどの四面体に属するかの判断に使用するアルゴリズムで、領域を細分化して、一つの領域に含まれる四面体の数を ntetelem 以下に限ることで、輸送計算を高速化します。

itetauto=1 を指定すると、連続四面体形状の各四面体の要素 universe に対応する領域を自動で作成します。連続四面体形状の構成物質が増えた場合に領域を手作業で作成する手間が省けます。主に NASTRAN バルクデータ形式で連続四面体形状を読み込む場合に便利です。詳しくは 5.6.5.4 をご参照ください。

表 32: ボクセル及び連続四面体形状のオプションに関するパラメータ

パラメータ	値	説明
icells	(D=3)	[cell] セクション全体のバイナリ形式による出力・読込。複雑な体系の場合、一度バイナリ化して再読込することにより計算時間が短縮されます。ただし、一つ一つの cell の定義が単純で cell 数が多い場合は、むしろ計算時間が長くなってしまう場合もありますのでご注意ください。また、icells=1 でバイナリ形式のファイルを読み込む際、[surface] と [cell] セクションの記述を消したりしないようにしてください。他に、同一の物質番号を用いて密度が異なる領域を定義した場合は本機能を使用しないようにしてください。
	= 0	バイナリの入出力なし。[cell] のエコーを出力しない。
	= 1	file(19) からバイナリ形式のジオメトリを読み込む。[cell] のエコーを出力しない。ivoxel は自動的に 1 になる。
	= 2	file(19) にバイナリ形式のジオメトリを出力する。ivoxel は自動的に 2 になる。
	= 3	通常モード。バイナリの入出力なし。[cell] のエコーは出力する。
ivoxel	(D=0)	ボクセルデータのバイナリ形式による出力・読み込み。
	=0	出力なし。
	=1	ファイル名 file(18) からバイナリ形式のボクセルデータを読み込む。
	=2	ファイル名 file(18) にバイナリ形式のボクセルデータを出力する。
	=3	=0 と同じ動作。ただし、ボクセルデータを入力ファイルエコーに出力（これ以外のオプションでは非表示）。
itetvol	(D=0)	連続四面体形状利用時 (LAT=3) における体積の自動計算。
	=0	自動で計算しない。
	=1	自動で計算しタリーなどで体積の値として使用する。
itetra	(D=0)	連続四面体データのバイナリ形式による出力・読み込み。バイナリファイルは ntetsurf や ntetelem に依存しますので，それらのパラメータを変更した場合は，再度，itetra=2 としてバイナリファイルを作り直してください。
	=0	出力なし。
	=1	ファイル Tetra.bin（固定）からバイナリ形式の連続四面体データを読み込む。
	=2	ファイル Tetra.bin（固定）にバイナリ形式の連続四面体データを出力する。
ntetsurf	(D=100)	連続四面体計算のメッシュ分割の際に、一つの領域に許容する連続四面体の外部面の数。
ntetelem	(D=200)	連続四面体計算のメッシュ分割の際に、一つの領域に許容する四面体の数。
itetauto	(D=0)	連続四面体 universe に対応する領域の自動作成。
	=0	自動で作成しない。
	=1	自動で作成する。TetGen 形式では.txt ファイル要。

5.2.22 幾何形状のエラー関係

表 33: 幾何形状のエラー関係のパラメータ

パラメータ	値	説明
nlost	(D=10)	lost particle の許容値 (1PE あたり)
igerr	(D=1)	region error 時の recovery 回数
igchk	(D=0)	=0: 領域横断後の幾何形状チェックを行いません。 =1: 領域横断後の flight mesh を deltb に設定し、 幾何形状のチェックを行います。
ichkmat	(D=0)	[cell] セクションにおいて同一の物質番号を用いて密度が異なる領域を定義した場合に、各 cell の物質番号と密度をチェックするオプション。
	= 0	チェックしない。条件により gshow でジオメトリが正しく描画できない場合があります。ただし、計算結果には影響を与えません。
	= 1	チェックする。gshow でジオメトリが正しく描画できますが、cell 数が多い体系の場合、読込時間が長くなります。
deltb	(D=1.e-9)	領域横断後の flight mesh (cm) igchk=1 で、この距離で幾何形状のチェックを行います。 また、importance, forced collision で追加される粒子の 境界面からの距離。また、source で粒子が境界上にある ときの移動距離としても使います。
deltm	(D=20.12345)	最大飛行メッシュ(cm)。 idelt に連動。
deltc	(D=2.012345)	電子・陽電子を除く荷電粒子に関して、エネルギー分散を考慮した場合 (nedisp=1) の最大飛行メッシュ(cm)。idelt に連動している。
delt0	(D=0.1)	クーロン多重散乱を考慮した場合 (nspred=1,2) の最小飛行メッシュ [cm]。厚さが 1cm 以下の薄膜等による散乱計算を行う場合は、散乱を正確に記述するためにその厚さの 1/10 に設定してください。
deltg	(D=1.012345)	電子・陽電子を除く荷電粒子に関して、磁場における最大飛行メッシュ (cm)。
deltt	(D=1.0)	時間に依存した磁場における最大飛行時間 (msec)。
idelt	(D=1)	最大飛行メッシュに関するオプション。 =0: 最大飛行メッシュとして deltm，deltc を使用。 =1: deltm，deltc を領域の密度で割った値を最大飛行メッシュとする。

igchk=1 とすると、輸送粒子が領域の境界面を横断した後に deltb “flight mesh.” だけ粒子を移動させて幾何形状のチェックを行います。具体的には粒子の運動方向に沿って境界面から deltb のみ移動させます。幾何形状のチェックとはその粒子がどの領域に属しているかを判定するものであり、境界面が曲面となる領域を粒子輸送する際に重要となる機能です。

バージョン 2.80 以降、idelt=1 となっており、deltm，deltc を領域の密度 (g/cm^3) で割った値が最大飛行メッシュとして使われます。これにより、数百 m 以上の空気中の粒子輸送計算を行う場合に、計算時間が短縮されます。

5.2.23 入出力ファイル名

表 34: 入出力ファイル名に関するパラメータ

パラメータ	値	説明
file(1)	(D=c:/phits)	PHITS インストールフォルダ名。このパラメータを指定することにより、フォルダ構造を変えない限り他の入力ファイル名 file(7,20,21,24,25,26) の指定が不要となります。ただし、それらのパラメータを直接指定した場合は、指定値が優先されます。
file(6)	(D=phits.out)	サマリーの出力ファイル名。指定しない時は、標準出力。
file(7)	(D=file(1)/data/xsdir.jnd)	断面積ディレクトリファイル名。
file(10)	(D=fort.10)	計算打切エネルギー以下の陽子情報の出力ファイル名。
file(11)	(D=nuclcal.out)	核反応計算の出力ファイル名。
file(12)	(D=fort.12)	計算打切エネルギー以下の中性子情報の出力ファイル名。
file(13)	(D=fort.13)	計算打切エネルギー以下の γ 線情報の出力ファイル名。
file(15)	(D=dumpall.dat)	dumpall=1 を指定した場合の出力ファイル名。
file(18)	(D=voxel.bin)	ivoxel=1, 2 を指定した際のバイナリデータ用ファイル名。
file(19)	(D=gcell.bin)	icells=1,2 を指定した際のバイナリデータ用ファイル名。指定したファイル名と、そのファイル名に.cfg の拡張子を付けた２つのファイルが生成されます。
file(20)	(D=file(1)/XS/egs/)	EGS5 用断面積データを格納したフォルダ名。negs=1 のときに必要。
file(21)	(D=file(1)/dchain-sp/data/)	DCHAIN 用データを格納したフォルダ名。
file(22)	(D=batch.out)	現在のバッチ情報を出力するファイル名。
file(23)	(D=pegs5)	PEGS5 用出力ファイル名。
file(24)	(D=file(1)/data)	DECDC データ (RIsource.dat) を格納したフォルダ。
file(25)	(D=file(1)/XS/tra)	飛跡構造解析モード用の断面積データを格納したフォルダ名。
file(26)	(D=file(1)/data/multiplier)	付録の [multiplier] セクションデータベースを格納したフォルダ名。
file(27)	(D=file(1)/XS/yield/)	ユーザー指定の放射化断面積を格納したフォルダ名。指定する場合はフォルダ名の最後に “/” を入れてください。
file(28)	(D=file(1)/data/aama.dat)	ユーザー指定の MUONIC ATOM CASCADE program 用入力ファイル名。
file(29)	(D=file(1)/data/dedx)	阻止能データベースファイルを格納したフォルダ名。

file(7) は、他と同様にフルパス名で書いてください。

バージョン 2.74 より、|igamma|≥1 のときでも、γ 崩壊情報データファイル trxcrd.dat を file(14) に指定する必要がなくなりました。

5.2.24 その他

表 35: その他のパラメータ

パラメータ	値	説明
inucr	(D=1)	核反応計算のオプション
	= 1	二重微分断面積計算
	= 2	全断面積、弾性散乱および非弾性散乱断面積の出力
	= 3	シミュレーションによる非弾性散乱の断面積計算
	= 4	弾性散乱の角分布計算
	= 5	pp, np, π-p, 断面積出力
	= 6	シミュレーションによる pp, np, π-p, 断面積計算
idam(i)	integer	ユーザー定義の整数変数
rdam(i)	real*8	ユーザー定義の実数変数
	i = 1 - 100	これらは、各プログラムの中で
		common /userp/ idam(100), rdam(100)
		を引用すれば用いることができる
icommat	(D=0)	[material] セクションにおけるコメント文字のオプション。
	= 0	c をコメント文字として扱わない。
	= 1	c をコメント文字として扱う。

icntl=1 の nuclear reaction 計算は、開発が取り残されているため、現在、まだ工事中と考えてください。これらの inucr の機能が現在も利用可能か定かではありません。

バージョン 2.89 より、デフォルトでは [material] セクションにおいて c をコメント文字として使用できなくなりました。コメント文字として使用したい場合は、icommat=1 としてください。ただし、icommat=1 を含む [parameters] セクションが c を使用している [material] セクションより前に書かれている必要があります。

5.2.25 Dumpall オプション

`dumpall=1, -1` を指定して計算すると、再計算に必要な輸送の情報を全てファイルに書き出します。このファイルを用いて `icntl=12` を指定すると、同じ `dumpall` の値で計算した時にダンプしたデータを基に再計算します。輸送に関しては、最初に計算した全ての情報を再現します。再計算の時には、最初に計算したインプットファイルが必要になります。タリーを除いて同じものを使わないと矛盾が生じます。maxcas, maxbch は、ダンプファイルに指定されているものを使いますので変更できません。最初に計算したときと異なるタリーを入れて再計算したいときなど有効です。ただし、`dumpall` の機能を利用するときは、巨大なデータファイルが作成されることがありますから注意してください。この dumpall オプションは、GG のときのみ有効です。

バイナリ形式で書き出したデータは、マシン間での互換性がありませんので注意してください。以下に書き出されるデータの種類と順序を説明します。

(1) NCOL
NCOL は、プログラムの中の変数で、粒子輸送のタイミングの認識変数です。

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

NCOL=1, 2, 3 の時は、書き出しは NCOL だけです。従って、以下の書き出しは、NCOL≥4 の時です。

(2) NOCAS, NOBCH, RCASC, RSOUIN
これら 4 つの変数は、NCOL=4 の時にだけ書き出されます。それぞれの意味は、

NOCAS	:	current event 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
これらの変数の意味は、

NO	:	cascade id in this event
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
この変数は、ITYP=12, 13 の時、すなわち電子、陽電子の時のみ出力されます。QS は、電子の dE/dx です。

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

(4) IBLZ1, IBLZ2, ILEV1, ILEV2
これらの変数の意味は、

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
階層構造をもつ CELL の変数です。書き出しは、ILEV1>0 の時に

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

で行われます

(b) ILAT2
階層構造をもつ CELL の変数です。書き出しは、ILEV2>0 の時に

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

で行われます

(5) COSTH, UANG(1), UANG(2), UANG(3), NSURF
各変数の意味は以下の通りです。これらはバージョン 2.30 から出力されるようになりました。

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)
これらの変数の意味は、

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

(7) WT, U, V, W
これらの変数の意味は、

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
これらの変数の意味は、

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
これらの変数の意味は、

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
これらの変数の意味は、

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

(11) NZST
この変数は粒子がもつ電荷 (charge state) を表します。

(12) NCLSTS
この変数は、Collision が起こった時に、すなわち、NCOL=13, 14 の時にのみ書き出されます。意味は、衝突もしくは崩壊による生成粒子の個数です。原子核も含みます。NCLSTS>0 の時に、次の変数が書き出されます。

(a) MATHZ, MATHN, JCOLL, KCOLL
これらの変数の意味は、

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

JCOLL, KCOLL はそれぞれ次のような意味を持ちます。

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

これらの変数は、各生成粒子の情報を記述するもので、NCLSTS だけ繰り返し、また、それぞれの変数も以下のように配列を持ちます。書き出しは以下のように行われます。

```
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
```

これらの変数の意味は、

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)

5.2.26 Event Generator Mode

通常、核データを使う計算では、イベント毎のエネルギーと運動量は保存しません。モンテカルロのシミュレーションを多数回繰り返して、平均を取ると、エネルギーと運動量が保存することが保証されます。これは、核データを用いたモンテカルロシミュレーションが、一体の Boltzmann 方程式を数値的に解いていることに対応しています。従って、核データを用いた計算では、一体の観測量しか計算できません。例えば、ある領域の熱量、フラックスなどです。これらは平均値です。また、モンテカルロ計算で用いる核データも一体 (inclusive) のデータしか含まれていません。平均値の周りの分散は、二体以上の分布関数に関係しますから、一体の Boltzmann 方程式では導くことはできません。例えば、PHITS の [t-heat] にある、deposit エネルギー分布は、平均値の周りの分散ですから、二体以上の分布関数によって決まる量です。ですから、核データを用いた計算では求めることができません。反応前と反応後で入射粒子と原子核が変化しないのであれば、つまり、弾性散乱だけが起こると仮定すると、一体の情報から全てのエネルギーと運動量が決まりますから、計算できるわけです。これは極めて例外的な状況で、反応後の生成粒子が変化したり、複数になったりすると破綻します。

高エネルギーの核反応に対しては、通常、核データが整備されていないので、カスケードモデル、JAM、QMD、等の核反応モデルを利用します。これらのモデルは、ひとつの核反応毎に、エネルギー、運動量の保存したモンテカルロ手法で核反応を計算します。このような核反応モデルを用いた輸送計算は、全体として、ひとつのイベント毎にエネルギー、運動量の保存したシミュレーションになっていますから、実際の現象を模擬するという意味で、Event Generator と呼ばれています。輸送コードとして、核データを使ったシミュレーションと似ていますが、概念的には全く異なるものです。このシミュレーションでは、ひとつのイベント毎の観測量は物理的に意味を持ちません。ヒストリーで平均した量だけが物理的に意味を持ちます。平均値周りのイベントによる分散も計算できますが、これは、モンテカルロ手法の統計分散で、物理的なものではありません。物理的な分散を記述する二体以上の相関の情報が核データにも基礎となる方程式にも含まれていないからです。一方、Event Generator では、観測量の分散は物理的な分散に対応します。その精度の問題はありますが、全ての相関を記述している枠組みになっています。高エネルギーのシミュレーションは、大抵この Event Generator になっていて、一体の観測量のほかに、コインシデンス実験の解析や、測定器のレスポンス関数のシミュレーションに応用することができます。

最近、低エネルギーの中性子の関係した分野でも、従来の一体の観測量、熱量やフラックスのほかに、deposit エネルギー分布、2 粒子相関などの高次の相関が含まれる物理量が求められてくるようになりました。例えば、半導体素子の放射線によるソフトエラーの問題、細胞の放射線による影響の問題などは、微視的な領域での deposit エネルギーの分散が問題になります。これらの問題に従来の核データを用いたこのシミュレーションでは対応することができません。そこで、PHITS では、核データを使う低エネルギー領域でも Event Generator としてシミュレートできるモデルを作りました。これが、`e-mode = 1,2` で実行される EventGenerator Mode [51] [52] [53] です。

このモデルの詳細は、他の論文を見ていただくこととして、ここでは、まず、モデルの概略を説明します。中性子入射反応で核データが記述するものは、全断面積、Capture、Elastic、(n,n')、(n,Nn') 等のチャンネル毎の断面積、それと放出粒子の inclusive な微分断面積です。これらの情報からは、残留核の運動量、2 粒子の相関などは一意に決められません。情報が足りないからです。そこで、PHITS では、これら核データの情報を用いるとともに、特殊な統計崩壊モデルを導入しました。まず、核データのチャンネル毎の断面積を用います。次に、チャンネル毎に次のようなモデルを設定します。まず、Capture の場合、入射中性子が Capture された場合の融合原子核の運動量、励起エネルギーは運動学的に一意に決定されますから、この原子核に、中性子の崩壊確率をゼロにした統計崩壊を行います。この場合、光子の放出、荷電粒子の放出だけが含まれる統計崩壊を行うことになります。`e-mode = 1` では放出する粒子は区別されませんが、`e-mode=2` にすると反応チャンネルを反映して放出する粒子を決定します。たとえば、(n,γ) 反応なら必ず光子のみを放

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

出、(n, α) 反応なら α 粒子一個と光子のみを放出する、ということになります。次に、Elastic では、核データに従って、放出中性子の運動量を決定します。この運動量から運動学的に残留核の運動量が一意に決定されます。(n, n') では、核データに従って、まず、放出中性子の運動量を決定します。この運動量から運動学的に残留核の運動量と励起エネルギーが一意に決定されますから、この原子核に、中性子の崩壊確率をゼロにした統計崩壊を行います。Capture の時と同じように、光子の放出、荷電粒子の放出だけが含まれる統計崩壊を行うことになります。Elastic では e-mode=1 も 2 も結果は同じになります。最後に、(n, Nn') 反応ですが、e-mode=1 の場合はまず、核データを用いて一個目の放出中性子の運動量を決定します。この運動量から運動学的に残留核の運動量と励起エネルギーが一意に決定されますから、この原子核に、通常の統計崩壊を行います。即ち、中性子崩壊も含んだ、光子、荷電粒子の統計崩壊を行うことになります。この場合だけ、終状態の中性子の数が核データで指定された N になる保証はありませんが、そのずれは小さいことを確認しています。e-mode=2 では、核データを元にした統計サンプリングを行うことで、N 個の中性子全ての運動量を同時に決定します。この運動量から運動学的に残留核の運動量と励起エネルギーが一意に決定され、続いて光子放出のみを考慮する統計崩壊を行います。このとき終状態の中性子の数は必ず N 個となり、二個目以降の中性子の運動量も核データから決定されるため、e-mode=1 より精度が上がります。このようなモデルを用いることにより、核データを用いた低エネルギーの中性子の反応も、イベント毎にエネルギー、運動量の保存した Event Generator になり、核データだけでは記述できなかった、残留核の運動量、2 粒子相関など全ての情報がイベント毎に記述できるようになります。

次に、e-mode を使う場合の注意点を示します。まず、dmax(2) を設定し、核データを用意し、中性子の核データによる計算を可能にします。粒子放出の精度をあげるために、原子核の基底状態の近くの励起状態の情報を必要としますので、igamma=1-3 が必要になります。次に、統計崩壊モデルは、GEM 即ち nevap=3 が必要です。これらは、e-mode=1,2 を指定すると自動的に設定されます（特に指定しなければ igamma は 2 になります)。もし過去の計算と比較するなどの理由で e-mode=1, igamma=1 を使いたい場合は、igamma=1 としても計算はできます。

次に、e-mode を使うと可能になる新しい観測量を示します。[t-deposit] に関して、通常 dmax(2) 以下のエネルギーの中性子による付与エネルギーはカーマ近似によって計算しますが[54]、e-mode を使う場合は、中性子反応から生成される 2 次荷電粒子や残留核の寄与を計算して付与エネルギーとします。当然この場合、中性子のカーマ近似による計算は考慮されません。[t-yield], [t-product] では、通常、dmax(2) 以下のエネルギー、即ち、中性子の核データを使う領域の生成核種、生成粒子はタリーされませんが、これが可能になります。[t-dpa] は、通常 dmax(2) 以下のエネルギーの中性子による寄与を計算しませんが、e-mode を使うことで考慮するようになります。

[54] バージョン 3.05 以降の仕様です。以前の version では、[t-deposit] ではカーマ近似を使った計算ができませんでした。

5.3 [Source] セクション

このセクションでは線源粒子の情報を定義します。粒子を発生させる領域の形状は s-type により指定し、表 36 にある値が選択できます。[source] で指定できるパラメータには、s-type 固有のものと共通のものがあり、同じパラメータでも s-type によって意味が変わる場合がありますのでご注意ください。これらの詳細については、次ページ以降の説明をご参照ください。

粒子のエネルギーは、単色であれば e0 で指定し、分布をもつ場合は e-type により指定します[55]。e-type の指定方法については、5.3.19 節をご覧ください。エネルギーの単位は原子核を除いて MeV です。原子核のエネルギーは核子あたりの MeV を表す MeV/n で指定します。

表 36: ソースタイプ

ソースタイプ	説明
s-type = 1, (4)	円柱（円面、ペンシルビーム）。
s-type = 2, (5)	角柱（長方形）。
s-type = 3, (6)	ガウス分布（x, y, z 独立）。
s-type = 7, (8)	一般パラボラ分布（x, y, z 独立）。
s-type = 9, (10)	球及び球殻分布。
s-type = 11	位相空間線源。
s-type = 12	Decay-turtle の出力を読み込む。
s-type = 13, (14)	ガウス分布（xy 平面）。
s-type = 15, (16)	一般パラボラ分布（xy 平面）。
s-type = 17	Dump data 線源（dump ファイルの読み込み）。
s-type = 18, (19)	円錐形状。
s-type = 20, (21)	三角柱形状。
s-type = 22, (23)	xyz メッシュ空間分布。
s-type = 24	連続四面体線源。
s-type = 26	面上分布。
s-type = 100	ユーザー定義のソースプログラム。 usrsors.f にプログラムを書きこみコンパイルする。

表 36 中の s-type=の後にある括弧内の数字は、バージョン 2.94 以前の仕様です。なお、バージョン 2.95 以降で e0 と e-type の両方が定義されている場合は、従来の s-type による指定が優先されますので、ご注意ください。例えば、s-type=1 であれば e0 が、s-type=4 であれば e-type の定義が有効となります。

[55] バージョン 2.95 以降の仕様です。バージョン 2.94 以前は、e0 と e-type のどちらを指定するかで、使用する s-type を使い分けていました。例えば、円柱形状で単色であれば s-type=1 とした上で e0 を定義し、エネルギー分布をもつ円柱形状線源であれば、s-type=4 とした上で e-type を定義していました。

5.3.1 <Source> : マルチソース

マルチソースの機能を使用して、エネルギーや線種を変えた線源を複数定義することができます。各線源を、<source>=相対比、の書式で始まるサブセクションで定義し、各線源の相対的な強度を “相対比” として与えます。各<source>サブセクションでは 1 つの s-type を指定し、その指定に伴うパラメータを書きます。ただし、<source>=の上の行で reg パラメータを指定すると<source>の行が継続行として扱われてエラーになるため、reg は各<source>サブセクションの最後に書かないように注意してください。

totfact=は規格化定数を意味し、タリー結果にその絶対値が掛かります。この値が正の場合は、与えられた “相対比” にしたがって各線源の粒子を発生させます。一方、負の場合は、各線源の粒子を同確率で発生させ、与えられた相対比にしたがって各粒子のウエイトを変更します。線源強度の相対比が極端に違う場合は、totfact を負で与えることにより、強度の大小にかかわらず統計量を増やすことができます[56]。

表 37: マルチソース

パラメータ	説明
<source> =	マルチソースの開始、数字はこのソースの相対比。
totfact =	(D=1) マルチソース全体の規格化定数。 正の数で与えた場合、相対比に従ってソース粒子を生成。 負の数で与えた場合、同数のソース粒子を生成し、相対比に従って weight を変化させる。
iscorr =	(D=0) マルチソース相関オプション。 0: 通常のマルチソース。 1: 複数のマルチソースを 1 つのイベントとしてまとめて発生させる。核反応により生成する複数の 2 次粒子を 1 つのイベントとして計算したい場合などに用いる。ただし、iscorr=2,3 の場合と違い、複数のソースが発生する場所は同一ではなく、それぞれが指定したソースの発生領域から確率的に選ばれる。 2: 1 と同じく複数のマルチソースを 1 つのイベントとするが、2 つ目以降の線源を 1 つ目の線源と同じ場所から発生させるオプション。任意の場所で発生する核反応を再現するときなどに使う。 3: 1 と同じく複数のマルチソースを 1 つのイベントとするが、2 つ目以降の線源を 1 つ目の線源と同じ場所から逆方向に発生させるオプション。任意の場所で 2 つに崩壊する粒子を再現するときなどに使う。 iscorr=1,2,3 の場合、<source>は整数の絶対値で定義し、その合計値を totfact で指定する必要があります。例えば、$(X, 1p2n)$ 反応を 1 イベントとして模擬する際は、陽子線源に対して<source>=1、中性子線源に対して<source>=2 とし、totfact=3 とします。

[56] バージョン 3.04 以前では、ウエイトを変更する線源を設定すると、[t-deposit] で output=deposit とする場合や [t-deposit2] を使用する場合には利用できませんでした。

5.3.2 共通パラメータ

各ソースタイプに共通なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。原子核や軽複合粒子のエネルギーは、核子当たりで指定します。

表 38: 共通ソースパラメータ 1

パラメータ	説明
`proj =`	入射粒子 : 粒子の指定方法は表 4 を参照。

`sx =`
`sy =`
`sz =`
	(D=0) スピンの方向ベクトルの x 成分。
	(D=0) スピンの方向ベクトルの y 成分。
	(D=0) スピンの方向ベクトルの z 成分。

`reg =` `ntmax =`	(D=all) `[cell]` セクションで定義したセル番号を指定して、ソースを発生させる領域を限定します。すなわち、ソースタイプ毎に決まる領域とここで指定する領域が重なった部分からソースを発生させます。書式は `reg = { 1 - 5 } 10 34` です。また、`reg = ( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造も指定できます。詳細は、6.1.1 節にあるタリーの領域指定箇所を参照して下さい。 (D=1000) 領域限定の際の最大再試行回数。
`trcl =`	(D=なし) 座標変換番号もしくは座標変換定義。
`wgt =`	(D=1.0) ソース粒子のウエイト。
`factor =`	(D=1.0) ソース粒子の規格化定数。全てのタリー結果を `factor` 倍します。ただし、マルチソースの場合は使えませんので、規格化は `totfact` を用いて行って下さい。また、マルチソースでない場合、`totfact * factor` が実質的な規格化定数になります。
`izst =`	(D=`proj` で指定した粒子の電荷) 入射粒子の電荷 (charge state)。 `[magnetic field]` や `[electro magnetic field]` セクションで定義できる磁場や電場中の運動を記述する際にのみ使用されます。また，`[parameters]` において `ifixchg=1` とした場合は，ATIMA で計算する阻止能もこの電荷となります。ここで指定した値は輸送の途中で変化しません。ただし、核反応で生成された原子核の電荷は Z に戻る。
`cnt(i) =`	(D=0) ソース粒子の counter 初期値。i は counter 番号 (i=1–3）に対応します。各`<source>`で異なるカウンター初期値を設定することにより，それぞれの寄与を区別してタリーすることが可能です。

入射粒子の方向は `dir`，`phi`，`dom` の 3 つパラメータを用いて指定します。図 8 にこれらと入射粒子の方向との関係を示します。入射粒子の方向を太い実線で表しており、極角と方位角をそれぞれ θ, ϕ とした場合に、$\cos\theta$ =`dir`, ϕ =`phi` の関係があります。ただし、`dir` は $\cos\theta$ の値を、`phi` は degree 単位の値を与えます。`phi` は省略可能で、その場合はランダムとなります。`dom` は入射粒子を発生させる際、一定の広がりをもたせる場合に利用します。`dir`，`phi` で決定した方向を中心に、平面角で ψ =`dom`、すなわち立体角として $2\pi(1-\cos\psi)$ の範囲で入射粒子の方向をランダムに決定します。`dom` の値も degree 単位です。

入射粒子の方向を等方分布させたい場合は `dir=all` と設定します。他に、`dir=data` とし、その下に `a-type` サブセクションを用意することにより、任意の関数やデータの角分布をもつ線源を模擬できます。

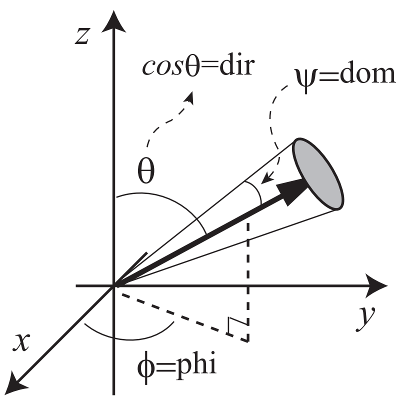


図 8: 入射粒子の方向とパラメータ `dir`，`phi`，`dom` の関係

dir は、s-type=9 とそれ以外で意味が違うので注意してください。また、s-type=11,12 では dir= ±1 しか指定できません。

trcl の指定により、座標変換が使えます。wgt は、ソース粒子の初期ウエイトです。また、規格化定数 factor は、タリーなどの出力に乗じる定数です。dump ファイル読み込みの際に、オリジナルのソース当たりの出力にする場合などに便利です。wgt と factor は、逆数の関係になります。

スピンは、指定しないか全てゼロの場合は、未定義のまま磁場に入ります。この場合、磁場に入ったときの磁場の方向と偏極率に従い、初期スピンが定義されます。ここで、スピンが指定された場合は、磁場の定義の偏極率に関係なく、ここで定義されたスピンの方向で磁場に入ります。

表 39: 共通ソースパラメータ 2

パラメータ	説明
ispfs = 0, 1, 2	(D=0) 自発核分裂からの中性子線源。 ispfs = 1,2 を指定し、proj として、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 の放射性核種を指定した時、これらの核種が自発核分裂を起こした場合に放出される中性子を線源とすることができる。 ispfs=1 の場合、タリーの計算結果が自発核分裂の回数当たりで規格化されます。これに対し、ispfs=2 の場合は、自発核分裂で発生した中性子 1 個当たりで規格化されます。 この線源を使用した場合、[source] セクションで指定した e0 や e-type で指定されるエネルギー分布、dir や a-type で指定される角度分布は無視されます。RI 線源機能 (e-type=28,29) とは違い、proj で指定した粒子ではなく、中性子が線源粒子となりますのでご注意ください。
ibatch =	(D=all) itall=4 の場合に各マルチソースが有効となるバッチ番号を指定する。書式は all, もしくはバッチ番号です。1-5 のように複数バッチ番号をまとめて指定することができます。

ispfs を用いた中性子線源については、文献[57] にあるデータを参照しました。また、本文献にあるデータを用いる際、これらのデータ及び記述する FORTRAN プログラムの開発については、株式会社ナイスの Liem Peng Hong 氏のご協力を得ました。ここに氏への謝意を表します。

次のように、proj に Pu-240 を指定し、ispfs=1 とすると、Pu-240 の自発核分裂による放出される中性子が線源粒子となります。この場合、e0 の値も定義されていますが、無視されます。

```
[ S o u r c e ]
    s-type =   1
      proj =   240Pu
        e0 =   1.0
        z0 =   0
        z1 =   0
        r0 =   0.0
       dir =   1.0000
     ispfs = 1
```

[57] 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 円柱分布ソース

円柱分布ソースタイプに必要なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 40: 円柱分布ソースパラメータ

s-type = 1	円柱のとき。
x0 =	(D=0.0) 円柱の中心の x 座標 [cm]。
y0 =	(D=0.0) 円柱の中心の y 座標 [cm]。
z0 =	(D=0.0) 円柱の z 軸方向の下限 [cm]。
z1 =	(D=0.0) 円柱の z 軸方向の上限（円面の場合は下限と同じ）[cm]。
r0 =	(D=0.0) 円柱の半径（r0=0.0 のときペンシルビーム）[cm]。
r1 =	(D=0.0) 円柱の内半径 [cm]。中空部分を指定する。
dir =	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	all を指定した時は、等方分布。
	data を指定した時は、a-type サブセクションが必要。
phi =	(D=all) 入射粒子の方位角 [degree]。
	= all; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	= −1 ; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

5.3.4 角柱分布ソース

角柱分布ソースタイプに必要なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 41: 角柱分布ソースパラメータ

s-type = 2	角柱のとき。
x0 =	(D=0.0) 角柱の下限の x 座標 [cm]。
x1 =	(D=0.0) 角柱の上限の x 座標 [cm]。
y0 =	(D=0.0) 角柱の下限の y 座標 [cm]。
y1 =	(D=0.0) 角柱の上限の y 座標 [cm]。
z0 =	(D=0.0) 角柱の下限の z 座標 [cm]。
z1 =	(D=0.0) 角柱の上限の z 座標（長方形の場合は下限と同じ）[cm]。
dir =	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	all を指定した時は、等方分布。
	data を指定した時は、a-type サブセクションが必要。
phi =	(D=all) 入射粒子の方位角 [degree]。
	= all; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	= -1 ; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

5.3.5 ガウス分布ソース (x, y, z 独立)

このガウス分布は、x, y, z 軸で独立なガウス分布です。ガウス分布 (x, y, z 独立) ソースタイプに必要なパラメーターを以下に示します。パラメーターの順序は自由です。(D=***) のあるものは、省略可能です。

表 42: ガウス分布 (x, y, z 独立) ソースパラメータ

s-type = 3	ガウス分布（x, y, z 独立）のとき。
x0 =	(D=0.0) ガウス分布の中心の x 座標 [cm]。
x1 =	(D=0.0) ガウス分布の x 方向の半値全幅 [cm]。
y0 =	(D=0.0) ガウス分布の中心の y 座標 [cm]。
y1 =	(D=0.0) ガウス分布の y 方向の半値全幅 [cm]。
z0 =	(D=0.0) ガウス分布の中心の z 座標 [cm]。
z1 =	(D=0.0) ガウス分布の z 方向の半値全幅 [cm]。
dir =	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	all を指定した時は、等方分布。
	data を指定した時は、a-type サブセクションが必要。
phi =	(D=all) 入射粒子の方位角 [degree]。
	= all; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	= −1 ; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

5.3.6 一般パラボラ分布ソース (x, y, z 独立)

このパラボラ分布は、x, y, z 軸で独立なパラボラ分布です。一般パラボラ分布 (x, y, z 独立) ソースタイプに必要なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 43: 一般パラボラ分布 (x, y, z 独立) ソースパラメータ

s-type = 7	一般パラボラ分布 (x, y, z 独立) のとき。
x0 =	(D=0.0) X パラボラ分布の中心の x 座標 [cm]。
x1 =	(D=0.0) X パラボラ分布の幅 [cm]。
y0 =	(D=0.0) Y パラボラ分布中心の y 座標 [cm]。
y1 =	(D=0.0) Y パラボラ分布の幅 [cm]。
z0 =	(D=0.0) Z パラボラ分布の中心の z 座標 [cm]。
z1 =	(D=0.0) Z パラボラ分布の幅 [cm]。
rn =	(D=2) 一般パラボラの次数。
dir =	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	all を指定した時は、等方分布。
	data を指定した時は、a-type サブセクションが必要。
phi =	(D=all) 入射粒子の方位角 [degree]。
	= all; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	= -1 ; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

5.3.7 ガウス分布ソース (*xy* 平面)

このガウス分布は、*xy* 平面でのガウス分布です。ガウス分布 (*xy* 平面) ソースタイプに必要なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 44: ガウス分布 (*xy* 平面) ソースパラメータ

s-type = 13	ガウス分布 (*xy* 平面) のとき。
x0 =	(D=0.0) ガウス分布の中心の x 座標 [cm]。
y0 =	(D=0.0) ガウス分布の中心の y 座標 [cm]。
r1 =	(D=0.0) ガウス分布の半値全幅 [cm]。
z0 =	(D=0.0) z 軸方向の下限 [cm]。
z1 =	(D=z0) z 軸方向の上限 [cm]。
dir =	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	all を指定した時は、等方分布。
	data を指定した時は、a-type サブセクションが必要。
phi =	(D=all) 入射粒子の方位角 [degree]。
	= all; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	= -1 ; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

5.3.8 一般パラボラ分布ソース (*xy* 平面)

このパラボラ分布は、*xy* 平面でのパラボラ分布です。一般パラボラ分布 (*xy* 平面) ソースタイプに必要なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 45: 一般パラボラ分布 (*xy* 平面) ソースパラメータ

s-type = 15	一般パラボラ分布 (*xy* 平面) のとき。
x0 =	(D=0.0) パラボラ分布の中心の x 座標 [cm]。
y0 =	(D=0.0) パラボラ分布の中心の y 座標 [cm]。
r1 =	(D=0.0) パラボラ分布の幅 [cm]。
z0 =	(D=0.0) z 軸方向の下限 [cm]。
z1 =	(D=z0) z 軸方向の上限 [cm]。
rn =	(D=2) 一般パラボラの次数。
dir =	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	`all` を指定した時は、等方分布。
	`data` を指定した時は、`a-type` サブセクションが必要。
phi =	(D=`all`) 入射粒子の方位角 [degree]。
	= `all`; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	= −1 ; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

5.3.9 球及び球殻分布ソース

球及び球殻分布ソースタイプに必要なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 46: 球及び球殻分布ソースパラメータ

s-type = 9	球及び球殻分布のとき。
x0 =	(D=0.0) 球及び球殻分布の中心の x 座標 [cm]。
y0 =	(D=0.0) 球及び球殻分布の中心の y 座標 [cm]。
z0 =	(D=0.0) 球及び球殻分布の中心の z 座標 [cm]。
r1 =	(D=0.0) 球及び球殻分布の内半径 [cm]。ゼロの場合は球。
r2 =	(D=0.0) 球及び球殻分布の外半径 [cm]。
dir =	(D=1.0) 入射粒子の方向。
	`dir = 1.0` : 球の中心からの外向き法線方向。
	`dir = -1.0` : 球の中心からの法線の内向き方向。
	`dir = all` : all を指定した時は、等方分布。
	`dir = -all` : 球の中心からの法線の内向き cos 分布。$\cos^2$ の bias がかかる。領域の体積や表面積の計算に用いることができる。
	`dir = data` : この場合は、`a-type` サブセクションが必要。
	`dir = iso` : 中心から `r1` の距離にある半径 `r2` の円面から垂直に球側に向けて一様分布。
ag1 =	(D=-1.0) `dir=iso` の場合の天頂核カットオフ最小値。
ag2 =	(D=1.0) `dir=iso` の場合の天頂核カットオフ最大値。
pg1 =	(D=0.0) `dir=iso` の場合の方位角カットオフ最小値 [degree]。
pg2 =	(D=360.0) `dir=iso` の場合の方位角カットオフ最大値 [degree]。
isbias =	(D=0) `ag1,ag2,pg1,pg2` を使って角度範囲を指定した際、カットオフされた粒子の処理方法。
	=0 エネルギー 0 の光子を発生させることにより、次のヒストリーに進む。
	=1 許容角度内に粒子が発生するまで再サンプリングする。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

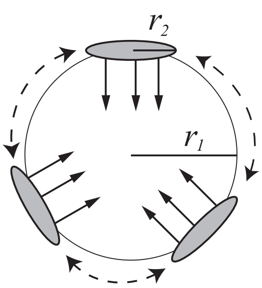


図 9: `dir=iso` の場合のイメージ図

`s-type=9` のソースを flux を用いた体積、面積計算に用いるには、`dir=-all, r1=r2` とします。その際の球の内側におけるフラックスは $1/\pi/(\mathrm{r1})^2$ となります。また、`dir=iso` とした場合も同様となります。

r1 と r2 の関係は、dir=iso の場合を除いて r1≤r2 としてください。dir=iso の場合は基本的に r1=r2 としてください（r1>r2 でも可）。

ag1,ag2,pg1,pg2 を使って角度範囲を指定した場合、その範囲外の粒子がサンプリングされた場合は、その粒子をエネルギー 0 の光子として即座にカットオフします。したがって、不要な方向からの粒子を発生させずに計算時間を短縮することができます。一方、即座にカットオフされた光子も線源としてカウントされますので、規格化定数 (totfact) は、ag1 や ag2 を指定しない場合（例えば、単位フルエンス ($/\mathrm{cm}^2$) あたりに規格化したい場合は線源面の表面積 $\pi(r2)^2$）から変更する必要はありません。ただし、isbias=1 とした場合は、許容角度内に粒子が発生するまで再サンプリングされますので、計算時間は短縮できますが、予めサンプリングが成功する割合を計算して totfact に乗じておく必要があります。

5.3.10 位相空間線源

ビーム方向に垂直な位相空間で指定するソース分布です。加速器における RMS エミッタンス（エネルギーによる規格化はされていません）の線源を模擬する際、有用となります。以下、位相空間線源で必要なパラメータを示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 47: 位相空間線源のパラメータ

`s-type = 11`	位相空間線源のとき。
`x0 =`	(D=0.0) ビーム中心の x 座標 [cm]。
`y0 =`	(D=0.0) ビーム中心の y 座標 [cm]。
`z0 =`	(D=0.0) z 軸方向の下限 [cm]。
`z1 =`	(D=0.0) z 軸方向の上限 [cm]。
`rx =`	(D=0.0) x 方向における位相空間の楕円の傾き [rad]。
`ry =`	(D=0.0) y 方向における位相空間の楕円の傾き [rad]。
`wem =`	(D=0.0) 位相空間からのサンプリング方法（図 10 参照）。
	`wem > 0`: 位相空間に一様な線源。`wem` は RMS エミッタンス [πcm × mrad] を表す。
	`wem = 0`: 位相空間にガウス分布する線源。
`x1 =`	(D=0.0)
	x 方向の最大半径/最大角度の比 [cm/mrad]。(`wem>0` の場合)
	`rx=0` における x 方向のガウス分布標準偏差 [cm]。(`wem=0` の場合)
`y1 =`	(D=0.0)
	y 方向の最大半径/最大角度の比 [cm/mrad]。(`wem>0` の場合)
	`ry=0` における y 方向のガウス分布標準偏差 [cm]。(`wem=0` の場合)
`xmrad1 =`	(D=0.0) `rx=0` における x 方向角度のガウス分布標準偏差 [mrad]。(`wem=0` のみ有効)
`ymrad1 =`	(D=0.0) `ry=0` における y 方向角度のガウス分布標準偏差 [mrad]。(`wem=0` のみ有効)
`x2 =`	(D=0.0) x 方向の位相空間中心 [cm]。
`y2 =`	(D=0.0) y 方向の位相空間中心 [cm]。
`xmrad2 =`	(D=0.0) x 方向角度の位相空間中心 [mrad]。
`ymrad2 =`	(D=0.0) y 方向角度の位相空間中心 [mrad]。
`dir =`	(D=1) 入射粒子の軸方向の方向余弦。1 か −1 のみ指定可能。
`e0 =`	（単色の場合）入射粒子のエネルギー [MeV/n]。
`e-type =`	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

サンプリングされた位相空間座標が $(X, X'), (Y, Y')$ の場合、位相空間の傾きと中心のずれを考慮した後の座標 $(X_m, X'_m), (Y_m, Y'_m)$ は以下の数式で計算されます。

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

最終的な線源の発生位置 x[cm],y[cm] 及びその方向ベクトル (u, v, w) は以下の計算式で決定されます（た

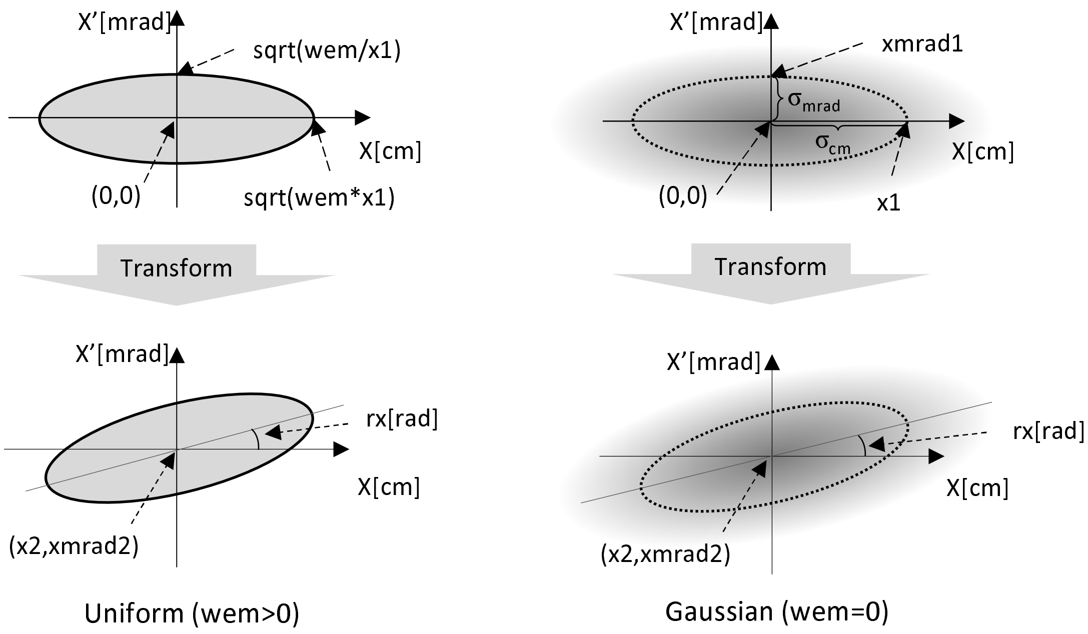


図 10: 位相空間からのサンプリング方法

だし `dir=1.0` の場合)。

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

例えば、位相空間にガウス分布する線源で 10π mm·mrad の rms エミッタンスを設定したい場合、`wem=0` とした上、`x1=0.2`、`xmrad1 = 5.0` というように、x と `xmrad` と π の積、つまり位相空間図上の距離 rms と角度 rms で表される楕円面積 (`x1*xmrad*`π mm·mrad) の大きさを包括する 2 次元ガウス分布を設定します。なお、`x2` や `xmrad2` などは、ビーム軸やビーム発散角の軸のずれを表現するためのパラメータで、理想的なビームに対しては 0 となります。一方、`wem>0` の場合は、$x = \sqrt{wem * x1}$ と `xmrad=` $\sqrt{wem/x1}$ と π の積で表される位相空間上の楕円の中に、発生線源のすべてが一応に収まるような分布の線源となります。この時、楕円の内部以外の部分には線源分布は発生しませんので、一般的な加速器ビームを模擬する場合は、`wem=0` を選択して、rms エミッタンスで表現される分布以外にも粒子分布を発生することができる 2 次元ガウス分布形状にしてお使いください。

5.3.11 s-type = 12 ソース

decay-turtle の出力をソースとして取り込みます。`s-type=12` の分布ソースタイプに必要なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

入力ファイルのソースの数が足りなくなった場合は、初めに戻って同じデータを繰り返し使います。

表 48: s-type = 12 のソースパラメータ

`s-type = 12`	decay-turtle の出力を読み込むとき。
`x0 =`	(D=0.0) ビームの x 座標のオフセット [cm]。
`y0 =`	(D=0.0) ビームの y 座標のオフセット [cm]。
`z0 =`	(D=0.0) ビームの z 座標のオフセット [cm]。
`dir =`	(D=1) 入射粒子の軸方向の方向余弦。1 か −1 のみ指定可能。
`file =`	decay-turtle 出力ファイル名（パスも含む）。

`decay-turtle` の出力のデータ形式は、倍精度、アスキー形式で、各レコードは、

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

で、それぞれは次の様な変数です。

表 49: decay-turtle データ

変数	説明
`xp, yp`	ビーム粒子の入射位置 [cm]。
`xq, yq`	ビーム粒子の進行方向に垂直な面となす角度 [mrad]。
`e0`	ビーム粒子の運動量 [GeV/c]。
`wt0`	ビーム粒子のウエイト。
`pz0`	ビーム粒子の偏極度（未使用）。

5.3.12 円錐形状分布ソース

円錐形状分布ソースタイプに必要なパラメータを以下に示します。線源粒子は円錐の側面からのみ発生します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 50: 円錐形状分布ソースパラメータ

s-type = 18	円錐形状。
x0 =	(D=0.0) 円錐の頂点の x 座標 [cm]。
y0 =	(D=0.0) 円錐の頂点の y 座標 [cm]。
z0 =	(D=0.0) 円錐の頂点の z 座標 [cm]。
x1 =	(D=0.0) 円錐頂点から底面中心に向けたベクトルの x 成分 [cm]。
y1 =	(D=0.0) 円錐頂点から底面中心に向けたベクトルの y 成分 [cm]。
z1 =	(D=0.0) 円錐頂点から底面中心に向けたベクトルの z 成分 [cm]。
r0 =	(D=0.0) (D=0.0) 頂点から円錐上面までの軸上における距離 [cm]。
r1 =	(D=0.0) 頂点から円錐下面までの軸上における距離 [cm]。
r2 =	(D=0.0) 円錐の母線と軸のなす角度 [degree]。
dir =	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	all を指定した時は、等方分布。
	data を指定した時は、a-type サブセクションが必要。
phi =	(D=all) 入射粒子の方位角 [degree]。
	= all; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	= −1 ; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

5.3.13 三角柱形状分布ソース

`s-type=20` とすることで、三角柱形状に線源を発生させることができます。パラメータ `exa` を指定することで、上底面もしくは下底面から、指数関数的 exp($-ax$) にソース強度が減衰する分布、もしくは一様な分布を定義することができます。前者は、`exa` に正の値を与えることで、係数の値 a を指定できます。

三角柱形状分布のソースタイプに必要なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 51: 三角柱状分布ソースパラメータ

`s-type = 20`	三角柱形状。
`x0 =`	(D=0.0) 三角柱の頂点の x 座標 [cm]。
`y0 =`	(D=0.0) 三角柱の頂点の y 座標 [cm]。
`z0 =`	(D=0.0) 三角柱の頂点の z 座標 [cm]。
`x1 =`	(D=0.0) 三角柱頂点から第 1 頂点へのベクトルの x 成分 [cm]。
`y1 =`	(D=0.0) 三角柱頂点から第 1 頂点へのベクトルの y 成分 [cm]。
`z1 =`	(D=0.0) 三角柱頂点から第 1 頂点へのベクトルの z 成分 [cm]。
`x2 =`	(D=0.0) 三角柱頂点から第 2 頂点へのベクトルの x 成分 [cm]。
`y2 =`	(D=0.0) 三角柱頂点から第 2 頂点へのベクトルの y 成分 [cm]。
`z2 =`	(D=0.0) 三角柱頂点から第 2 頂点へのベクトルの z 成分 [cm]。
`x3 =`	(D=0.0) 三角柱頂点から底面頂点へのベクトルの x 成分 [cm]。
`y3 =`	(D=0.0) 三角柱頂点から底面頂点へのベクトルの y 成分 [cm]。
`z3 =`	(D=0.0) 三角柱頂点から底面頂点へのベクトルの z 成分 [cm]。
`exa =`	(D=0.0) 指数関数減衰の係数。exa>0; exp ($-ax$), exa=0; 一様分布。
`dir =`	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	`all` を指定した時は、等方分布。
	`data` を指定した時は、`a-type` サブセクションが必要。
`phi =`	(D=`all`) 入射粒子の方位角 [degree]。
	= `all`; 0 から 360 の範囲でランダムに決定。
`dom =`	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	= -1 ; cos^2 の bias がかかった分布。
`e0 =`	（単色の場合）入射粒子のエネルギー [MeV/n]。
`e-type =`	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

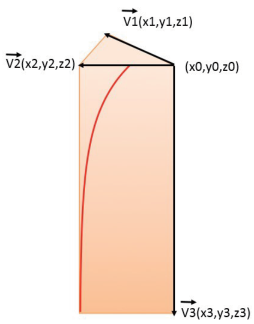


図 11: 三角柱分布の座標の指定

5.3.14 xyz メッシュ空間分布ソース

`s-type=22` により複雑な空間分布をもつ線源を発生させることができます。空間分布を定義するパラメータは、`x`，`y`，`z-type` サブセクションとその直後に記載する相対強度 $I_{ijk}, (i = 1, \cdots, n_x; j = 1, \cdots, n_y; k = 1, \cdots, n_z)$ です。`dir`，`phi`，`dom` によって線源の方向を指定することも可能です。ただし、空間メッシュ毎に方向を変えることはできないので注意してください。表 52 にメッシュ空間分布ソースのパラメータを示します。パラメータの順序は、相対強度のデータを除き、自由です。(D=***) のあるものは、省略可能です。

表 52: xyz メッシュ空間分布ソースパラメータ

`s-type = 22`	xyz メッシュ空間分布。
`mesh = xyz`	`mesh` は `xyz` のみ選択できます。
`x-type =`	`x-type` の型。この下に `x-type` サブセクションが必要。
`y-type =`	`y-type` の型。この下に `y-type` サブセクションが必要。
`z-type =`	`z-type` の型。この下に `z-type` サブセクションが必要。
	`x-,y-,z-type` サブセクションの下に、相対強度 I_{ijk} のデータを並べます。データは x, y, z の順に変化させてください。ただし、`x-,y-,z-type` サブセクションの `nx,ny,nz` が正の場合は昇順、負の場合は降順となります。
`dir =`	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	`all` を指定した時は、等方分布。
	`data` を指定した時は、`a-type` サブセクションが必要。
`phi =`	(D=`all`) 入射粒子の方位角 [degree]。
	= `all`; 0 から 360 の範囲でランダムに決定。
`dom =`	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	= −1 ; $\cos^2$ の bias がかかった分布。
`e0 =`	（単色の場合）入射粒子のエネルギー [MeV/n]。
`e-type =`	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。
`isbias =`	(D=0) 各メッシュから発生させる粒子の数を変化させて線源強度を調整。
	= 1 各メッシュからランダムに粒子を発生させ、その weight を変化させて線源強度を調整。
	= 2 各メッシュから均等に粒子を発生させ、その weight を変化させて線源強度を調整。この場合、一度、粒子が発生したメッシュからは、全てのメッシュから粒子が発生するまで再度粒子が発生しなくなります。ただし、メモリ分散型（MPI）並列の場合は、各ノードで粒子が発生したメッシュを記録しますので、別のノードの計算で同じメッシュから粒子が発生する場合もあります。

相対強度 I_{ijk} のデータの順番は、`nx,ny,nz` が全て正の場合は次のようになります。

$$I_{111}, I_{211}, \cdots, I_{n_x 11}, I_{121}, I_{221}, \cdots, I_{n_x n_y 1}, I_{112}, I_{212}, \cdots, I_{n_x n_y n_z} \tag{2}$$

ここで、n_x =|`nx`|, n_y =|`ny`|, n_z =|`nz`|です。そして、例えば `nx,ny` が正、`nz` が負の場合は、

$$I_{11n_z}, I_{21n_z}, \cdots, I_{n_x 1 n_z}, I_{12n_z}, I_{22n_z}, \cdots, I_{n_x n_y n_z}, I_{11n_z-1}, I_{21n_z-1}, \cdots, I_{n_x n_y 1} \tag{3}$$

のように、$k = n_z$ =|`nz`|となる行列要素を最初に並べてください。

`s-type=22` とした例題を以下に示します。

例 2: xyz メッシュ空間分布ソースの例題

```
 1:   [ S o u r c e ]
 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
```

この例題では、x, y, z 軸に関して -10cm から 10cm の範囲を線源領域とし、それぞれ 3,3,2 に分割しています。ただし、ny と nz は負で定義しているため、相対強度 I_{ijk} は j と k に関して降順に並べます。よって、例題の 19-21 行目には、$k = 2$、すなわち [0cm≤ z ≤10cm] の範囲のデータが並べられており、

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

となります。また、23-25 行目には、$k = 1$、すなわち [-10cm≤ z ≤0cm] の範囲のデータが並んでおり、

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

となります。図 12 に、この条件で発生させた線源の空間分布を示します。[t-product] において output=source として求めており、左図が [0cm≤ z ≤10cm]、右図が [-10cm≤ z ≤0cm] の結果です。左図において、番号 $1,2,3,4,5,6,\cdots$ の領域が、それぞれ $I_{132}, I_{232}, I_{332}, I_{122}, I_{222}, I_{322}, \cdots$ と対応しており、領域番号が増加するにつれて、段階的に線源の発生量が増加しているのがわかります。右図においては、番号 2,4,6,8 の領域で線源が発生しておらず、式 5 の右辺において 0 とした I_{ijk} と対応しています。

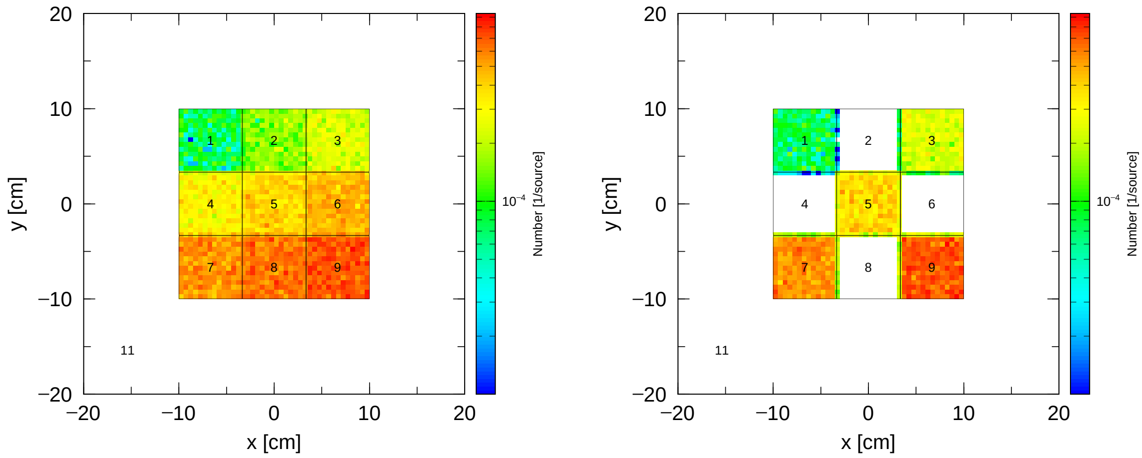


図 12: 例題 2 の線源を `[t-product]` によりタリーした結果。左図が [0cm≤ z ≤10cm]、右図が [−10cm≤ z ≤0cm] の結果である。

5.3.15 連続四面体を線源とするソース

s-type=24 により、体系で定義した連続四面体形状（5.6.5.4 節）の各四面体から発生する線源を考慮することができます。四面体が属する universe のセル番号を指定することで、同じ universe に属する四面体から各四面体の体積の割合に従って粒子が均一に発生します。ここでは、セル番号は一個しか設定できないので、二個以上のセル番号を指定したい場合は、マルチソース（5.3.1 節）を用いて設定してください。**dir, phi, dom** によって線源の方向を指定することも可能です。表 53 に連続四面体ソースのパラメータを示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。\utility\TetraGEOM フォルダにある資料やサンプルインプットもご参照ください。

表 53: 連続四面体ソースパラメータ

s-type = 24	連続四面体形状。
tetreg =	四面体が属する universe のセル番号。1 つしか指定できない。
dir =	(D=1.0) 入射粒子の z 軸方向からの方向余弦。
	all を指定した時は、等方分布。
	data を指定した時は、a-type サブセクションが必要。
phi =	(D=all) 入射粒子の方位角 [degree]。
	= all; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。
	$= -1$; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

s-type=24 とした例題を以下に示します。

例 3: 連続四面体ソースの例題

```
1:  [ S o u r c e ]
2:   s-type = 24
3:     proj = proton
4:   tetreg = 201
5:      dir = all
6:       e0 = 70
```

この例題では、tetreg により指定したセル番号 201 の領域から一様に発生する線源を定義しています。ここで指定するセル番号は、例 44 の 13 行目のように何らかの universe に属している必要があります。

5.3.16 面上分布ソース

特定の面から発生するソースを定義します。面線源では線源の発生位置での面の法線を z 軸として、角度分布は既存の定義が使えます。定義に必要なパラメータを以下に示します。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 54: 面上分布ソースパラメータ

s-type = 26	面線源のとき。
suf =	粒子を発生させる面番号。[surface] セクションで定義した面番号で指定します。指定できる面の種類は PX，PY，PZ，SO，S，SX，SY，SZ，C/X，C/Y，C/Z，CX，CY，CZ で，2 つ以上の面は指定できません。
cut =	粒子を発生する面上の領域を制限します。[surface] セクションで定義した面番号に符号を付けて指定します。複数の面番号を指定することができます。面番号の符号は、[cell] セクションで領域を定義する時と同様の意味を持ちます。ただし、:や ( ) は使えません。例えば cut = 1 -2 3 -4 であれば、面 1 及び 3 のプラスもしくは外側かつ面 2 及び 4 のマイナスもしくは内側で囲まれる領域を指定することになります。
dir =	(D=1.0) 線源の発生位置での面の法線からの方向余弦。all を指定した時は、等方分布。data を指定した時は、a-type サブセクションが必要。
phi =	(D=all) 入射粒子の方位角 [degree]。 = all; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。 = −1 ; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。

suf や cut で指定する面番号を [surface] セクションで定義する際、trcl は使用できません。[source] セクションで指定した trcl は有効です。

5.3.17 dump データソース

s-type=17 とすることで dump ファイルに落とされた粒子情報を線源として利用できます。パラメータ idmpmode を切り替えることにより、粒子情報の取り扱い方法として 2 種類を選択することができます。idmpmode=0 の場合は、dump ファイルに記録された粒子を一つ一つの独立したイベントによる粒子であるとして統計処理を行います。対して idmpmode=1 の場合は、dump ファイルに記録されたヒストリーとバッチの情報を利用することで、dump ファイルを作成した際のイベント情報を基に粒子間の相関を考慮した統計処理を行います。dump ファイルを用いたつなぎ計算を行う場合は、前段階のイベント情報を考慮して統計誤差を評価できるように idmpmode=1 としてください。[58]

このソースタイプに関連するパラメータは以下の通りです。パラメータの順序は自由です。(D=***) のあるものは、省略可能です。

表 55: dump データのソースパラメータ (1)

s-type = 17	dump ファイルを読み込むとき。
file =	dump ファイル名（パスも含む）。
dump = (次行)	dump データの個数。負で与えた時はアスキーファイル。 dump データの情報。
jpsf =	Phase space file 用の特別オプション。z0 の指定が必要。 =1 は変換時に weight 無しの場合。=2 は有りの場合。
(以下省略可能)	以下のものが指定されたとき、dump データがある場合も dump データより優先します。
x0 =	下限 x 座標 [cm]。
x1 =	上限 x 座標 [cm]。
y0 =	下限 y 座標 [cm]。
y1 =	上限 y 座標 [cm]。
z0 =	下限 z 座標 [cm]。
z1 =	上限 z 座標 [cm]。
sx =	スピンの方向ベクトルの x 成分。
sy =	スピンの方向ベクトルの y 成分。
sz =	スピンの方向ベクトルの z 成分。
dir =	入射粒子の z 軸方向からの方向余弦。 all を指定した時は、等方分布。 data を指定した時は、a-type サブセクションが必要。
phi =	(D=all) 入射粒子の方位角 [degree]。 = all; 0 から 360 の範囲でランダムに決定。
dom =	(D=0.0) 入射粒子方向の立体角範囲 [degree]。 = −1 ; $\cos^2$ の bias がかかった分布。
e0 =	（単色の場合）入射粒子のエネルギー [MeV/n]。
e-type =	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。
wgt =	(D=1.0) ソース粒子のウエイト。

[58] idmpmode=1 および dmpmulti による使い回し計算は、日本原子力学会「2015 年春の年会」での波戸氏 (KEK) らの発表 “モンテカルロつなぎ計算における不確かさ評価”（演題番号 N50）を参考に改良を行ったものです。

表 56: dump データのソースパラメータ (2)

(以下省略可能)	以下のものも指定可能です。
factor = t-type = reg = ntmax = trcl =	(D=1.0) ソース粒子のウエイトの規格化定数。 (D=0) 時間分布. (D=all) 領域を限定する。 (D=1000) 領域限定の際の最大再試行回数。 (D=なし) 座標変換番号もしくは座標変換定義。
idmpmode =	dump ソースモードの選択。 (D=0) dump データが nocas と nobach の情報を含まない場合。 (D=1) dump データが nocas と nobach の情報を含む場合。 =0: dump ファイルに記録された粒子情報を 1 つ 1 つの独立したヒストリーとして統計処理を実行する。 =1: dump ファイルを作成した際のヒストリーの情報（粒子間の相関）を考慮し統計処理を実行する。
dmpmulti =	dump ファイルの使い回しの回数。 (D=0.0) idmpmode=0 の場合。 (D=1.0) idmpmode=1 の場合。 =0.0: (maxcas*maxbch) で指定した全ヒストリー数の計算が終了するまで使いまわす。ただし、idmpmode=1 の場合は指定できない。 >0.0: 指定した回数だけ粒子情報を使いまわす。小数点以下の数字はデータを確率的に採用（ロシアンルーレット）することで考慮する。

idmpmode=1 を使用するためには、使用する dump データにヒストリー番号 (nocas) とバッチ番号 (nobch) が含まれている必要があります。PHITS のタリー出力により dump ファイルを生成すると、ファイル名に “_dmp ” が付いた dump ファイルと共に “_dmp ” が付かない通常のタリーファイルが生成されますが（6.7.20 節参照）、idmpmode=1 を使用するためには、この通常タリーファイルも必要とされますので、使用する dump ファイルと同じフォルダに配置してください。idmpmode=1 では、この通常タリーファイルから前段階の maxcas,maxbch の情報を読み込みます。これによりインプットファイルに記述した maxcas,maxbch の値は無視されます。また、idmpmode=1 ではソースの規格化定数 totfact の値が無視されます。ソースの規格化定数 totfact を乗じた計算を行いたい場合は、dump ファイルを作成する前段階の計算において乗じるようにしてください。dump ファイル内の粒子のウェイト値に反映されます。idmpmode=1 はマルチソースと同時に使うことはできません。

dmpmulti は dump ファイルの使い回しの回数を制御します。dmpmulti=2.0 の場合 dump ファイルを 2 回使い回しますが、具体的には、1 個の粒子情報に対してウエイト値を 0.5 倍した 2 個の粒子を線源として発生させた計算を実行します。小数点以下の数字は、使い回す回数をもう 1 つ増やすかどうかに関連し、各々の粒子情報に対して乱数により決定します。例えば、dmpmulti=2.3 の場合、使い回す回数が 70%の確率で 2 回、30%の確率で 3 回となります。また、dmpmulti=0.0 を指定した場合は、maxcas,maxbch で指定される全イベントが終了するまで、dump ファイルを使い回します。ただし、これは idmpmode=0 でのみ有効となります。

dump ファイルを線源として使う計算では、基本的に再開始計算 (istdev<0) はできません。dmpmulti=0.0 の時のみ可能ですが、この場合は idmpmode=0 となるため、dump データの粒子情報を 1 つ 1 つの独立したヒストリーとして扱うことになり、データが十分にないと統計的に偏った結果を与えることになります。よって、統計量を増やした結果を得るためには、基本的には dmpmulti の値を大きく取って再度計算してく

ださい。ただしこの場合も、dump ファイルにあるデータ量が十分でないと、`dmpmulti` を増やしても、求める精度まで統計誤差が小さくならない可能性があるのでご注意ください。データ量が少ない場合は、dump ファイルにデータを蓄える計算の段階からやり直す必要があります。

dump データの情報は、まず `dump =`でデータの個数を指定します。正で与えた時はバイナリーファイルの読み込み、負で与えた時はアスキーファイルの読み込みをします。次の行に 1 つのレコードのデータの並びを指定します。その時の番号と物理量の関係は、以下の通りです。

表 57: dump データの種類と番号 (1)

物理量	kf	x	y	z	u	v	w	e	wt	time	c1	c2	c3	sx	sy	sz
番号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

表 58: dump データの種類と番号 (2)

物理量	name	nocas	nobch	no
番号	17	18	19	20

ここで、kf は粒子を識別する kf-code（表 4 を参照）、x, y, z 単位を cm とする空間座標、u, v, w は運動量の単位ベクトル、e は MeV を単位とするエネルギー（原子核の場合は MeV/n）、wt は粒子ウエイト、time は時間 (単位は nsec)、c1, c2, c3 はカウンターの値、sx, sy, sz はスピンの方向ベクトルです。name は粒子の衝突回数、nocas はあるバッチの中のヒストリー番号、nobch はバッチ番号、no はそのヒストリー中でのカスケード ID です。これらはバイナリーのときは real*8 のデータで、アスキーのときは *n*(1p1d24.15) のデータフォーマットで格納されています。

例えば、9 つのデータが次の順番で並んでいる時、

```
kf  e  wt  x  y  z  u  v  w
```

このデータを読み込むには、

```
dump = 9
  1  8  9  2  3  4  5  6  7
```

と指定します。

5.3.18 ユーザー定義ソース

usrsors.f にプログラムを書き込むことにより、ユーザー定義のソースを利用することができます。以下に示すソースのパラメータをソースセクションに指定した場合は、ユーザーソースで定義された値よりも優先します。

表 59: ユーザー定義ソースで指定できるパラメータ

`s-type = 100`	ユーザー定義ソースのとき。
(以下省略可能)	以下のものが指定されたとき、ユーザー定義データがある場合も ユーザー定義データより優先します。
`x0 =`	下限 x 座標 [cm]。
`x1 =`	上限 x 座標 [cm]。
`y0 =`	下限 y 座標 [cm]。
`y1 =`	上限 y 座標 [cm]。
`z0 =`	下限 z 座標 [cm]。
`z1 =`	上限 z 座標 [cm]。
`sx =`	スピンの方向ベクトルの x 成分。
`sy =`	スピンの方向ベクトルの y 成分。
`sz =`	スピンの方向ベクトルの z 成分。
`dir =`	(D=1.0) 入射粒子の z 軸方向からの方向余弦。 `all` を指定した時は、等方分布。 `data` を指定した時は、`a-type` サブセクションが必要。
`phi =`	(D=`all`) 入射粒子の方位角 [degree]。 = `all`; 0 から 360 の範囲でランダムに決定。
`dom =`	(D=0.0) 入射粒子方向の立体角範囲 [degree]。 = −1 ; $\cos^2$ の bias がかかった分布。
`e0 =`	（単色の場合）入射粒子のエネルギー [MeV/n]。
`e-type =`	（分布をもつ場合）入射粒子のエネルギー [MeV/n]。
`wgt =`	(D=1.0) ソース粒子のウエイト。
`factor =`	(D=1.0) ソース粒子のウエイトの規格化定数。
`t-type =`	(D=0) 時間分布。
`reg =`	(D=all) 領域を限定する.
`ntmax =`	(D=1000) 領域限定の際の最大再試行回数。
`trcl =`	(D=なし) 座標変換番号もしくは座標変換定義。

インプットファイルで `set` により指定したユーザー定義定数 `c1-c99` は、プログラム内で `cval(1-99)` により参照できます。ただし、複数回指定した場合は最後に定義した値が有効となります。

デフォルトで usrsors.f に入っているプログラムは、以下のようなものです。最初のコメント部に、ソースに必要な変数と、粒子指定に必要な kf code のリストがあります。次に、プログラム内で使用できる一様乱数と、ガウス乱数の function の説明があります。プログラムの最初の部分は、初期化の際のファイルのオープンの例があります。最後の部分に必要な変数の値の一つの例が書かれています。この例題を参考にユーザー定義ソースを書いてください。ソースセクションで指定した変数は、優先しますから、その場合は、ここで定義する必要はありません。

ファイル 2: 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 :         Lanmda0                                   *
*             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 )
      data ifirst / 0 /
      save ifirst
      character filenm*50
*-----------------------------------------------------------------------
*        example of initialization
*-----------------------------------------------------------------------
         if( ifirst .eq. 0 ) then
c              filenm = 'input.dat'
c              inquire( file = filenm, exist = exex )
c              if( exex .eqv. .false. ) then
c                 write(*,*) 'file does not exist =>  ', filenm
c                 call parastop( 887 )
c                 end if
c              open(71, file = file(i), status = 'old' )

c              close(71)
               ifirst = 1
         end if
*-----------------------------------------------------------------------
*        example for 3 GeV proton with z-direction
*-----------------------------------------------------------------------
            x = 0.0
            y = 0.0
            z = 0.0
            u = 0.0
            v = 0.0
            w = 1.0
            e = 3000.0
           wt = 1.0
         time = 0.0
         name = 1
           kf = 2212
          nc1 = 0
          nc2 = 0
          nc3 = 0
           sx = 0.d0
           sy = 0.d0
           sz = 0.d0
*-----------------------------------------------------------------------
      return
      end
```

5.3.19 エネルギー分布の定義

エネルギー分布をもつ線源粒子を定義する場合は、`e0` パラメータではなく、`e-type` サブセクションを用います。ただしその際、`e0` はコメントアウトなどで未定義となるようにしてください。また、エネルギーの単位は、原子核以外は MeV、原子核の場合は MeV/n となります。ただし、本節の以下の説明では、エネルギーの単位が MeV の場合のみを示していますので、原子核の場合は適宜 MeV/n に読み替えてください。

表 60 に、利用できる線源のエネルギー分布の種類をまとめます。幾つかの特殊な場合を除き、分布の種類は積分型か微分型で大きく分かれています。積分型はエネルギーに関して積分した強度を定義するのに対し、微分型はエネルギー微分値、すなわち単位が [particles/MeV] で与えられた線源スペクトルを定義してください。また、線源の発生方法には、与えられた強度に対して生成する粒子数を調整するものと、粒子のウエイト値を調整するものがあります。前者は、分布を再現するように、各エネルギービンに生成する粒子の数を統計的に調整します。これに対して後者は、各ビンに同数の粒子を生成させ、粒子のウエイト値を強度に合わせて変化させることにより、分布を再現します。

表 60: 線源のエネルギー分布の種類。発生方法には、粒子のウエイト値を一定にして各ビンで発生させる粒子数を調整する方法と、粒子数を一定にして各ビンでウエイト値を変更する方法があります。e-type に括弧の数字を指定すると、エネルギー点を波長 (Å) で与えることができます。

エネルギー分布の種類	発生方法	e-type	説明
積分型連続エネルギー分布	粒子数 ウエイト値	1, (11) 4, (14)	各エネルギービンの下限値とそのビンにおける線源粒子の生成確率の積分値を与えることにより、連続的なエネルギー分布を指定する。
微分型連続エネルギー分布	粒子数 ウエイト値	21, (31) 24, (34)	各エネルギービンの下限値とそのビンにおける線源粒子の生成確率の微分値を与えることにより、連続的なエネルギー分布を指定する。
離散的エネルギー分布	粒子数 ウエイト値	8, (18) 9, (19)	各エネルギー点とその点における線源粒子の生成確率を与えることにより、離散的なエネルギー分布を指定する。
離散的+積分型連続エネルギー分布	粒子数 ウエイト値	22, (32) 23, (33)	各エネルギービンの下限値・上限値とそのビンにおける線源粒子の生成確率の積分値を与えることにより、任意のエネルギー分布を指定する。
微分型ガウス分布	粒子数	2, (12)	分布の中心値と半値全幅を与えることにより、微分線源スペクトルをガウス分布で与える。
微分型マクスウェル分布	粒子数 ウエイト値	3 7	温度パラメーターを与えることにより、微分線源スペクトルをマクスウェル分布で与える。
微分型任意関数分布	粒子数 ウエイト値	5, (15) 6, (16)	微分線源スペクトルを任意関数で与える。
RI 線源によるエネルギー分布	粒子数 ウエイト値	28 29	放射性核種と放射能を与えることにより、その核種の崩壊に伴って放出される放射線を線源とする。
タリー結果を用いたエネルギー分布	粒子数	20	タリーで求めたエネルギー分布を線源の分布として指定する。
宇宙線フラックスのエネルギー・角度分布	粒子数 ウエイト値	25 26	宇宙空間及び大気圏内における宇宙線フラックスのエネルギー及び角度分布を絶対値も含めて再現する線源。

定義したいエネルギー分布の種類や発生方法に応じて、該当する e-type を指定してください。e-type の値によって、各サブセクションの入力形式が変わります。以下では、各 e-type の入力形式とパラメータについてご説明します。(D=***) のあるものは、省略可能です。

5.3.19.1 積分型連続エネルギー分布

表 61: 線源エネルギー分布パラメータ (1)

パラメータ	説明
e-type = 1, (11)	各エネルギービンの下限値 e(i) とそのビンにおける線源粒子の生成確率の積分値 w(i) を与えることにより、連続的なエネルギー分布を指定する。統計的に w(i) に比例するように各ビンに生成する粒子の数を調整し、エネルギー分布を表現する。11 の時は、エネルギーを波長 (Å) で与える。
ne =	エネルギー群数。ne が正の場合は、各ビン内の微分フラックスを [1/MeV] の単位で表示させた際に一定となるように粒子を生成し、負の場合は、[1/Lethargy] の単位で一定となるように粒子を生成する。 データは自由フォーマットで次の様に与える。 (e(i),w(i),i=1,|ne|), e(|ne|+1) 各ビンに生成される粒子数の積分値は w(i) に比例する。
e-type = 4, (14)	e-type = 1, (11) と同じエネルギー分布を発生させる。ただし、e-type = 1, (11) が生成数を調整してエネルギー分布を表現するのに対し、e-type = 4, (14) では、全てのエネルギービンに同数の粒子を生成させ、粒子のウエイトの積分値を w(i) に比例して変化させることによりエネルギー分布を表現する。また、p-type = 1 とし生成個数比 p(i) を与えることにより、各ビンの生成個数を変化させ、特定のエネルギーをもつ線源の統計量を変化させることができる。14 の時は、エネルギーを波長 (Å) で与える。
ne =	エネルギー群数。ne が正の場合は、各ビン内の微分フラックスを [1/MeV] の単位で表示させた際に一定となるように粒子を生成し、負の場合は、[1/Lethargy] の単位で一定となるように粒子を生成する。 データは自由フォーマットで次の様に与える。 (e(i),w(i),i=1,|ne|), e(|ne|+1) デフォルト（p-type = 0）では各ビンに等しい個数が生成される。 p-type = 1 で p(i) を設定した場合は、それらの値に各ビンの生成粒子数の積分値は比例する。
p-type = 0, 1	(D=0) 生成個数のオプション。 for 0, 全てのビンで p(i)=1、以下のデータは無し。 for 1, 各ビンの生成個数比 p(i) を次の行からデータで与える。 (p(i),i=1,ne)

中性子光学のために、エネルギー分布を波長 (Å) で指定できるオプションを設けました。e-type=1，2 の代わりに e-type=11，12 を指定すると、エネルギーの入力を全て波長に置き換えて読み込みます。その他の場合は、入力で変換式を使って MeV に直してください。例えば、e0 = 8.180425e-8/13**2 は、13Åの中性子のエネルギーを与えます。

`e-type = 1` の場合の入力フォーマットは以下のようになります。

```
    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)
```

この場合、エネルギー分布は、

```
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)
```

のように与えられます。よって例えば、0-2MeV 間を 0.2、2-4MeV 間を 0.6、4-6MeV 間を 0.2 としたい場合、入力のフォーマットは

```
    e-type = 1
        ne = 3
           0   0.2
           2   0.6
           4   0.2
           6
```

となります。

5.3.19.2 微分型連続エネルギー分布

表 62: 線源エネルギー分布パラメータ (2)

パラメータ	説明
e-type = 21, (31)	各エネルギービンの下限値 e(i) とそのビンにおける線源粒子の生成確率の微分量 [dφ/dE](i) を与えることにより、連続的なエネルギー分布を指定する。統計的に [dφ/dE](i)*{e(i+1)-e(i)} に応じて各ビンに生成する粒子を調整し、エネルギー分布を表現する。31 の時は、エネルギーを波長 (Å) で与える。
ne =	エネルギー群数。ne が正の場合は、各ビン内の微分フラックスを [1/MeV] の単位で表示させた際に一定となるように粒子を生成し、負の場合は、[1/Lethargy] の単位で一定となるように粒子を生成する。 データは自由フォーマットで次の様に与える。 (e(i),[dφ/dE](i),i=1,|ne|), e(|ne|+1) 各ビンに生成される粒子数の積分値は dφ/dE(i)*{e(i+1)-e(i)} に比例する。
e-type = 24, (34)	e-type = 21, (31) と同じエネルギー分布を発生させる。ただし、e-type = 21, (31) が生成数を調整してエネルギー分布を表現するのに対し、e-type = 24, (34) では、全てのエネルギービンに同数の粒子を生成させ、粒子のウエイトの積分値を w(i)*{e(i+1)-e(i)} に比例して変化させることによりエネルギー分布を表現する。また、p-type = 1 とし生成個数比 p(i) を与えることにより、各ビンの生成個数を変化させ、特定のエネルギーをもつ線源の統計量を変化させることができる。34 の時は、エネルギーを波長 (Å) で与える。
ne =	エネルギー群数。ne が正の場合は、各ビン内の微分フラックスを [1/MeV] の単位で表示させた際に一定となるように粒子を生成し、負の場合は、[1/Lethargy] の単位で一定となるように粒子を生成する。 データは自由フォーマットで次の様に与える。 (e(i),w(i),i=1,|ne|), e(|ne|+1) デフォルト（p-type = 0）では各ビンに等しい個数が生成される。 p-type = 1 で p(i) を設定した場合は、それらの値に各ビンの生成粒子数の積分値は比例する。
p-type = 0, 1	(D=0) 生成個数のオプション。 for 0, 全てのビンで p(i)=1、以下のデータは無し。 for 1, 各ビンの生成個数比 p(i) を次の行からデータで与える。 (p(i),i=1,ne)

5.3.19.3 離散的エネルギー分布

表 63: 線源エネルギー分布パラメータ (3)

パラメータ	説明
e-type = 8, (18)	各エネルギー点 e(i) とその点の線源粒子の生成確率 w(i) を与えることにより、離散的なエネルギー分布を指定する。統計的に w(i) に比例するように各点に粒子を生成させ、離散的なエネルギー分布を表現する。放射性元素の崩壊により生成される γ 線のようにエネルギー幅を持たない線源を定義するために使います。18 の時は、エネルギーを波長 (Å) で与える。
ne =	エネルギー点数。データは自由フォーマットで次の様に与える。 (e(i),w(i),i=1,ne) 各点に生成される粒子数は w(i) に比例。
e-type = 9, (19)	e-type = 8, (18) と同じエネルギー分布を発生させる。ただし、e-type = 8, (18) が生成数を調整してエネルギー分布を表現するのに対し、e-type = 9, (19) では、全てのエネルギー点に同数の粒子を生成させ、粒子のウエイトを w(i) に比例して変化させることにより不連続なエネルギー分布を表現する。また、p-type = 1 とし生成個数比 p(i) を与えることにより、各点の生成個数を変化させ、特定のエネルギーをもつ線源の統計量を変化させることができる。19 の時は、エネルギーを波長 (Å) で与える。
ne =	エネルギー点数。データは自由フォーマットで次の様に与える。 (e(i),w(i),i=1,ne) デフォルト (p-type=0) で各点に等しい個数が生成される。 各点に生成される粒子数は p(i) に比例。
p-type = 0, 1	(D=0) 生成個数のオプション。 for 0, 全ての点で p(i)=1、以下のデータは無し。 for 1, 各点の生成個数比 p(i) を次の行からデータで与える。 (p(i),i=1,ne)

5.3.19.4 離散的+積分型連続エネルギー分布

表 64: 線源エネルギー分布パラメータ (4)

パラメータ	説明
e-type = 22, (32)	各エネルギービンの下限値 emin(i) と上限値 emax(i) 及びそのビンにおける線源粒子の生成確率の積分値 w(i) を与えることにより、任意のエネルギー分布を指定する。e-type = 8,9,18,19 のように離散的なエネルギー分布を表現するときは、エネルギービンの下限値と上限値を同じ値にする。統計的に w(i) に比例するように各ビンに生成する粒子の数を調整し、エネルギー分布を表現する。32 の時は、エネルギーを波長 (Å) で与える。
ne =	エネルギー群数。ne が正の場合は、各ビン内の微分フラックスを [1/MeV] の単位で表示させた際に一定となるように粒子を生成し、負の場合は、[1/Lethargy] の単位で一定となるように粒子を生成する。 データは自由フォーマットで次の様に与える。 (emin(i),emax(i),w(i),i=1,|ne|) 各ビンに生成される粒子数の積分値は w(i) に比例する。
e-type = 23, (33)	e-type = 22, (32) と同じエネルギー分布を発生させる。ただし、e-type = 22, (32) が生成数を調整してエネルギー分布を表現するのに対し、e-type = 23, (33) では、全てのエネルギービンに同数の粒子を生成させ、粒子のウエイトの積分値を w(i) に比例して変化させることによりエネルギー分布を表現する。また、p-type = 1 とし生成個数比 p(i) を与えることにより、各ビンの生成個数を変化させ、特定のエネルギーをもつソースの統計量を変化させることができる。33 の時は、エネルギーを波長 (Å) で与える。
ne =	エネルギー群数。ne が正の場合は、各ビン内の微分フラックスを [1/MeV] の単位で表示させた際に一定となるように粒子を生成し、負の場合は、[1/Lethargy] の単位で一定となるように粒子を生成する。 データは自由フォーマットで次の様に与える。 (emin(i),emax(i),w(i),i=1,|ne|) デフォルト（p-type = 0）では各ビンに等しい個数が生成される。 p-type = 1 で p(i) を設定した場合は、それらの値に各ビンの生成粒子数の積分値は比例する。
p-type = 0, 1	(D=0) 生成個数のオプション。 for 0, 全てのビンで p(i)=1、以下のデータは無し。 for 1, 各ビンの生成個数比 p(i) を次の行からデータで与える。 (p(i),i=1,ne)

中性子光学のために、エネルギー分布を波長 (Å) で指定できるオプションを設けました。e-type=22, 23 の代わりに e-type=32, 33 を指定すると、エネルギーの入力を全て波長に置き換えて読み込みます。その他の場合は、入力で変換式を使って MeV に直してください。例えば、e0 = 8.180425e-8/13**2 は、13Åの中性子のエネルギーを与えます。

e-type = 22 の場合の入力フォーマットは以下のようになります。

```
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)
```

よって例えば、0-2MeV 間を 0.2、2-4MeV 間を 0.6、4-6MeV 間を 0.2、及び 5.6MeV 単色を 0.4 としたい場合、入力のフォーマットは

```
e-type = 22
    ne = 4
       0   2   0.2
       2   4   0.6
       4   6   0.2
       5.6 5.6 0.4
```

となります。

5.3.19.5 微分型ガウス分布・マクスウェル分布

表 65: 線源エネルギー分布パラメータ (5)

パラメータ	説明
`e-type = 2, (12)`	微分線源スペクトル (dφ/dE) をガウス分布で与える。12 の時は、エネルギーを波長 (Å) で与える。

`eg0 =`
`eg1 =`
`eg2 =`
`eg3 =`
	ガウス分布の中心値 [MeV]。
	ガウス分布の半値全幅 [MeV]。
	ガウス分布のカットオフ最小値 [MeV]。
	ガウス分布のカットオフ最大値 [MeV]。

`e-type = 3`	微分線源スペクトル (dφ/dE) を Maxwellian 分布 $f(E) = E^a \exp(-E/T)$ で与える。
`nm =`	エネルギー群数。`nm` を正の数で与えた時は線形で等分点を決定し、負の数で与えた時は対数で等分点を決定。この数値が少ない場合は分布が滑らかにならないので、100 以上（対数の場合は −100 以下）を奨励します。

`et0 =`
`et1 =`
`et2 =`
`et3 =`
	温度パラメーター T[MeV]。
	Maxwellian 分布のカットオフ最小値 [MeV]。
	Maxwellian 分布のカットオフ最大値 [MeV]。
	`(D=0.5)` エネルギーのベキ (上式のパラメータ a)。通常の Maxwellian 分布では 1。

`e-type = 7`	`e-type = 3` と同じエネルギー分布を発生させる。ただし、`e-type = 3` が生成数を調整してエネルギー分布を表現するのに対し、`e-type = 7` では、全てのエネルギービンに同数の粒子を生成させ、粒子のウエイトの積分値を Maxwellian 分布 $f(E) = E^a \exp(-E/T)$ に応じて変化させることによりエネルギー分布を表現する。また、`p-type = 1` とし生成個数比 `p(i)` を与えることにより、各ビンの生成個数を変化させ、特定のエネルギーをもつ線源の統計量を変化させることができる。
`nm =`	エネルギー群数。`nm` を正の数で与えた時は線形で等分点を決定し、負の数で与えた時は対数で等分点を決定。この数値が少ない場合は分布が滑らかにならないので、100 以上（対数の場合は −100 以下）を奨励します。 デフォルト（`p-type = 0`）では各ビンに等しい個数が生成される。 `p-type = 1` で `p(i)` を設定した場合は、それらの値に各ビンの生成粒子数の積分値は比例する。

`et0 =`
`et1 =`
`et2 =`
`et3 =`
	温度パラメーター T[MeV]。
	Maxwellian 分布のカットオフ最小値 [MeV]。
	Maxwellian 分布のカットオフ最大値 [MeV]。
	`(D=0.5)` エネルギーのベキ (上式のパラメータ a)。通常の Maxwellian 分布では 1。

`p-type = 0, 1`	(D=0) 生成個数のオプション。 for 0, 全てのビンで `p(i)=1`、以下のデータは無し。 for 1, 各ビンの生成個数比 `p(i)` を次の行からデータで与える。`(p(i),i=1,nm)`

5.3.19.6 微分型任意関数分布

表 66: 線源エネルギー分布パラメータ (6)

パラメータ	説明
e-type = 5, (15) f(x) =	微分線源スペクトル (dφ/dE) を任意の関数 f(x) で与える。15 の時は、エネルギーを波長 (Å) で与える。 Fortran 形式で書いた関数。x はエネルギー [MeV/n] を表す。 内部変数や定数が使えます。例）f(x) = exp(-c1*x**2)
nm =	(D=−200) エネルギー群数。nm を正の数で与えた時は線形で等分点を決定し、負の数で与えた時は対数で等分点を決定。
eg1 = eg2 =	エネルギー分布のカットオフ最小値 [MeV]。 エネルギー分布のカットオフ最大値 [MeV]。
e-type = 6, (16) f(x) =	e-type = 5, (15) と同じエネルギー分布を発生させる。ただし、e-type = 5, (15) が生成数を調整してエネルギー分布を表現するのに対し、e-type = 6, (16) では、全てのエネルギービンに同数の粒子を生成させ、粒子のウエイトの積分値が任意の関数 $f(x)$ に応じて変化させることによりエネルギー分布を表現する。また、p-type = 1 とし生成個数比 p(i) を与えることにより、各ビンの生成個数を変化させ、特定のエネルギーをもつ線源の統計量を変化させることができる。16 の時は、エネルギーを波長 (Å) で与える。 Fortran 形式で書いた関数。x はエネルギー [MeV/n] を表す。 内部変数や定数が使えます。例）f(x) = exp(-c1*x**2)
nm =	(D=−200) エネルギー群数。nm を正の数で与えた時は線形で等分点を決定し、負の数で与えた時は対数で等分点を決定。 デフォルト（p-type = 0）では各ビンに等しい個数が生成される。p-type = 1 で p(i) を設定した場合は、それらの値に各ビンの生成粒子数の積分値は比例する。
eg1 = eg2 =	エネルギー分布のカットオフ最小値 [MeV]。 エネルギー分布のカットオフ最大値 [MeV]。
p-type = 0, 1	(D=0) 生成個数のオプション。 for 0, 全てのビンで p(i)=1、以下のデータは無し。 for 1, 各ビンの生成個数比 p(i) を次の行からデータで与える。 (p(i),i=1,nm)

5.3.19.7 RI 線源によるエネルギー分布

表 67: 線源エネルギー分布パラメータ (7)

パラメータ	説明
e-type= 28,29	放射性核種とその放射能（単位 Bq) を与えることにより、その放射性核種の崩壊に伴って放出される α 線、β 線（オージェ電子含む）、または γ 線を線源とした計算が実行できます。放出される粒子のスペクトル計算には、放射性核種崩壊データベース DECDC[59]が使用されます。なお、このデータベースは ICRP-07 と同等です。 この機能を使う場合は、崩壊データファイル RIsource.dat が置かれたフォルダを [parameters] セクションにおいて file(24) (D=c:/phits/data/) により指定する必要があります。 e-type= 28 の場合、崩壊データベースにより計算された放出強度にしたがって、各エネルギーの粒子を生成します。 e-type= 29 の場合、崩壊データベースにより計算された全てのエネルギー点に同数の粒子を生成させ、その放出強度にしたがって、各エネルギーのウエイトを変化させます。
ni=	放射性核種の数。放射性核種とその放射能は次の様なフォーマットで指定する。 (RI(i),A(i),i=1,ni) ただし、核種 RI(i) は 137Cs あるいは Cs-137 の書式で指定する。放射能 A(i) は Bq（ベクレル）単位で指定する。
dtime=	(D=−10.0) 時間経過の指定オプション。 dtime>0（単位は sec）: 指定した時間が経過した時点の指定核種及びその娘核種の放射能による α 線,β 線（オージェ電子含む）, γ 線を設定します。各 RI の放射能は、dtime 経過後の値に減衰します。例えば、半減期 1 分の RI を 100Bq で指定し、dtime=60.0 と設定した場合、その RI の放射能は 50Bq と見なされます。 dtime=0: 指定した放射能（時間経過なし）による崩壊ガンマ線を設定します。 dtime<0: 指定した放射性核種の半減期 ×|dtime| が経過した時点の指定核種及びその娘核種の放射能による α 線, β 線（オージェ電子含む）, γ 線を設定します。ただし、dtime>0 の場合と異なり、放射能の絶対値は減衰しません。例えば、半減期 1 分の RI を 100Bq で指定し、dtime=−1.0 と設定した場合、その RI の放射能は 100Bq のまま娘核種の崩壊を考慮可能です。放射平衡まで達する時間が分からない場合は、dtime=−10.0 のように長めの値を設定しておくと便利です。dtime<0 とする場合、親核と娘核（例えば Cs-137 と Ba-137m）を同時に指定することはできません。

マルチソースでこの機能を使用する際は、<source>=1.0 とし totfact を以下のように指定してください。

- マルチソースの全てで e-type = 28,29 を使用する場合は、<source>の個数を入力する。
- マルチソースで e-type = 28,29 以外を含む場合は、各<source>が絶対強度となるように設定し、すべての<source>の和を入力する。

なお、上記の方法で決めた値に − をつけ、負の値として totfact を指定した場合は、各線源の粒子を同確率で発生させ、与えられた放射能（Bq 数）の比に応じて各粒子のウエイトを変更します。放射能が大きく違う複数の線源を定義する場合は、totfact を負で与えることにより、Bq 数の大小にかかわらず統計量を増やすことができます。

[59] A. Endo, Y. Yamaguchi and K.F. Eckerman, Nuclear decay data for dosimetry calculation - Revised data of ICRP Publication 38, JAERI 1347 (2005).

表 68: 線源エネルギー分布パラメータ (8)

パラメータ	説明
	（e-type= 28,29 の場合の続き）
actlow=	(D=1.0e-10) 放射能の下限の指定オプション (Bq)。指定した時間経過後の放射能がこの値より小さい場合、その放射性核種の α 線, β 線（Auger 電子、内部転換電子含む）, γ 線（特性 X 線含む）は線源に設定されません。
norm=	(D=0) 線源強度の規格化オプション。 0: 1 秒あたり (/sec) の量を出力するように規格化します。その際、線源強度は、Bq 単位で指定した放射能強度（dtime>0 の場合はその崩壊も考慮した後）とします。 1: 崩壊に伴って放出される α 線, β 線（Auger 電子含む）, γ 線 1 つあたりの量を出力するように規格化します。（通常の PHITS の規格化方法に準拠）
iaugers=	(D=0) Auger 電子（内部転換電子含む）の取扱オプション（proj=electron のみ有効）。 0: Auger 電子を含む（β 線 + Auger 電子 ）。 1: Auger 電子を含まない（β 線のみ）。 2: Auger 電子のみを取り扱う。
iannih=	(D=0) 電子対消滅により生成される光子（消滅光子）の取扱オプション（proj=photon のみ有効）。 0: 消滅光子を含む。 1: 消滅光子を含まない。別のマルチソースで proj = positron として陽電子の放出を明示的に考慮する場合は，消滅光子の 2 重カウントを防ぐため，こちらのオプションを選択してください。
icharctx=	(D=0) 特性 X 線の取扱オプション (proj=photon のみ有効)。 0: 特性 X 線を含む（γ 線 + 特性 X 線）。 1: 特性 X 線を含まない（γ 線のみ）。 2: 特性 X 線のみ取り扱う。

e-type = 28,29 の場合の入力例は以下のようになります。

```
[ S o u r c e ]
 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
```

ここでは、Cs-137 と Cs-134 が共に 100Bq ずつ存在し、それらが放射平衡になった状況を線源として指定しています。また、actlow を 1(Bq) することで、この数値以下の放射能の影響を無視しています。

また、例えば、Cs-137 の崩壊に伴って放出される photon と electron を同時に考慮したい場合は、photon と electron の 2 つの<source>セクションをもつマルチソースを定義します。両者の崩壊数は同じとなるため、ソースの相対比は 1.0 と指定します (<source>=1.0)。その際、マルチソース全体の規格化定数 totfact は、各<source>の和となるため、2.0 と指定します。

5.3.19.8 タリー結果を用いたエネルギー分布

表 69: 線源エネルギー分布パラメータ (9)

パラメータ	説明
e-type= 20	file=によってファイル名を指定することにより、タリーで求めたエネルギー分布を線源の分布として設定できます。使用できるのは、[t-track], [t-cross], [t-point], [t-product], [t-time], [t-interact] において、axis=eng として求めた結果のみです。file=には、各タリーで出力ファイルとして指定したファイル名 (***.out) を書いてください。各タリーで得られる分布は unit=の値によりエネルギーの微分量か積分量の 2 つの形式に分かれますが、線源として使用する際も各形式に沿ってエネルギー分布を表現します。ただし、時間 (nsec) や角度 (sr) に関する微分は考慮しません。また、分布の絶対値は、タリー結果を積分した値で規格化します。
file=	タリーの出力ファイル名。

線源として使用する分布は、タリー結果を得る際に part=で指定した最初の粒子の結果です。ただし、その粒子が線源粒子として発生するわけではないのでご注意ください。線源として発生させる粒子の種類は proj=で指定します。

入力例は以下の通りです。e-type=20 と file=を指定すれば本機能を使用できます。

```
[ S o u r c e ]
   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 宇宙線フラックスのエネルギー・角度分布

表 70: 線源エネルギー分布パラメータ (10)

パラメータ	説明
e-type= 25,26	宇宙空間及び大気圏内における宇宙線フラックスのエネルギー及び角度分布を絶対値も含めて再現する線源です。この機能を使うには、file(24) で指定したフォルダ（デフォルト file(1)/data/）に cosmicray フォルダ（及びその中に含まれる入力データファイル）がある必要があります。 e-type=25 の場合、宇宙線モデルで計算したフラックスに従って、各エネルギーの粒子を生成します。 e-type=26 の場合、各エネルギー群に同数の粒子を発生させ、宇宙線モデルで計算したフラックスに従って、各粒子のウェイトを変化させます。 e-type=25,26 の場合、各マルチソースの<source>は 1.0 とする必要があります。また、計算する宇宙線フラックスの単位は (/cm^2/sec) であり、totfact に線源面積（例えば、s-type=9 の場合は $\pi(r2)^2$）を設定することにより、タリー結果を自動的に (/sec) に規格化することができます（ただし、銀河宇宙線の場合のみ）。totfact を負値で指定した場合は、各マルチソースから同数の粒子を発生させますので、フラックスの大小にかかわらず統計量を増やすことができます。なお、e-type=25,26 を使う場合は、全てのマルチソースで同じ e-type 及び icenv を指定してください。
icenv=	(D=0) 宇宙線環境パラメータ。 1–5: PARMA/EXPACS[60]で計算した大気圏内における銀河宇宙線 (Galactic Cosmic-Ray, GCR) モード。それぞれ、理想大気中 (=1)、地表面 (=2)、航空機コックピット (=3)、航空機客室 (=4)、ブラックホールモード (=5) を表します。icenv=2–4 の場合は、周辺環境パラメータ environ を定義する必要があります。ブラックホールモードは、地面からの反射中性子を全てカットするので、大地を含んだシミュレーションに最適です。なお、周辺環境が影響するのは中性子のみで、それ以外の粒子に対しては icenv=1–5 は同じ結果を与えます。また、s-type=9 かつ dir=iso とした場合、大気圏内の複雑な宇宙線角度分布も自動で再現します（z 軸+方向が天頂方向)。粒子種類 (proj) は、中性子、陽子、重イオン ($Z \leq 28$)、μ 粒子、光子、電子、陽電子が指定可能です。 0: DLR モデル[61]で計算した宇宙空間における銀河宇宙線 (GCR) モード。粒子種類 (proj) は、陽子及び重イオン ($Z \leq 28$) が指定可能です。 −1: Tylka モデル[62]で計算した宇宙空間における太陽高エネルギー粒子 (Solar Energetic Particle, SEP) モード。environ パラメータを指定することにより、4 種類の太陽高エネルギー粒子イベント (Solar Particle Event, SPE) から再現するフルエンスを選ぶことができます。粒子種類 (proj) は陽子のみが指定可能です。このモードを選択して線源面積を totfact に指定した場合、タリー結果は (/sec) ではなく各イベントの積算値 (/SPE) に規格化されます。

[60] 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

[61] Matthia D, Berger T, Mrigakshi AI, Reitz G, A ready-to-use galactic cosmic ray model. Adv Space Res 51: 329-338 (2013).

[62] 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).

表 71: 線源エネルギー分布パラメータ (11)

パラメータ	説明
	（e-type= 25,26 の場合の続き）
nm=	(D=−200) エネルギー群数。nm を正の数で与えた時は線形で等分点を決定し、負の数で与えた時は対数で等分点を決定。
eg1= eg2=	エネルギー分布のカットオフ最小値 [MeV/n]。デフォルト値は、中性子は 1.0e-8、その他の粒子は大気圏内モード (icenv>0) の場合は 1.0e-2、宇宙空間モード (icenv≤0) の場合は 1.0 となります。 エネルギー分布のカットオフ最大値 [MeV/n]。デフォルト値は、中性子、光子、電子、陽電子は 1.0e4、μ 粒子は 1.0e8、SEP 陽子は 1.0e5、その他の粒子は 1.0e6 となります。
solarmod=	(D=0) 銀河宇宙線フラックスを計算するために必要となる太陽活動度 (W-index)。icenv≥0 のみ有効。太陽活動が活発になると、太陽風の影響により銀河宇宙線のフラックスは低下します。太陽極小及び極大期におけるおおよその値はそれぞれ 0 と 150 程度ですが、これよりも低くなる場合や高くなる場合もあります。このパラメータを指定しない場合は、icyear, icmonth, icday で指定した年月日の W-index をデータベースから参照し、デフォルトの日付から推定した W-index は 0.1 です。そのデータベースには、1951 年以降は中性子モニタの計数率[63]から推定した各日の W-index が、それ以前は Usoskin ら[64]が太陽黒点数より再構築した太陽活動度より推定した各月の W-index が格納されています。なお、データベースのない年月日（1611 年以前、1647–1699 年、及び直近の数か月）を指定した場合、「Specified date is wrong (too past or too recent)」というエラーが出ます。また、直近の W-index は WASAVIES ホームページ[65]よりダウンロード可能です。
icyear=	(D=2009) 太陽活動度を計算するための年。icenv≥0 かつ solarmod を指定しない場合のみ有効となります。
icmonth=	(D=10) 太陽活動度を計算するための月。icenv≥0 かつ solarmod を指定しない場合のみ有効となります。
icday=	(D=20) 太陽活動度を計算するための日。icenv≥0 かつ solarmod を指定しない場合のみ有効となります。
depatom=	大気深度 (g/cm^2)。大気圏内モード (icenv>0) のみ有効。このパラメータを指定しない場合は、alti で定義した高度 (km) より US Standard Atmosphere 1986[66]を使って大気深度を計算します。デフォルトの高度 (=0 km) から計算した大気深度は約 1033 g/cm^2 です。
alti=	(D=0) 大気深度を計算するための高度 (km)。icenv>0 かつ depatom を指定しない場合のみ有効。

[63] NMDB: Real-Time Database for high-resolution Neutron Monitor measurements, http://www01.nmdb.eu/

[64] 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).

[65] Warning System for Aviation Exposure to Solar Energetic Particle, NICT, https://wasavies.nict.go.jp/FFPday.txt

[66] U.S. Standard Atmosphere, 1976, U.S. Government Printing Office, Washington, D.C., 1976.

表 72: 線源エネルギー分布パラメータ (12)

パラメータ	説明
	（e-type= 25,26 の場合の続き）
rigid=	宇宙線のカットオフリジディティ(GV)。カットオフリジディティとは、磁気圏を透過可能な宇宙線の最低リジディティ（磁場による曲がりにくさ）を表しており、極域で低く赤道付近で高くなります（おおよそ 0–18 GV 程度）。大気圏内モード (icenv>0) の場合は、地球に垂直に入射した場合のカットオフリジディティ(vertical cut-off rigidity) を表し、このパラメータを指定しない場合、glat 及び glong で指定した緯度・経度より MAGNETOCOSMICS[67]で構築したデータベースを使って自動的に決定します。デフォルトの glat 及び glong から推定したカットオフリジディティは 0 GV です。宇宙空間モード (icenv≤0) の場合は、rigid パラメータで指定した値以下のリジディティを持つ宇宙線を全てカットし、そのデフォルト値は 0 GV です。
glat=	(D=90) カットオフリジディティを計算するための緯度 (degree)。正値が北緯、負値が南緯を表し、icenv>0 かつ rigid を指定しない場合のみ有効となります。
glong=	カットオフリジディティを計算するための経度 (degree)。正値が東経、負値が西経を表し、icenv>0 かつ rigid を指定しない場合のみ有効となります。
environ=	大気圏内の中性子フラックスに関する周辺環境パラメータ (icenv=2–4) もしくは SEP イベント ID(icenv=−1)。icenv=2 の場合は地中の水分含有率、icenv=3 及び 4 の場合は航空機の重量（100 ton）を表し、そのデフォルト値は 0.15 となります。ただし，どちらの量も，理想的な環境における中性子フラックスからの「ゆがみ」の大きさを表すためのパラメータであり，純粋な物理量ではなく，条件により調整が必要となることに注意する必要があります[68]。icenv=−1 の場合は再現する SPE の種類を 1–4 の整数で表し、それぞれ、Feb. 1956 （=1、デフォルト）、Nov. 1960 (=2)、Aug. 1972 (=3)、Oct. 1989 (=4) に発生した SPE に対応しています。

[67] http://cosray.unibe.ch/~laurent/magnetocosmics/
[68] T. Sato and K. Niita, Analytical Functions to Predict Cosmic-Ray Neutron Spectra in the Atmosphere, Radiat. Res. 166, 544-555 (2006).

5.3.20 角分布の定義

`dir = data` を指定した場合は、次のような `a-type =` で始まる角分布サブセクションが必要です。(D=***) のあるものは、省略可能です。

表 73: 角分布パラメータ (1)

パラメータ	説明
`a-type = 1, (11)`	角度分点 `a(i)` と各ビンにおける線源粒子の生成確率 `w(i)` を与えることにより、任意の角度分布を指定する。 分点は、1 の時は cos で、11 の時は degree で与える。 各ビンに生成される積分粒子数は、`w(i)` に比例する。
`na =`	角度群数。データは自由フォーマットで次の様に与える。 `(a(i),w(i),i=1,|na|), a(|na|+1)`
`a-type = 4, (14)`	`a-type = 1, (11)` と同じ角度分布を発生させる。ただし、`a-type = 1, (11)` が生成数を調整して角度分布を表現するのに対し、`a-type = 4, (14)` では、全ての角度ビンに同数の粒子を生成させ、粒子のウエイトの積分値が `w(i)` となるように角度分布を表現する。 分点は、4 の時は cos で、14 の時は degree で与える。 各ビンに生成される積分粒子数は、`q(i)` に比例する。
`na =`	角度群数。データは自由フォーマットで次の様に与える。 `(a(i),w(i),i=1,|na|), a(|na|+1)` デフォルト（`q-type = 0`）では各ビンに等しい個数が生成される。 `q-type = 1` で `q(i)` を設定した場合は、それらの値に各ビンの生成粒子数の積分値は比例する。
`q-type = 0, 1`	(D=0) 生成個数のオプション。 for 0, 全てのビンで `q(i)=1`。以下のデータは無し。 for 1, 各ビンの生成個数比 `q(i)` を次の行からデータで与える。 `(q(i),i=1,na)`

表 74: 角度分布パラメータ (2)

パラメータ	説明
a-type = 5, (15)	角度分布 ($d\varphi/d\Omega$) を任意の関数 g(x) で与える。
	分点は、5 の時は cos で与え、15 の時は degree で与える。
g(x) =	Fortran 形式で書いた関数。x は角度を表す。
	内部変数や定数が使えます。例）g(x) = exp(-c1*x**2)
nn =	角度群数。
ag1 =	角度分布のカットオフ最小値。
ag2 =	角度分布のカットオフ最大値。
a-type = 6, (16)	a-type = 5, (15) と同じ角度分布を発生させる。ただし、a-type = 5, (15) が生成数を調整して角度分布を表現するのに対し、a-type = 6, (16) では、全ての角度ビンに同数の粒子を生成させ、粒子のウエイトを任意の関数 $g(x)$ に比例して変化させることにより角度分布を表現する。
	分点は、6 の時は cos で与え、16 の時は degree で与える。
g(x) =	Fortran 形式で書いた関数。x は角度を表す。
	内部変数や定数が使えます。例）g(x) = exp(-c1*x**2)
nn =	角度群数。 デフォルト（q-type = 0）では各ビンに等しい個数が生成される。q-type = 1 で q(i) を設定した場合は、それらの値に各ビンの生成粒子数の積分値は比例する。
ag1 =	角度分布のカットオフ最小値。
ag2 =	角度分布のカットオフ最大値。
q-type = 0, 1	(D=0) 生成個数のオプション。 for 0, 全てのビンで a(i)=1。以下のデータは無し。 for 1, 各ビンの生成個数比 a(i) を次の行からデータで与える。 (q(i),i=1,nm)

5.3.21 時間分布の定義

t-type =で始まる時間分布サブセクションを指定することにより、線源に時間分布をもたせることができます。t-type =1,2 はそれぞれ矩形分布と Gauss 分布、t-type =3,4 は時間分布を分点と強度分布で与えます。t-type =5,6 は時間に関する強度分布を任意の関数で与えます。また、t-type =100 では、エネルギーの関数として時間分布の型を指定できます。(D=***) のあるものは、省略可能です。

表 75: 時間分布パラメータ (1)

パラメータ	説明
t-type = 0, 1, 2	(D=0) 時間分布。
	0: 時間分布なし。t=0.0。
	1: 矩形分布。
	2: Gauss 分布。
t0 =	(D=0.0) 第 1 の時間分布の中心時刻 [ns]。
tw =	時間分布の幅 [ns]。矩形分布の場合は全幅。Gauss 分布の場合は半値全幅。
tn =	時間分布の数。
td =	時間分布の間隔 [ns]。
tc =	(D=10×tw) Gauss 分布の場合のカットオフ時間 [ns]。

表 76: 時間分布パラメータ (2)

パラメータ	説明
t-type = 3	時間分点 t(i) と各ビンにおける線源粒子の生成確率の積分値 w(i) を与えることにより、任意の時間分を指定する。統計的に w(i) に比例するように各ビンに生成する粒子の数を調整し、時間分布を表現する。
ntt =	時間群数。データは自由フォーマットで次のように与える。 (t(i),w(i),i=1,ntt), t(ntt+1) 各ビンに生成される粒子数の積分値は w(i) に比例する。
t-type = 4	t-type=3 と同じ時間分布を発生させる。ただし、t-type=3 が生成数を調整して時間分布を表現するのに対し、t-type=4 では、全ての時間ビンに同数の粒子を生成させ、粒子のウエイトの積分値を w(i) に比例して変化させることにより時間分布を表現する。また、o-type=1 とし生成個数比 o(i) を与えることにより、各ビンの生成個数を変化させ、特定の時刻をもつ線源の統計量を変化させることが出来る。
ntt =	時間群数。データは自由フォーマットで次のように与える。 (t(i),w(i),i=1,ntt), t(ntt+1) デフォルト (o-type=0) では各ビンに等しい個数が生成される。 o-type=1 で o(i) を設定した場合は、それらの値に各ビンの生成粒子数の積分値は比例する。
o-type = 0, 1	(D=0) 生成個数のオプション。 for 0, 全てのビンで o(i)=1。以下のデータは無し。 for 1, 各ビンの生成個数比 o(i) を次の行からデータで与える。 (o(i),i=1,ntt)

表 77: 時間分布パラメータ (3)

パラメータ	説明
`t-type = 5` $h(x)$	微分時間スペクトル ($d\varphi/dt$) を任意の関数 $h(t)$ で与える。 Fortran 形式で書いた関数。x は時刻 [ns] を表す。内部変数や定数が使える。
`ll`	時刻群数。
`tg1` `tg2`	時間分布のカットオフ最小値 [ns]。 時間分布のカットオフ最大値 [ns]。
`t-type = 6` $h(x)$	`t-type=5` と同じ時間分布を発生させる。ただし、`t-type=5` が生成数を調整して時間分布を表現するのに対し、`t-type=6` では、全ての時間ビンに同数の粒子を生成させ、粒子のウエイトの積分値を `w(i)` に比例して変化させることにより時間分布を表現する。また、`o-type=1` とし生成個数比 `o(i)` を与えることにより、各ビンの生成個数を変化させ、特定の時刻をもつ線源の統計量を変化させることが出来る。 Fortran 形式で書いた関数。x は時刻 [ns] を表す。内部変数や定数が使える。
`ll`	時刻群数。 デフォルト (`o-type=0`) では各ビンに等しい個数が生成される。`o-type=1` で `o(i)` を設定した場合は、それらの値に各ビンの生成粒子数の積分値は比例する。
`tg1` `tg2`	時間分布のカットオフ最小値 [ns]。 時間分布のカットオフ最大値 [ns]。
`o-type = 0, 1`	(D=0) 生成個数のオプション。 for 0, 全てのビンで `o(i)=1`。以下のデータは無し。 for 1, 各ビンの生成個数比 `o(i)` を次の行からデータで与える。 `(o(i),i=1,ntt)`

`t-type =100` を指定し、PHIITS のソースファイルを書き換えることで、任意の時間分布を与えることができます。この機能を使用する場合は、ソースファイルのひとつ sors.f ファイルにある subroutine tdis01 を書き換えてください。

表 78: 時間分布パラメータ (4)

パラメータ	説明
`t-type = 100`	エネルギーの関数として時間分布を与える。詳細は、ソースプログラムに書き込んで実現する。
`tg1` `tg2`	時間分布のカットオフ最小値 [ns]。 時間分布のカットオフ最大値 [ns]。

5.3.22 マルチソースの例題

マルチソースの例題を紹介します。この例題には、関数を用いたエネルギー分布や、角度分布の例も含まれています。まず、ソースセクションのリストを示します。

例 4: マルチソースの例題

```
[ S o u r c e ]
 totfact = 3
<source> = 9.72
  s-type = 1
    proj = proton
      z0 = 2
      z1 = 29
      r0 = 5
      r1 = 4
     dir = 0.0
  e-type = 6
     eg1 = 1.e-6
     eg2 = 1.e-3
     nm  = -200
    set: c10[1.e-4]
    f(x) = x**(1.5)*exp(-x/c10)
<source> = 1
  s-type = 1
    proj = photon
      z0 = 1
      z1 = 2
      r0 = 5
     dir = -1
  e-type = 5
     eg1 = 1.e-3
     eg2 = 5.e-1
     nm  = 200
     set: c10[1.e-1]
     set: c20[1.e-1/2.35482]
    f(x) = exp(-(x-c10)**2/2/c20**2)
<source> = 1
  s-type = 1
    proj = neutron
      z0 = 29
      z1 = 30
      r0 = 5
  e-type = 6
     eg1 = 1.e-2
     eg2 = 1.e+3
     nm  = -200
     set: c10[92.469]
     set: c20[5.644e+10]
    f(x) = c10/c20*exp(-sqrt(x*(x+1876))/c10)*(x+938)/sqrt(x*(x+1876))
     dir = data
  a-type = 5
     ag1 = 0
     ag2 = 1
     nn  = 200
    g(x) = exp(-(x-1)**2/0.3**2)
```

この例題には、3 つの<source>で始まるソースが含まれる。最初のソースは、円柱で z が 2cm から 29cm で、半径が 5cm、ただし、r1=4 が定義してあるので、半径 4cm の内側は含まれていない。中空の円柱ソースである。次のソースは、やはり円柱で半径 5cm、z が 1cm から 2cm までの厚さが 1cm の薄い円柱である。最後のソースは、前のと同じ薄い円柱で、z 座標が 29cm から 30cm までである。各ソースの<source>に定義される、各ソースの相対比である。ここでは、各ソースの体積比になっている。従って、このマルチソースでは、定義されるソース領域に均一に粒子が生成される。この座標分布を [t-product] で output=source, icntl=6 で計算したものを以下に示す。このソースは、円柱の表面、厚さ 1cm の領域を定義している。

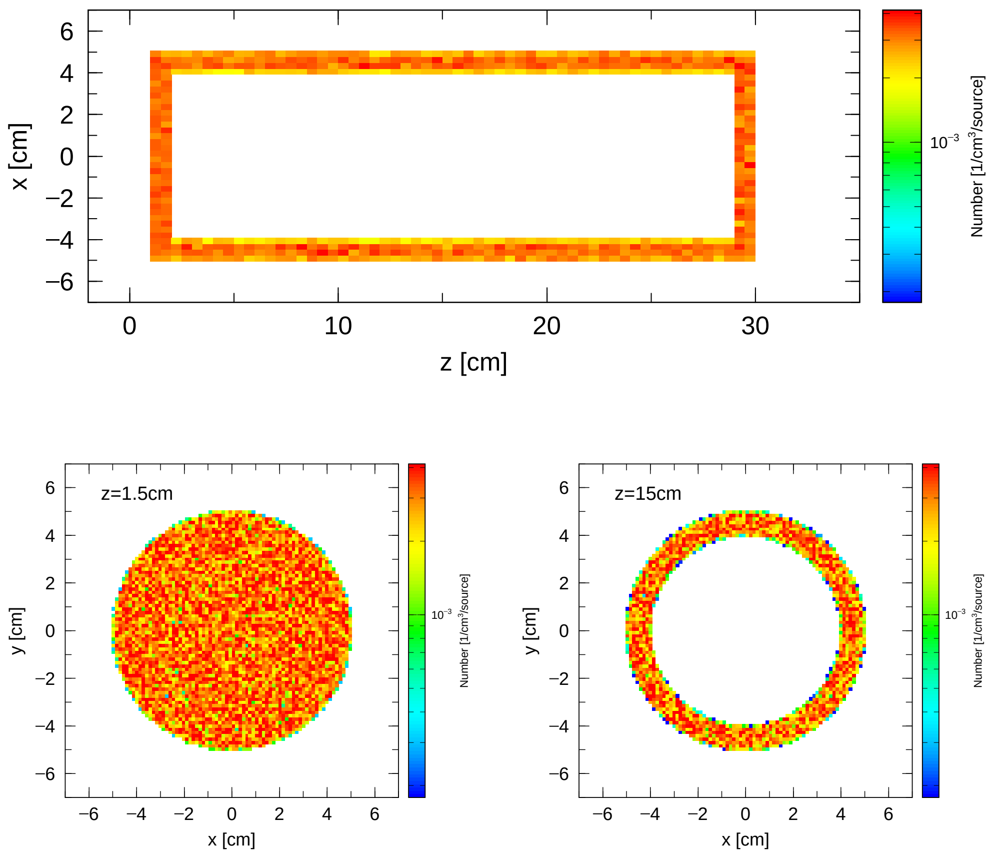


図 13: マルチソース、空間分布

次に、3 つのソースは、ソース粒子が、陽子、光子、そして中性子である。それぞれエネルギー分布が関数で定義してある。最初のものは、Maxwellian 分布、次がガウス分布、最後が任意の関数である。最初の Maxwellian 分布は、e-type=7 の

```
e-type = 7
   et0 = 1.e-4
   et1 = 1.e-6
   et2 = 1.e-3
```

と同義である。また、2 番目のガウス分布は、e-type=2 の

```
e-type = 2
   eg0 = 1.e-1
   eg1 = 1.e-1
   eg2 = 1.e-4
   eg3 = 5.e-1
```

と同義である。

これらのエネルギー分布を、やはり、[t-product] で output=source, icntl=6 で計算したものを以下に示す。結果は、粒子ごとにプロットしているので、各ソースごとのエネルギー分布が色別に表されている。

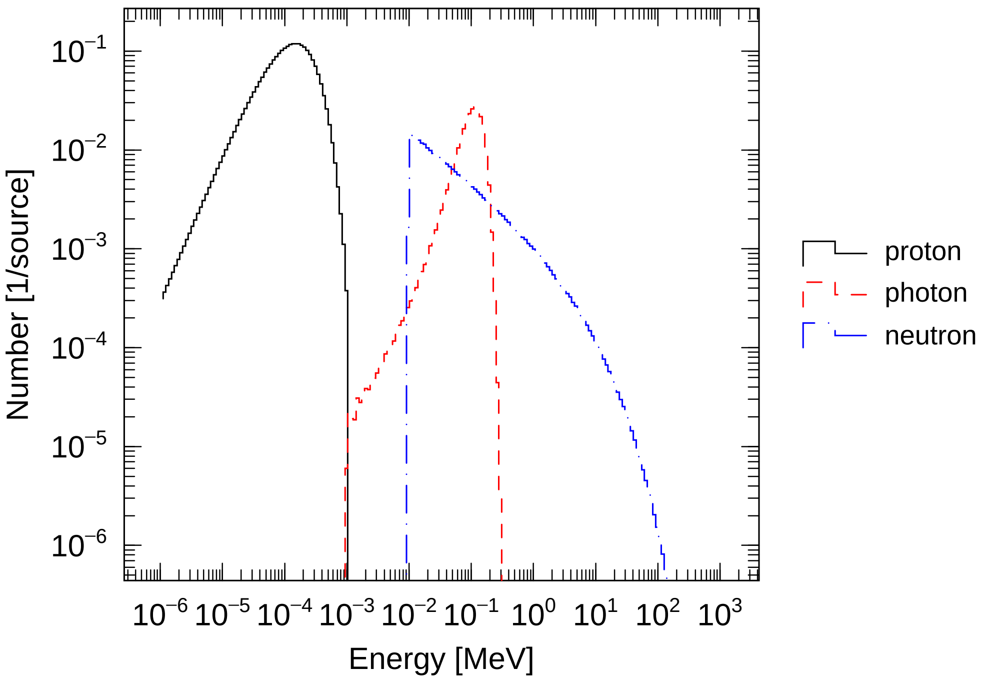


図 14: マルチソース、エネルギー分布

最初のソースは、dir=0, 即ち 90 度方向、2 番目が dir=-1, 180 度方向、3 番目が dir=data で角度分布を持つ。ここでは、0 度を中心としたガウス分布が関数形で定義されている。この結果は、[t-cross] を用いて次のように表される。

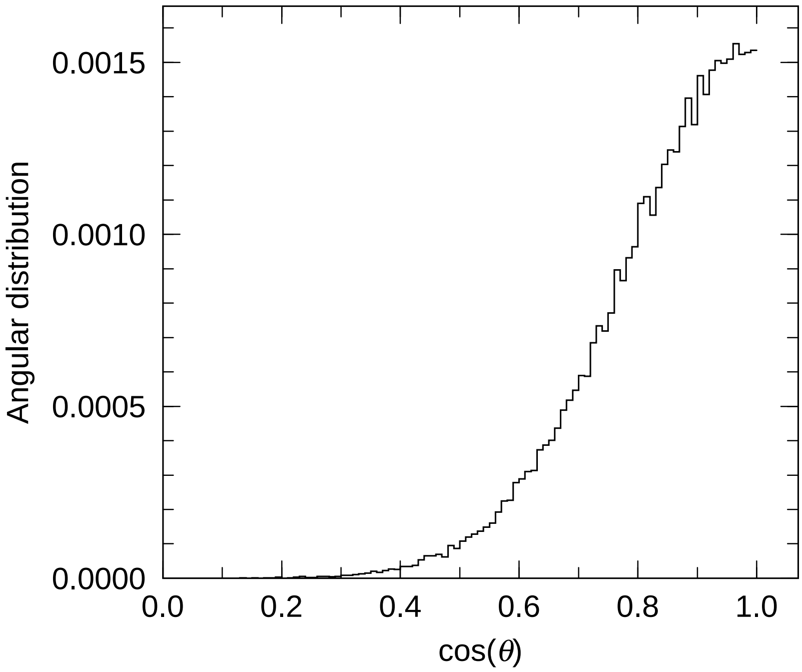


図 15: マルチソース、角度分布

5.3.23 ダクトソースオプション

中性子分光施設の非常に長い中性子導管を扱うために、特別なソースを生成するダクトソースオプションを作りました。等方的なソースを仮定すると、長いビームラインの中のカレントは、立体角の関係からソースの位置からの距離の二乗に反比例します。このことは、ビームラインの壁にぶつかるカレントは、”wall current”と呼びますが、例えば 100m 位置で、1m 位置に比べて 6 桁小さくなります。このような状況での計算の統計を上げるために、粒子のウエイトを変化させて、シミュレーション粒子の wall current が、ビームラインのどの場所でも同じになるようなオプションを設けました。

このダクトソースオプションは、`s-type = 1, 2` の円柱と角柱のソースに `dom = -10` を指定することにより適応されます。各パラメーターとその意味を下にまとめます。

表 79: ダクトソースオプションのパラメータ

パラメータ	説明
`dom = -10`	ダクトソースの指定。
`dl0 =`	ビームラインの始点 z0 からの距離 [cm]。
`dl1 =`	ダクトソースの始点 z0 からの距離 [cm]。
`dl2 =`	ダクトソースの終点 z0 からの距離 [cm]。
`dpf =`	dl2 を通過する粒子の割合。
`drd =`	`s-type = 1` 円柱の時のビームラインの半径 [cm]。
`dxw =`	`s-type = 2` 角柱の時のビームラインの X サイズ [cm]。
`dyw =`	`s-type = 2` 角柱の時のビームラインの Y サイズ [cm]。

ビームラインの形状は、`s-type=1` の時は円柱、`s-type=2` の時は角柱を仮定しています。また、`z1=z0` と `dir=1` も仮定しています。後者は、ビームラインの方向です。これを変更する時は、`trcl` を用いてください。ソース粒子は、`z0` 地点で、`s-type=1` の時は `r0` の内部、もしくは、`s-type=2` の時は、`x0, x1, y0, y1` の内部で生成されます。

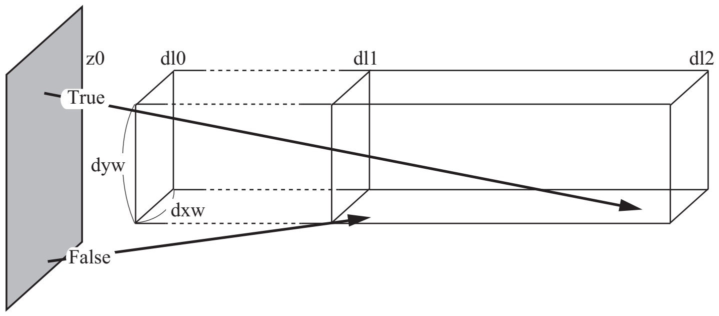


図 16: 角柱のダクトソースのイメージ図

ソース粒子の方向は、ダクトに当たる位置から決定されます。`dl1` から `dl2` の間のダクトに粒子が均一に当たるように壁の位置が決められます。そのとき、等方分布を仮定して粒子のウエイトが決められます。全体の規格化定数は、`z0` 地点で `dl0` 地点のダクト面積と同じ領域で発生した粒子が `dl0` 地点のダクトを通過する粒子数で規格化されます。通常、`dl0` から `dl2` のダクト壁から全てのソース領域が見える時、この規格化定数は 1 です。もし、`z0` 地点でのソース領域が、`dl0` 地点のダクト面積より大きい場合は、`z0` 地点でダクト面積より大きい領域からダクトに入った粒子は、ソース粒子の規格化には含まれません。このことは、`z0` 地点でダクト面積より大きい領域からダクトに入った粒子は、規格化定数を変えずにダクト管内のカレ

ントを増加させます。このダクトソースでは、粒子のソース位置での角分布は、ビームラインの方向のダクト面積の立体角で均一であるということを仮定しています。

次に、簡単な例題を示します。最初の例は、角柱のビームラインを仮定し、ソースサイズとガイド断面積は同じとしています。ダクトソースのセクションは、

例 5: ダクトソースオプションの例題 (1)

```
[ S o u r c e ]

set: c1[200]     $dl0
set: c2[500]     $dl1
set: c3[5000]    $dl2
set: c4[5.0]     $x*2 at z0
set: c5[5.0]     $y*2 at z0
set: c10[5.0]    $dxw
set: c20[5.0]    $dyw
set: c30[0.001]  $dpf

s-type = 2
proj    =  neutron
e0      =  20.0
x0      = -c4/2
x1      =  c4/2
y0      = -c5/2
y1      =  c5/2
z0      =  0.0
z1      =  0.0
dir     =  1.0
phi     =  0.0

dom = -10
dl0 = c1
dl1 = c2
dl2 = c3
dxw = c10
dyw = c20
dpf = c30
```

最初のパートでこのダクトソースに必要な定数、**dl0**, **dl1**, **dl2** と **dxw**, **dyw**, **dpf** を定義しています。次のパートで、ソースの位置、エネルギーを定義しています。最後のパートで、ダクトソースのパラメーターを定義しています。ここでは、5m から 50m のビームラインの中の current と wall current を [t-cross] タリーを用いて計算しています。結果を右の図に示します。同じ図に、理想的な場合の解析的な解、つまり、current が $1/L^2$ に比例し、wall current が $1/L^3$ に比例するグラフも描いてあります。十字の印が、**dl0** の位置です。ここの current が 1 です。ダクトソースオプションの結果は、理想的な場合と完全に一致しています。

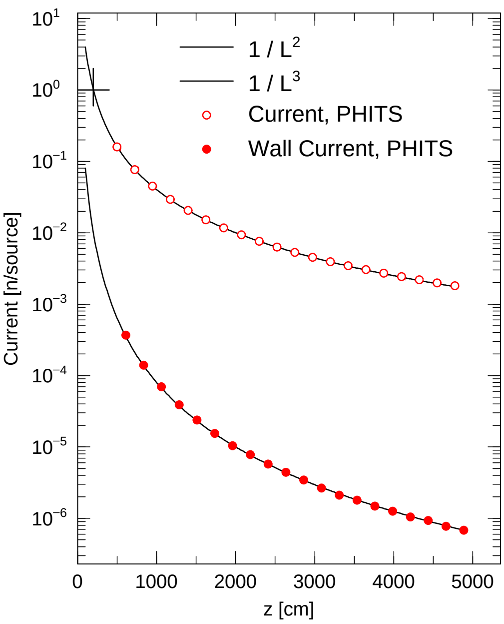


図 17: ダクトソースオプション例題 1

次の例題は、ソース領域の大きさを変えたものです。

例 6: ダクトソースオプションの例題 (2)

```
[ S o u r c e ]

set: c1[200]    $dl0
set: c2[500]    $dl1
set: c3[5000]   $dl2
set: c4[10.0]   $x*2 at z0
set: c5[10.0]   $y*2 at z0
set: c10[5.0]   $dxw
set: c20[5.0]   $dyw
set: c30[0.001] $dpf

s-type = 2
proj   =  neutron
e0     =  20.0
x0     = -c4/2
x1     =  c4/2
y0     = -c5/2
y1     =  c5/2
z0     =  0.0
z1     =  0.0
dir    =  1.0
phi    =  0.0

dom = -10
dl0 = c1
dl1 = c2
dl2 = c3
dxw = c10
dyw = c20
dpf = c30
```

次のグラフが示すように、この例題では、current、wall current ともに増加しています。この増加分がソース領域のダクトより大きい部分からの寄与を示します。従って、このオプションによって、ソースのマージン領域からの寄与を自動的に扱うことができます。

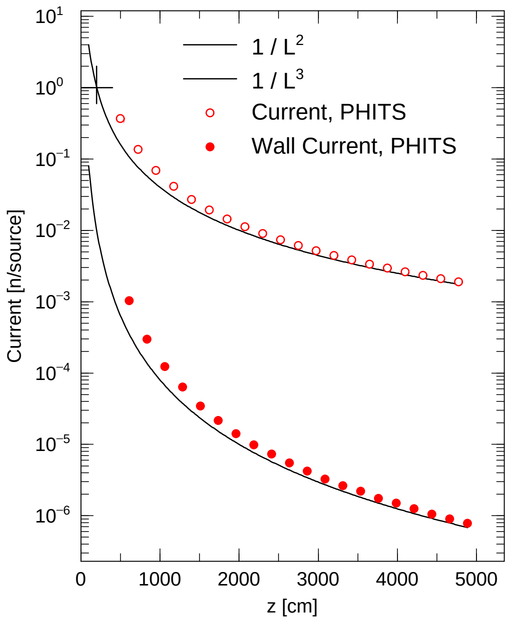


図 18: ダクトソースオプション例題 2

5.4 [Material] セクション

5.4.1 書式

このセクションでは、体系を構成する物質を定義します。ここで定義した物質は、[cell] セクションをはじめとする幾つかのセクションで参照されます。

書式は以下の通りです。物質番号の後に、物質を構成する元素（核種）とその組成比を交互に書きます。

```
[ M a t e r i a l ]
MAT[物質番号]    元素   組成比    元素   組成比    ...    ...
```

物質番号 *n* は 1 から 99999 まで選択でき、MAT[*n*] あるいは M*n* の 2 種類の表現で指定することができます。ただし、MAT[*n*] において、MAT と [の間には空白文字を入れることができないのでご注意ください。

コメント文字として#，%，!，$の 4 文字が使用できます。バージョン 2.88 までは c もコメント文字として使用することができていましたが、それより後のバージョンではデフォルトでは使用できなくなりました。もし、コメント文字として使用したい場合は、[parameters] セクションにおいて icommat=1 としてください[69]。また、ファイルのインクルード文、定数定義などは、セクションの途中でも用いることが出来ます。

5.4.2 元素（核種）の定義

上記書式の元素の箇所には、定義したい物質の元素記号を書いてください。核種を指定する場合は、質量数を加えた表式を書いてください。208Pb，Pb-208，82208 のどの形式でも可能です。例えば水素は、1H，H-1，1001 と指定します。元素記号のみ、あるいは質量数をゼロにした場合（例えば Pb，82000）は、その元素の同位体を天然存在比で展開します。ただし、JENDL-4.0 に含まれない核種に関しては展開されませんのでご注意ください。

PHITS パッケージに含まれている JENDL-4.0 以外の核データライブラリーを使用する場合は、断面積ディレクトリファイル file(7) (D=xsdir) にそのライブラリ付属の xsdir の情報を追加してください。もし、20MeV 以下の中性子データライブラリーについて、JENDL-4.0 以外のものを使用する場合は、物質番号毎にそのライブラリーの *id* を表 80 の NLIB により指定してください。ライブラリーの *id* は、ライブラリー番号（2 桁の数字）とデータのクラス（1 文字のアルファベット）を組み合わせて定義されており、例えば JENDL-4.0 の中性子ライブラリーであれば 50c となります。中性子以外にも、PLIB，ELIB，HLIB を指定することで、それぞれ光子、電子、陽子のライブラリーを指定して利用することができます。また、物質番号毎ではなく、核種毎に使用するデータライブラリーを指定する場合は、その核種の指定を 208Pb.50c，Pb-208.50c，82208.50c のような形式で行ってください。ただし、基本的に PHITS では、中性子などの入射粒子に対して各々1 種類のライブラリーを指定するようにし、その指定は NLIB，PLIB，ELIB，HLIB により行ってください。もし、2 種類目のライブラリーを指定したい場合は、中性子に関してのみ可能ですので、NLIB による指定を行った上で、ライブラリーを変更したい核種を 208Pb.50c のような形式で定義してください。なお、*id* によるライブラリーの指定がない場合は、file(7) 中のリストを上から検索し、その核種で最初に該当したライブラリーを使用します。どの核データが使用されたのかを知りたい場合は、[parameters] において kmout=1 とすることで、サマリーの出力ファイル file(6) (D=phits.out) にその情報が書き出されます。

[69] ただし、バージョン 2.89 から 2.95 までは、icommat=1 を含む [parameters] が c を使用している [material] より前に書かれている必要があります。

EGS5 を使用する場合は、1 つの物質番号に含む元素の数（核種の数ではありません）は 20 以下としてください。

5.4.3 物質組成比の定義

上記書式の組成比の箇所には、その物質番号で定義した物質に含まれる元素（核種）の組成比を書いてください。正の数で書いた場合は原子数比、負の数で書いた場合は質量比の意味となります。例えば、水 (H_2O) を定義したい場合は、

```
MAT[1] H 2  O 1
```

あるいは、

```
MAT[1] H -2/18  O -16/18
```

と指定します。水の分子量がおよそ 18 であることから、水に含まれる水素と酸素の質量比がそれぞれ約 2/18 と 16/18 となるためです。

粒子輸送計算時に使用される物質の密度は、`[cell]` セクションで定義します。ただし、`[t-deposit]`, `[t-deposit2]`, `[t-let]`, `[t-sed]` セクションにおいて `letmat` を指定する場合は、本セクションの組成比の箇所で物質密度を定義する必要があります。その際、正の数の場合は粒子密度（単位は [10^{24} atoms/cm^3]）、負の数の場合は質量密度（単位は [g/cm^3]）として定義します。

5.4.4 物質パラメータ

核データライブラリを使う領域に関して、物質毎のパラメータを *keyword=value* の形で指定します。挿入場所は、その物質番号のサブセクション中なら自由です。指定できるパラメータとその意味は以下の通りです。

表 80: 物質パラメータ

パラメータ	値	説明
GAS	D = 0	電子の阻止能の密度効果
	= 0	液体もしくは固体の計算に適する
	= 1	気体の計算に適する
ESTEP	= *n*	電子輸送のサブステップの数を *n* にする
		n が build-in のデフォルト値より小さい時は無視される
NLIB	= *id*	デフォルトの中性子ライブラリー番号を *id* にする
PLIB	= *id*	デフォルトの光子ライブラリー番号を *id* にする
ELIB	= *id*	デフォルトの電子ライブラリー番号を *id* にする
HLIB	= *id*	デフォルトの陽子ライブラリー番号を *id* にする
COND		伝導体設定
	< 0	絶縁体
	= 0	(Default) 1 個でも絶縁体があれば絶縁体、それ以外は伝導体
	> 0	1 個でも伝導体があれば伝導体

5.4.5 阻止能の指定

PHITS では、通常、物質の阻止能は ndedx パラメータで指定するモデル（ATIMA、SPAR など）で計算します。しかし、それらのモデルでは精度が十分ではない場合や、ICRU90 で定義した阻止能を使いたい場合は、阻止能のデータベースを直接読み込ませせることができます。その指定方法は

```
dedxfile = ファイル名
```

で指定できます。この阻止能データベースファイルは、file(29) で指定したフォルダ（初期設定は file(1)/data/dedx）に格納しておく必要があります。また、様々な物質に対する陽子及び α 粒子に対して PSTAR 及び ASTAR[70] で計算した阻止能データベースを上記フォルダに格納していますので、必要に応じて指定してご使用ください。なお、本機能は ATIMA で阻止能を計算する放射線に対してのみ有効となります。また、エネルギー分散 (nedisp) や角度分散 (nspred) は、従来と同様に機能します。

5.4.5.1 阻止能データベースファイルのフォーマット

dedxfile で指定する阻止能データベースファイルでは、$と#がコメントとして利用できます。また、PHITS 入力ファイルとほぼ同じく大文字小文字の区別はありません。指定できるパラメータは unit と kf のみで、それぞれ阻止能を与えるエネルギーの単位と放射線の種類を表します。

```
unit = 1: MeV (全運動エネルギー、デフォルト)
     = 2: MeV/u（atomic mass unit あたりの運動エネルギー）
     = 3: MeV/n（核子あたりの運動エネルギー）
```

unit は、最初の kf コードを指定する前に指定してください。kf は対象となる粒子の kf コードを指定し、その後に阻止能データベースを

エネルギー　阻止能

順にエネルギーの昇順で定義してください。ここで、エネルギーは unit で指定した単位、阻止能は MeV/(g/cm^2) の単位で与えてください。阻止能の単位は変更できませんのでご注意ください。指定したエネルギー範囲外の放射線や kf コードを指定しなかった放射線に対しては、ATIMA で計算した値がそのまま利用されます。複数の放射線に対する阻止能を定義する場合は、1 つのデータベースファイルにまとめてください。複数用意したデータベースファイルを 1 つの物質に対して使用することはできません。

5.4.6 $S(\alpha,\beta)$ の指定

低エネルギー中性子の輸送の際、熱中性子散乱データ $S(\alpha,\beta)$ のライブラリーが必要になる場合があります。特に、熱エネルギー付近で挙動が大きく変わりますのでご注意ください。$S(\alpha,\beta)$ ライブラリーは、

```
MTn 物質 ID
```

で指定できます。ここで、n はその物質の物質番号です。物質 ID は、xsdir に書かれている ID 番号（lwtr.20t など）です。例えば、常温の水の場合は、

```
M1   H    2.0
     O    1.0
MT1  lwtr.20t
```

のように指定します。各データの物質や温度情報は/XS/tsl/tsl-table に書かれていますので、そちらをご参照下さい。

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

5.4.7 例題

幾つかの例題を下に示します。

例 7: [material] セクションの例題 (1)

```
[ M a t e r i a l ]
MAT[ 1 ]
   1H        1.0000000E-04
  208Pb      1.7238000E-02
  204Pb      4.6801000E-04
  206Pb      7.9430000E-03
  207Pb      7.2838000E-03
MAT[ 2 ]
   1H        1.0000000E-09
   14N       4.6801000E-05
   16O       7.9430000E-06
```

元素（核種）と組成比の順番は、元素-組成比がデフォルトで、これを逆にしたいときは、

例 8: [material] セクションの例題 (2)

```
[ M a t e r i a l ]
   den            nuc       <------
MAT[ 1 ]
1.0000000E-04      1H
1.7238000E-02    208Pb
4.6801000E-04    204Pb
7.9430000E-03    206Pb
7.2838000E-03    207Pb
MAT[ 2 ]
1.0000000E-09      1H
4.6801000E-05     14N
7.9430000E-06     16O
```

のように、den nuc を入れて定義します。

例 9: `[material]` セクションの例題 (3)

```
[ M a t e r i a l ]
m1     80196.49c  5.9595d-5
       80198.49c  3.9611d-3
       80199.49c  6.7025d-3
       80200.49c  9.1776d-3
       80201.49c  5.2364d-3
       80202.49c  1.1863d-2
       80204.49c  2.2795d-3
$      ...Be...
m3     4009.37c  1.2362E-1
mt3    be.01
$      ...h2o (25C)...
m4     1001.37c  6.6658d-2      8016.37c  3.3329d-2
mt4    lwtr.01
$      ...b4c (natural boron; 25%-density)...
m5     6012.37c  6.8118d-3
       5011.37c  2.1825d-2
$      ...liquid-h2 (20K)...
m6     1001.49c  3.1371d-2      1011.49c  1.0457d-2
mt6    orthoh.00   parah.00
```

5.5 [Surface] セクション

5.5.1 書式

このセクションでは、[cell] セクションでセル（小部屋）を定義するために使用する面 (surface) を定義します。 PHITS で定義できる面には平面、球面、円柱面などがあり、これらの形状に対応した方程式の各パラメータを指定することで、それぞれの面を定義します。コメント文字として C の他、$が使用できます。ファイルのインクルード文、定数定義などは、セクションの途中でも用いることが出来ます。継続行には、先頭に 5 桁の空白が必要です。

書式は以下の通りです。面番号、座標変換番号、面記号、面定義数値の順番で、各面を定義します。各項目の意味は表 81 を参照してください。

```
[ S u r f a c e ]
面番号    座標変換番号    面記号    面定義数値
```

面記号は、表 82 に示す 1 つの方程式で表現できる面や、表 83 にある複数の面を組み合わせたマクロボディーの記号により指定します。面定義数値では定数や数式が使えます。また、面番号の前に “*” をつけると全反射面、“+” をつけると白色反射面を定義できます。例えば、無限に拡がる繰り返し構造を模擬する場合は、全反射面の機能を使うと便利です。

表 81: 面記述書式

項目	説明
面番号	1 ~ 999999 が使用できます。
座標変換番号	座標変換を行わない場合は指定しません。座標変換がある場合は、[transform] セクションで定義する TR*n* の番号 *n* を指定します。
面記号	表 82 および表 83 にある面記号を指定します。
面定義数値	定義する面により 1 個から 15 個の入力値を指定します。

表 82: 方程式で定義される面

面記号	型式	解説	方程式	入力数値
P	平面	汎用	$Ax+By+Cz-D=0$	$A,\ B,\ C,\ D$
PX		X 軸に垂直	$x-D=0$	D
PY		Y 軸に垂直	$y-D=0$	D
PZ		Z 軸に垂直	$z-D=0$	D
P	平面	3 点を通る面		$x_1,y_1,z_1,\ x_2,y_2,z_2,$ x_3,y_3,z_3
SO	球	中心が原点	$x^2+y^2+z^2-R^2=0$	R
S		汎用	$(x-x_0)^2+(y-y_0)^2+(z-z_0)^2-R^2=0$	$x_0,\ y_0,\ z_0,\ R$
SX		中心が X 軸上	$(x-x_0)^2+y^2+z^2-R^2=0$	$x_0,\ R$
SY		中心が Y 軸上	$x^2+(y-y_0)^2+z^2-R^2=0$	$y_0,\ R$
SZ		中心が Z 軸上	$x^2+y^2+(z-z_0)^2-R^2=0$	$z_0,\ R$
C/X	円柱	X 軸に平行	$(y-y_0)^2+(z-z_0)^2-R^2=0$	$y_0,\ z_0,\ R$
C/Y		Y 軸に平行	$(x-x_0)^2+(z-z_0)^2-R^2=0$	$x_0,\ z_0,\ R$
C/Z		Z 軸に平行	$(x-x_0)^2+(y-y_0)^2-R^2=0$	$x_0,\ y_0,\ R$
CX		X 軸上	$y^2+z^2-R^2=0$	R
CY		Y 軸上	$x^2+z^2-R^2=0$	R
CZ		Z 軸上	$x^2+y^2-R^2=0$	R
K/X	円錐	X 軸に平行	$\sqrt{(y-y_0)^2+(z-z_0)^2}\mp\lvert t\rvert\,(x-x_0)=0$	$x_0,\ y_0,\ z_0,\ \lvert t\rvert^2,\ k$
K/Y		Y 軸に平行	$\sqrt{(x-x_0)^2+(z-z_0)^2}\mp\lvert t\rvert\,(y-y_0)=0$	$x_0,\ y_0,\ z_0,\ \lvert t\rvert^2,\ k$
K/Z		Z 軸に平行	$\sqrt{(x-x_0)^2+(y-y_0)^2}\mp\lvert t\rvert\,(z-z_0)=0$	$x_0,\ y_0,\ z_0,\ \lvert t\rvert^2,\ k$
KX		X 軸上	$\sqrt{y^2+z^2}\mp\lvert t\rvert\,(x-x_0)=0$	$x_0,\ \lvert t\rvert^2,\ k$
KY		Y 軸上	$\sqrt{x^2+z^2}\mp\lvert t\rvert\,(y-y_0)=0$	$y_0,\ \lvert t\rvert^2,\ k$
KZ		Z 軸上	$\sqrt{x^2+y^2}\mp\lvert t\rvert\,(z-z_0)=0$	$z_0,\ \lvert t\rvert^2,\ k$ k は ±1 or 未指定
SQ	楕円 双曲面 放物面	X, Y, または Z 軸に 平行な軸	$A(x-x_0)^2+B(y-y_0)^2+C(z-z_0)^2+$ $2D(x-x_0)+2E(y-y_0)+2F(z-z_0)+G=0$	$A,\ B,\ C,\ D,\ E,$ $F,\ G,\ x_0,\ y_0,\ z_0$
GQ	円柱 円錐 楕円 双曲面 放物面	X, Y, または Z 軸に平行 でない軸	$Ax^2+By^2+Cz^2+Dxy+Eyz+$ $Fzx+Gx+Hy+Jz+K=0$	$A,\ B,\ C,\ D,\ E,$ $F,\ G,\ H,\ J,\ K$
TX	楕円環体 円環体	X, Y, または Z 軸に 平行	$(x-x_0)^2/B^2+$ $(\sqrt{(y-y_0)^2+(z-z_0)^2}-A)^2/C^2-1=0$	$x_0,\ y_0,\ z_0,\ A,\ B,\ C$
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			$(z-z_0)^2/B^2+$ $(\sqrt{(x-x_0)^2+(y-y_0)^2}-A)^2/C^2-1=0$	$x_0,\ y_0,\ z_0,\ A,\ B,\ C$

`[surface]` セクションで定義した面を `[cell]` セクションで使用する際、基本的にはその面によって作られる形状の内側をマイナス側、外側をプラス側として指定してください。ただし、平面などの閉じた形状でない面の場合は、その面を表す方程式 $f(x,y,z)=0$ に対して、任意の座標 (x_0,y_0,z_0) を代入して判別してください。左辺の値 $f(x_0,y_0,z_0)$ が正であれば座標 (x_0,y_0,z_0) が含まれる領域がプラス側、負であればマイナス側となります。例えば、`PY` で $D=5$ とした場合、方程式が $f(x,y,z)=y-5=0$ となるため、原点 $(0,0,0)$ を含む領域は $f(0,0,0)=0-5=-5$ のマイナス側となります。

面記号 `P` を用いて、3 つの座標を与えることにより平面を定義できます。この場合、座標原点を含む領域がマイナス側として定義されますので、`[cell]` セクションで使用する際はご注意ください。

円錐の場合、x_0,y_0,z_0 等で指定する頂点を中心に、中心軸に沿ってプラス側とマイナス側の両方に円錐面が定義されます。k の入力を 1 とすると軸のプラス側に広がる面のみを指定し、$k=-1$ とすると軸のマイナス側に広がる面のみを指定します。k に何の値も与えない場合はその両方を指定します。ただし、片方の円錐面しか使用しない場合は、`[cell]` セクションで領域を定義する際に、円錐の頂点を通る平面でも囲むよ

うにしてください。すなわち、円錐を領域として定義する場合は、円錐の頂点を通る平面、片方の円錐面、円錐の底面の 3 つが必要です。

楕円環帯や円環体を定義する `TX,TY,TZ` に対しては `[transform]` による座標変換が利用できませんのでご注意ください。

表 83: マクロボディー

面記号	型式	入力数値	解説
BOX	任意の BOX 平面 （全ての角が 90 度）	x_0, y_0, z_0, A_x, A_y, A_z, B_x, B_y, B_z, C_x, C_y, C_z	基点の座標 基点からの第 1 面のベクトル 基点からの第 2 面のベクトル 基点からの第 3 面のベクトル
RPP	直方体 （各面が x,y,z 軸に 垂直）	$x_{\min}$, $x_{\max}$, $y_{\min}$, $y_{\max}$, $z_{\min}$, $z_{\max}$	x 座標の最小値、最大値 y 座標の最小値、最大値 z 座標の最小値、最大値
SPH	球 （汎用球 S と同じ）	x_0, y_0, z_0, R	中心座標 球の半径
RCC	円柱	x_0, y_0, z_0, H_x, H_y, H_z, R	底面の中心座標 底面の中心からの上面中心へのベクトル 円柱の半径
RHP or HEX	任意の 6 角柱 プリズム	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	基点の座標 基点からの高さベクトル 基点からの第 1 面へのベクトル 基点からの第 2 面へのベクトル 基点からの第 3 面へのベクトル
REC	楕円柱	x_0 y_0 z_0, H_x, H_y, H_z, A_x, A_y, A_z, B_x, B_y, B_z	楕円柱底面の中心座標 シリンダーの高さベクトル ($\boldsymbol{H}$) $\boldsymbol{H}$ に垂直な楕円の長軸ベクトル ($\boldsymbol{A}$) $\boldsymbol{H}$ と $\boldsymbol{A}$ に垂直な楕円の短軸ベクトル ($\boldsymbol{B}$)
TRC	カットされた円錐	x_0, y_0, z_0, H_x, H_y, H_z, R_1, R_2	円錐底面の中心座標 円錐の高さベクトル 底面の半径 上面の半径
ELL	楕円体	$R > 0$ のとき、 x_1, y_1, z_1, x_2, y_2, z_2, R $R < 0$ のとき、 x_0, y_0, z_0, A_x, A_y, A_z, R	 第 1 焦点の座標 第 2 焦点の座標 長軸の半径 この場合、長軸を中心に回転して楕円体を作りますので、 長軸 < 短軸となっても問題ありません 楕円体の中心 長軸のベクトル 短軸の半径
WED	くさび形	x_0, y_0, z_0, A_x, A_y, A_z, B_x, B_y, B_z, H_x, H_y, H_z	頂点の座標 底面の三角形においてある頂点から別の頂点へのベクトル ($\boldsymbol{A}$) $\boldsymbol{A}$ の頂点とは別の頂点へのベクトル ($\boldsymbol{B}$) 高さベクトル ($\boldsymbol{H}$)

カットされた円錐 (`TRC`) を用いる際、上面の半径 (R_2) を 0 にしないよう注意してください。例えば、`TRC` を用いてカットされていない（頂点をもつ）円錐を定義する場合は、R_2 の値に小さな値を入れてください。

くさび形 `WED` は、その底面が直角三角形である必要があります。任意の三角形は直角三角形を組み合わせて作成してください。

5.5.2 面の記述方法

本節では、表 82 と 83 で示した各面と各マクロボディーについて、例題と図を用いて説明します。

例 10: 平面の例題 (1)

```
1:   [surface]
2:     1   PY     5
```

`PX`,`PY`,`PZ` はそれぞれ x, y, z 軸に垂直な平面を定義します。この例題では、面記号に `PY`、面定義数値に 5 を指定することで、面番号 1 の面を y 軸に垂直な $y = 5$ の平面として定義しています。この面は、図 19 の破線で示した面のようになります。

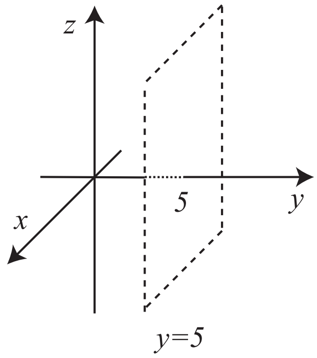


図 19: 例題 10 で定義した $y = 5$ の平面。

例 11: 平面の例題 (2)

```
1:   [surface]
2:     1   P      2 2 1  10
3:     2   P      5 0 0  0 5 0  0 0 10
```

任意の面を定義する際は、面記号に `P` を指定します。この場合、面定義数値の指定方法には 2 種類あり、方程式 $Ax+By+Cz-D = 0$ を満たす A, B, C, D を指定する方法と、平面を通る 3 つの座標 $(x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3)$ を指定する方法があります。前者の方法は、定義したい面に垂直な法線ベクトル (s, t, u) とその面が通る点の座標 (x_0, y_0, z_0) がわかっている場合に利用できます。この条件を満たす平面の方程式は、

$$s(x - x_0) + t(y - y_0) + u(z - z_0) = 0 \tag{6}$$

となり、これを変形すると、

$$sx + ty + uz - (sx_0 + ty_0 + uz_0) = 0 \tag{7}$$

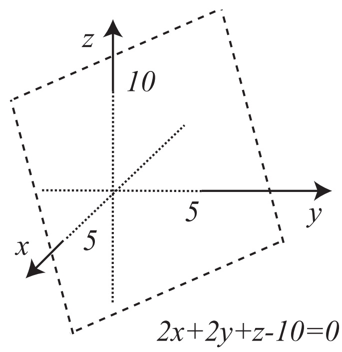


図 20: 例題 11 で定義した面。

となるため、$A = s, B = t, C = u, D = sx_0 + ty_0 + uz_0$ を指定することで、目的の面を定義できます。例題 (2) に示した面番号 1 は、法線ベクトルを (2, 2, 1) とし、座標 (5, 0, 0) を通る面です。この面は、図 20 の破線で示した面のようになります。後者の方法では、定義したい面が通過する 3 つの座標を指定します。例題 (2) の面番号 2 は、座標 (5, 0, 0), (0, 5, 0), (0, 0, 10) を通る面となっており、面番号 1 の面と同じものです。

例 12: 球面の例題

```
[surface]
  1   SO    5
  2   SZ    10  3
  3   S     10 10 0  3
```

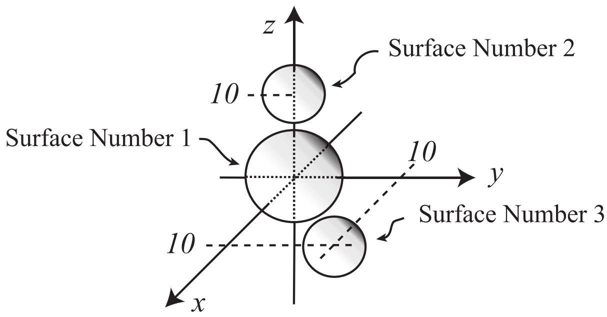


図 21: 例題 12 で定義した 3 つの球面。

球面を定義する際は、SO,SX,SY,SZ,S を指定します。SO は座標原点を中心とする球面を定義する場合に指定し、面定義数値としてその半径 R を指定します。この例題では、2 行目において、原点を中心として半径を 5cm とする球面を面番号 1 の面として定義しています。SX,SY,SZ は、それぞれ x, y, z 軸上に中心をもつ球面を定義する際に指定します。面定義数値は、各軸上の値と半径 R を指定します。例題の 3 行目では、z 軸上の座標 $(0, 0, 10)$ を中心とする半径 3cm の球面を面番号 2 の面として定義しています。任意の座標 (x_0, y_0, z_0) を中心とする球面を定義する場合は S を指定します。この場合、面定義数値には、球の中心を示す座標 (x_0, y_0, z_0) と半径 R を指定します。例題では、$(10, 10, 0)$ を中心とする半径 3cm の球面を面番号 3 の面として定義しています。例題で定義した面番号 1,2,3 の面は図 21 に示す 3 つの球面のようになります。

例 13: 円柱側面の例題

```
[surface]
  1   CY    5
  2   C/Y   15 0  3
```

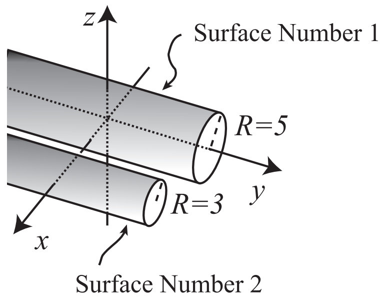


図 22: 例題 13 で定義した 2 つの円柱の側面。

x, y, z 軸を中心軸とする円柱の側面を定義する際には、それぞれ CX,CY,CZ を指定します。この場合、面定義数値には円柱の半径 R を指定します。例題の 2 行目では、y 軸を中心軸とする半径 5cm の円柱の側面を定義しています。これは図 22 の右上に示す円柱の側面に対応しています。次に、C/X,C/Y,C/Z は、それぞれ x, y, z 軸に平行な直線を中心軸とする円柱の側面を定義する際に指定します。この場合、円柱の半径 R の他に、中心軸が通る点の座標を指定する必要があります。例題の 3 行目では、y 軸に平行で座標 $(15, 0, 0)$ を通る直線を中心軸とする円柱の側面を定義しています。$x_0 = 15, z_0 = 0$ の値を面定義数値の 1,2 番目のそれぞれで指定しており、最後に半径 3cm を与えています。この面は、図 22 の左下に示す円柱の側面に対応します。

x, y, z 軸に平行でない直線を中心軸とする円柱は面記号 GQ を用いて定義できますが、[transform] を使った座標変換を利用する方が便利です。

例 14: 円錐面の例題

```
1:   [surface]
2:     1   KZ     0  1
3:     2   K/Z    0 20 0  1/3  1
```

x, y, z 軸を中心軸とする円錐面を定義する際は、KX,KY,KZ を指定します。面定義数値には、各軸上の頂点の座標、円錐の角度に関するパラメータ $|t|^2$ および円錐の広がる方向に関するパラメータ k（省略可）を指定します。例題の 2 行目に KZ を指定した例を示しており、面定義数値の 1 番目に頂点の z 座標 $z_0 = 0$、2 番目に $|t|^2 = 1$ を指定しています。面記号 KZ の方程式は、

$$\sqrt{x^2 + y^2} \mp |t|\,(z - z_0) = 0 \tag{8}$$

なので、各面定義数値の値を代入すると、

$$\sqrt{x^2 + y^2} - z = 0,\ (z > 0) \tag{9}$$

$$\sqrt{x^2 + y^2} + z = 0,\ (z < 0) \tag{10}$$

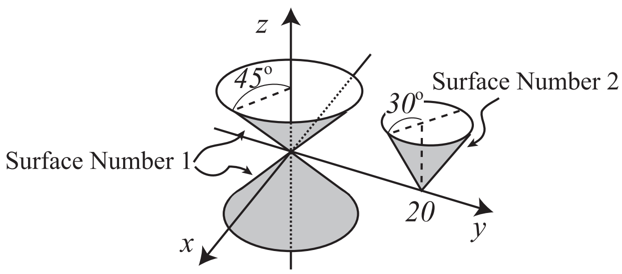


図 23: 例題 14 で定義した円錐面。

となります。この場合、面番号 1 の面は、円錐の頂点を $(0, 0, 0)$ とし、z 軸を中心軸とする円錐面となり、図 23 の左側に示す面のようになります。頂点を境にして、z 軸に沿ってプラス側が式 (9) に対応しており、マイナス側が式 (10) に対応しています。これらはそれぞれ面定義数値の最後に $k = 1$ あるいは $k = -1$ を指定することで定義できますが、例題の 2 行目では k を未指定としており、この場合は両方の面を定義することになります。また、面定義数値の 2 番目は、円錐の中心軸と母線のなす角度 θ と、$|t|^2 = \tan^2\theta$ の関係にあります。よって、面番号 1 の場合は $\tan\theta = 1$、すなわち $\theta = 45°$ となります。次に、x, y, z 軸に平行な中心軸をもち、座標 (x_0, y_0, z_0) を頂点とする円錐面を定義する場合は、K/X,K/Y,K/Z を指定します。例題の 3 行目に、K/Z を指定した例を示します。頂点は $(0, 20, 0)$ であり、z 軸に平行な中心軸をもつ円錐面を定義しています。$|t|^2 = 1/3$ であることから、$\tan\theta = 1/\sqrt{3}$、すなわち $\theta = 30°$ となります。また、$k = 1$ とすることで、プラス側の円錐面のみを定義しています。

例 15: 楕円面の例題

```
1:   [surface]
2:     1   SQ     1/9**2 1/6**2 1/3**2  0 0 0  -1  0 0 0
```

図 24 に示すような楕円面を定義する場合は、SQ を指定します。楕円面の方程式は、x, y, z 軸方向の半径がそれぞれ a, b, c の場合に、

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \tag{11}$$

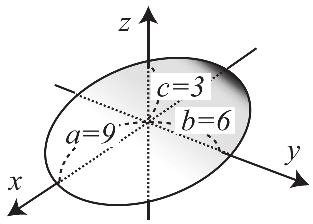


図 24: 例題 15 で定義した楕円面。

と書けます。よって、SQ の面定義数値の 1,2,3 番目にそれぞれ $A = 1/a^2, B = 1/b^2, C = 1/c^2$ を指定し、4,5,6 番目には $D = E = F = 0$、7 番目には $G = -1$ を指定することで、楕円面を定義できます。例題では、x, y, z 軸方向の半径がそれぞれ 9, 6, 3cm の場合の楕円面を定義しています。楕円を回転させて定義される回転楕円体を指定する場合は、回転軸以外の軸方向の径の長さを一致させます。例えば、x 軸を回転軸とする場合は、$b = c$ すなわち $B = C$ と指定します。また、面定義数値の 8,9,10 番目で、楕円体の中心の座標 (x_0, y_0, z_0) を指定します。例題では、座標原点 $(0, 0, 0)$ を楕円体の中心としています。なお、D, E, F に 0 以外の値を指定すると、楕円面以外の 2 次曲面を表すことになるのでご注意ください。

例 16: 双曲面の例題

```
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
```

図 25 の中央と右側に示したような双曲面を定義する場合は、面記号に SQ を指定します。中央に示したものは一葉双曲面、右側に示したものは二葉双曲面と呼ばれており、それぞれ次の方程式によって定義されます。

$$\frac{x^2}{a^2}+\frac{y^2}{b^2}-\frac{z^2}{c^2} = 1 \quad (12)$$

$$-\frac{x^2}{a^2}-\frac{y^2}{b^2}+\frac{z^2}{c^2} = 1 \quad (13)$$

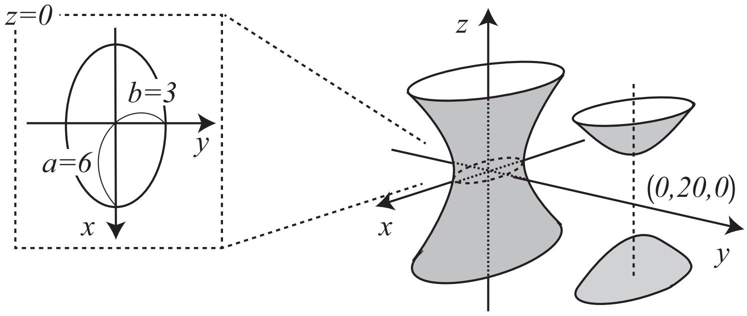


図 25: 例題 16 で定義した双曲面。

ただし、これらは z 軸を中心軸としたものです。また、$a=b$ の場合は回転双曲面となります。例題の 2 行目で定義している面番号 1 が、図 25 の中央にある一葉双曲面となります。面定義数値の 1,2,3 番目にそれぞれ $A=1/6^2, B=1/3^2, C=-1/5^2$ を指定し、4,5,6 番目には $D=E=F=0$、7 番目には $G=-1$ を指定しています。中心軸は z 軸であり、面定義数値の 8,9,10 番目に $(0,0,0)$ を指定することで、中心軸が座標原点を通ります。この双曲面と z 軸に垂直な平面の交線は楕円となります。図 25 の左側に示したのが、$z=0$ の平面上で面番号 1 の面を見た図です。x, y 軸方向の半径がそれぞれ $a=6, b=3$cm の楕円となっています。例題の 3 行目で定義しているのが、図 25 の右側に示す二葉双曲面です。面番号 1 とは a, b, c の符号を逆転させており、面定義数値の 4,5,6 番目を 0、7 番目を $G=-1$ とすることで、二葉双曲面を定義しています。ただし、面定義数値の 8,9,10 番目において $(x_0, y_0, z_0)=(0,20,0)$ と指定することで、中心軸が $(0,20,0)$ を通過する面となっています。また、二葉双曲面を用いて `[cell]` セクションで領域を定義する際は、座標 (x_0, y_0, z_0) を含む領域がマイナス側、含まない方がプラス側となります。図の右下にある双曲面の内側の領域もプラス側として指定することになりますので、ご注意ください。

例 17: 放物面の例題

```
1:   [surface]
2:     1   SQ     1 1/2**2 0  0 0 -1  0  0 0 0
```

図 26 の中央に示したような放物面を定義する場合には、例題 17 のように面記号 SQ を用いて指定します。一般に z 軸を中心軸とする放物面の式は次のようになります。

$$z=\frac{x^2}{a^2}+\frac{y^2}{b^2} \quad (14)$$

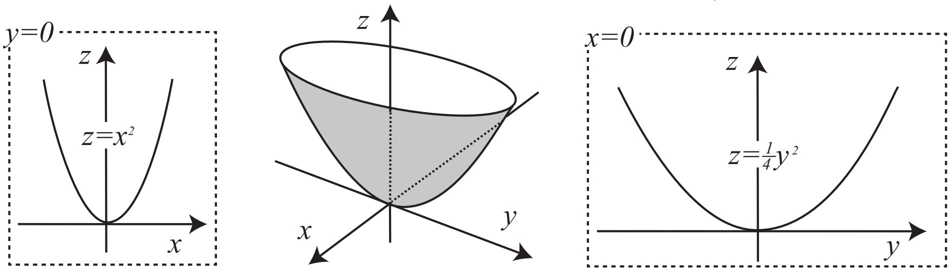


図 26: 例題 17 で定義した放物面。

よって、面定義数値の 1,2,3 番目にそれぞれ $A=1/a^2, B=1/b^2, C=0$ を指定し、4,5,6 番目には $D=E=0, F=-1$、7 番目には $G=0$ を指定することで、z 軸を中心軸とする放物面を定義できます。例題では $a=1, b=2$ の放物面を定義しています。この放物面と z 軸を含む平面の交線は放物線となり、$y=0$ と $x=0$ の平面上で見ると、それぞれ図 26 の左側と右側に示す放物線となります。また、放物面の頂点 (x_0, y_0, z_0) は面定義数値の 8,9,10 番目で指定します。

例 18: GQ の例題

```
[surface]
set: c1[30]
set: c2[cos(c1/180*pi)]
set: c3[sin(c1/180*pi)]
  1   GQ     c2**2 1/2**2 c3**2  0 0 -2*c2*c3  -c3 0 -c2  0
```

面記号 GQ は、x, y, z の 2 次式で表現される曲面を定義する場合に使用します。同様に 2 次式で表現される CX 等（円柱）、KX 等（円錐）、SQ（楕円面等）との違いは、中心軸が x, y, z 軸と平行でない曲面に対しても定義できる点です。ただし、単純な 1 次変換であれば、CX,KX,SQ 等で定義した面に対して [transform] セクションで定義した座標変換を適用する方が便利です。例題 18 では、例題 17 で取り扱った放物面を y 軸回りに 30° だけ回転させた例を示しており、図 27 ような面を定義できます。この例題で指定した面定義数値は、下記の計算により求めたものです。座標 (x, y, z) を y 軸周りに θ 回転させた座標を (x', y', z') とすると、これらの関係は、

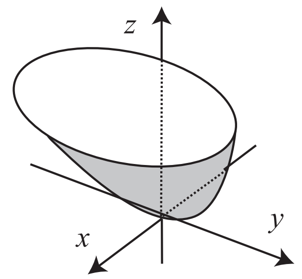


図 27: 例題 18 で定義した放物面。

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}, \tag{15}$$

と書けます。この関係式より、

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} x'\cos\theta - z'\sin\theta \\ y' \\ x'\sin\theta + z'\cos\theta \end{pmatrix} \tag{16}$$

が求まるので、式 (14) に代入すると、

$$\begin{aligned} x'\sin\theta + z'\cos\theta &= \frac{(x'\cos\theta - z'\sin\theta)^2}{a^2} + \frac{y'^2}{b^2}, \\ x'\sin\theta + z'\cos\theta &= \frac{x'^2\cos^2\theta + z'^2\sin^2\theta - 2x'z'\cos\theta\sin\theta}{a^2} + \frac{y'^2}{b^2}, \\ x'\sin\theta + z'\cos\theta &= \frac{\cos^2\theta}{a^2}x'^2 + \frac{\sin^2\theta}{a^2}z'^2 - \frac{2\cos\theta\sin\theta}{a^2}x'z' + \frac{y'^2}{b^2}, \\ 0 &= \frac{\cos^2\theta}{a^2}x'^2 + \frac{1}{b^2}y'^2 + \frac{\sin^2\theta}{a^2}z'^2 - \frac{2\cos\theta\sin\theta}{a^2}x'z' - x'\sin\theta - z'\cos\theta \end{aligned} \tag{17}$$

と x', y', z' の方程式が得られます。この式と表 82 にある GQ の方程式を比べ、各面定義数値を指定すると、放物面を y 軸周りに θ だけ回転させた面を定義できます。この例題では、$\theta = 30°$ を c1 として定数定義しており、これを用いた $\sin\theta, \cos\theta$ をそれぞれ c2,c3 として定数定義しています。

例 19: 楕円環面（トーラス面）の例題

```
[surface]
  1   TZ     0 0 0  10 3 5
```

図 28 の中央に示すような楕円環面（トーラス面）は、楕円をその円の外側にある軸を中心に回転させて得られる面です。面記号 `TX,TY,TZ` を用いて定義することができ、それぞれ x, y, z 軸を回転軸とする楕円環面を定義します。z 軸を回転軸とする楕円環面の方程式は、

$$\frac{\left(\sqrt{x^2+y^2}-R\right)^2}{a^2}+\frac{z^2}{b^2}=1 \tag{18}$$

と書くことができます。この場合の楕円は長軸方向に $2a$ 短軸方向に $2b$ の大きさをもち、楕円の中心と回転軸の距離は R となります。例題 19 では楕円環体の中心を原点としており、面定義数値の 1,2,3 番目は全て 0 と指定しています。面定義数値の 4 番目に $A = R =$ 10cm、5 番目に $B = b =$ 3cm、6 番目に $C = a =$ 5cm を指定することで、長軸方向に 10cm、短軸方向に 6cm の大きさをもつ楕円を z 軸から 10cm 離れた位置で回転させた楕円環面を定義しています。図 28 の左側に示したのが $z = 0$ の平面との交線です。原点と環の中心との距離は 10cm、環の大きさは $5 \cdot 2 =$ 10cm となります。図 28 の右側に示したのが $y = 0$ の平面との交線です。ただし、x 軸のプラス側のみ示しています。環の断面図となっており、楕円が長軸方向に 10cm、短軸方向に 6cm の大きさをもっていることがわかります。

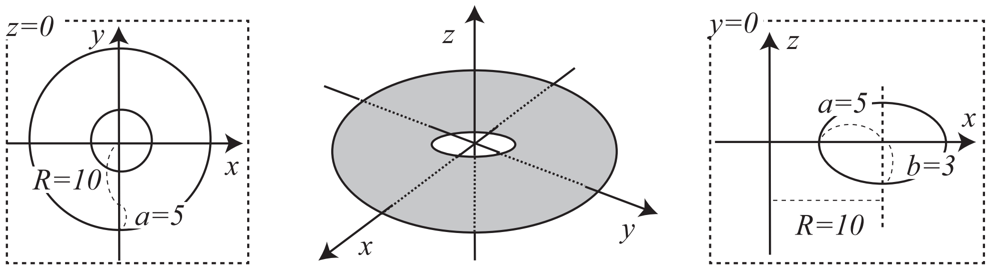


図 28: 例題 19 で定義した楕円環面。

例 20: `BOX` の例題

```
[surface]
  1   BOX    5 5 0  9 0 0  0 6 0  0 0 3
```

任意の直方体の面を定義する場合は、面記号 `BOX` を指定します。面定義数値には、基点の座標 $P(x_0, y_0, z_0)$ と、直方体を形作る 3 つのベクトル $\boldsymbol{A}(A_x, A_y, A_z)$, $\boldsymbol{B}(B_x, B_y, B_z)$, $\boldsymbol{C}(C_x, C_y, C_z)$ の各成分を指定します。これらの関係は図 29 のようになります。例題 20 では、点 $(5, 5, 0)$ を頂点の 1 つとし、$\boldsymbol{A} = (9, 0, 0)$, $\boldsymbol{B} = (0, 6, 0)$, $\boldsymbol{C} = (0, 0, 3)$ の 3 つのベクトルが示す方向に広がった直方体の面を定義しています。任意の直方体の面を定義する際、$\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ は x, y, z 軸に平行である必要はありませんが、各々のベクトルの関係は垂直である必要があります。

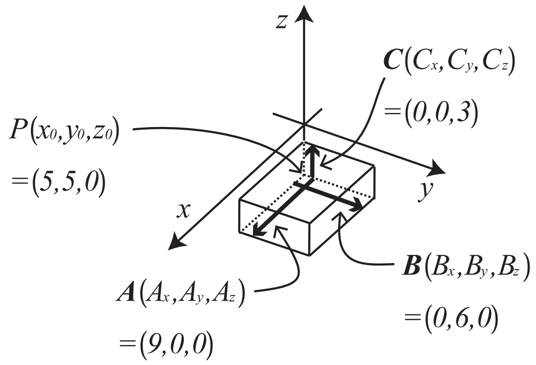


図 29: 例題 20 で定義した直方体の面。

例 21: RPP の例題

```
[surface]
  1   RPP    5 14  5 11  0 3
```

面記号 RPP は、xy, yz, zx 平面に平行な各々2 つずつの平面で囲まれる直方体の面を定義する場合に指定します。よって、BOX とは違い、各軸に対して任意の傾きをもつ直方体は定義できません。面定義数値として、x, y, z に関するそれぞれの最小値と最大値を与えます。例題 21 では、面定義数値の 1 から 6 番目に、$x_{\min} = 5, x_{\max} = 14, y_{\min} = 5, y_{\max} = 11, z_{\min} = 0, z_{\max} = 3$ を指定しており、例題 20 で定義したものと同じ直方体の面を定義しています。

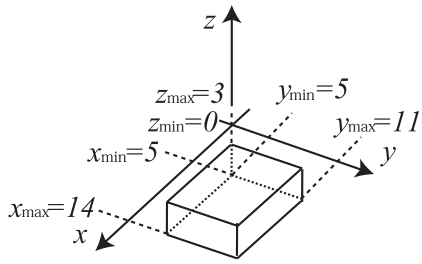


図 30: 例題 21 で定義した直方体の面。

例 22: SPH の例題

```
[surface]
  1   SPH    5 5 5  5
```

面記号 SPH を指定することで、任意の点を中心とする球面を定義できます。面定義数値には、球の中心の座標 (x_0, y_0, z_0) を 1,2,3 番目に指定し、球の半径を 4 番目に指定します。例題 22 では、点 $(5, 5, 5)$ を中心とし、半径を $R = 5$cm とする球面を定義しています。なお、任意の球面 S との違いはありません。

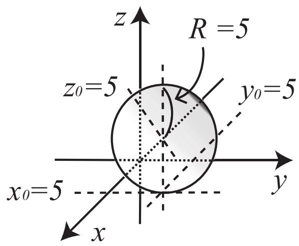


図 31: 例題 22 で定義した球面。

例 23: RCC の例題

```
[surface]
  1   RCC    5 5 0  0 0 10  5
```

面記号 RCC を指定することで、円柱面を定義できます。面定義数値として、円柱の底面の中心座標 $P(x_0, y_0, z_0)$ の各成分、その点から上面中心へのベクトル $\boldsymbol{H}(H_x, H_y, H_z)$ の各成分と半径 R を指定します。これらの関係は図 32 のようになります。例題では、$P(5, 5, 0)$ を円柱底面の中心とし、上面中心へのベクトルを $\boldsymbol{H} = (0, 0, 10)$、半径を $R = 5$cm としています。CX や C/X 等とは違い、x, y, z 軸に平行ではない直線を中心軸とする円柱面を定義する際に便利です。

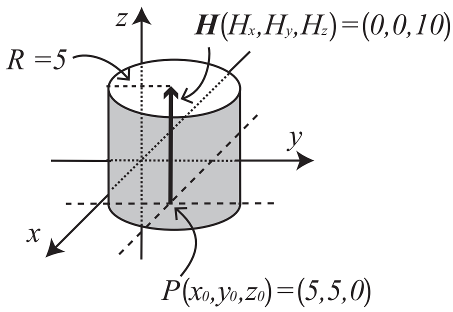


図 32: 例題 23 で定義した円柱面。

例 24: RHP の例題

```
1:  [surface]
2:    1  RHP   0 0 0  0 0 10  5 0 0  2 -5 0  -2 -5 0
```

図 33 の右側に示すような六角柱の面を定義する場合は、面記号 RHP あるいは HEX を指定します。面定義数値には、六角形となる底面の中心の座標 $P(x_0, y_0, z_0)$ の各成分と、その点から上面中心へのベクトル $\boldsymbol{H}(H_x, H_y, H_z)$ の各成分の他、六角形を定義するための 3 つのベクトル $\boldsymbol{A}(A_x, A_y, A_z)$, $\boldsymbol{B}(B_x, B_y, B_z)$, $\boldsymbol{C}(C_x, C_y, C_z)$ の各成分を指定します。六角形の中心 P と 3 つのベクトル $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ の関係は、図 33 の左側に示すようになります。すなわち、点 P から六角形の連続する 3 つの辺に下ろした垂線の大きさと方向を 3 つのベクトルとして与えます。例題 24 では、原点を底面の中心とし、z 軸方向に 10cm の高さをもつ六角柱の面を定義しています。面定義数値の 1,2,3 番目には原点 $(0, 0, 0)$ の各成分を指定し、4,5,6 番目では $\boldsymbol{H}(H_x, H_y, H_z) = (0, 0, 10)$ の各成分を指定しています。7 番目から 15 番目までは $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$ の x, y, z 成分をそれぞれ順番に指定し、図 33 の左側に示す y 軸に関して対称な六角形を定義しています。

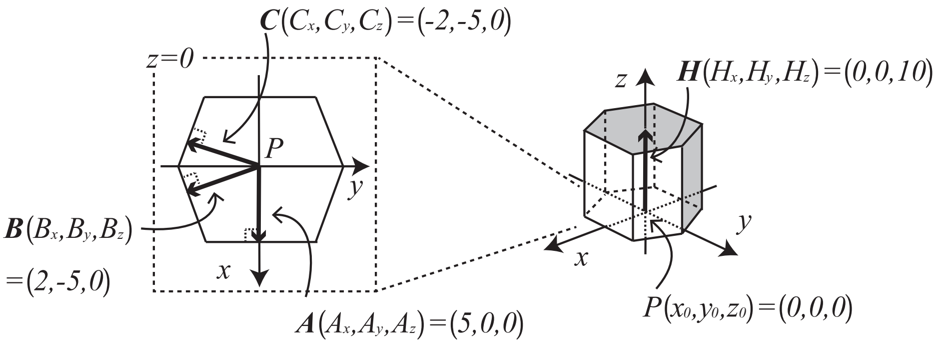


図 33: 例題 24 で定義した六角柱の面。

例 25: REC の例題

```
1:  [surface]
2:    1  REC   0 0 0  0 0 10  5 0 0  0 2 0
```

図 34 の右側に示すような楕円柱の面を定義する場合は、面記号 REC を指定します。面定義数値には、底面の中心の座標 $P(x_0, y_0, z_0)$ の各成分と、その点から上面中心へのベクトル $\boldsymbol{H}(H_x, H_y, H_z)$ の各成分の他、楕円を定義するための 2 つのベクトル $\boldsymbol{A}(A_x, A_y, A_z)$, $\boldsymbol{B}(B_x, B_y, B_z)$ の各成分を指定します。楕円の中心 P と 2 つのベクトル $\boldsymbol{A}, \boldsymbol{B}$ の関係は、図 34 の左側に示すようになります。楕円の中心を始点として、図のように長軸ベクトル $\boldsymbol{A}$ と短軸ベクトル $\boldsymbol{B}$ を定義します。例題 25 では、原点を底面の中心とし、z 軸方向に 10cm の高さをもつ楕円柱の面を定義しています。面定義数値の 1,2,3 番目には原点 $(0, 0, 0)$ の各成分を指定し、4,5,6 番目では $\boldsymbol{H}(H_x, H_y, H_z) = (0, 0, 10)$ の各成分を指定しています。7 から 12 番目までは $\boldsymbol{A}$ と $\boldsymbol{B}$ の x, y, z 成分をそれぞれ順番に指定し、図 34 の左側に示す楕円を定義しています。

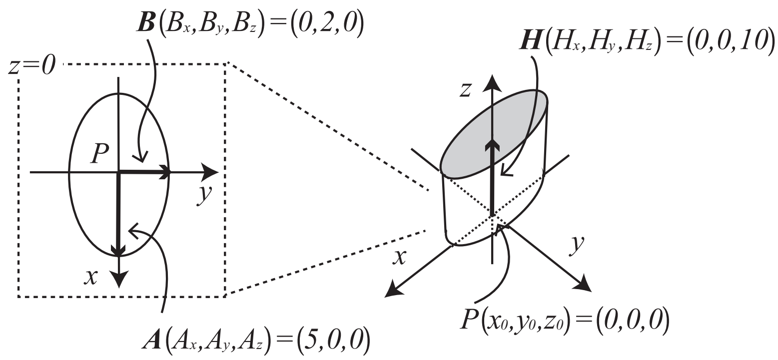


図 34: 例題 25 で定義した楕円柱の面。

例 26: TRC の例題

```
1:   [surface]
2:     1   TRC    0 0 0  0 0 10  5  2
```

図 35 に示すようなカットされた円錐の面を定義する場合は、面記号 TRC を指定します。面定義数値には、底面の中心の座標 $P(x_0, y_0, z_0)$ の各成分と、その点から上面中心へのベクトル $\boldsymbol{H}(H_x, H_y, H_z)$ の各成分に加えて、底面の円の半径 R_1 と上面の円の半径 R_2 を指定します。例題 26 では、原点を底面の中心とし、z 軸方向に 10cm の高さをもつカットされた円錐の面を定義しています。面定義数値の 1,2,3 番目には原点 $(0, 0, 0)$ の各成分を指定し、4,5,6 番目では $\boldsymbol{H}(H_x, H_y, H_z) = (0, 0, 10)$ の各成分を指定しています。底面の半径は R_1 = 5cm、上面の半径は R_2 = 2cm としており、これらを面定義数値の 7,8 番目のそれぞれで指定しています。

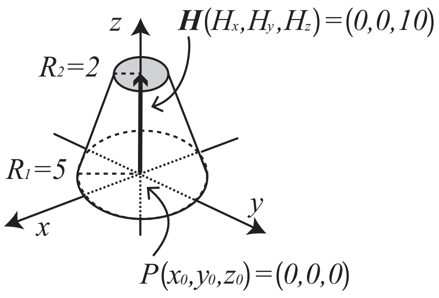


図 35: 例題 26 で定義したカットされた円錐の面。

なお、カットされた円錐 (TRC) を用いる際は、上面の半径 (R_2) を 0 にしないようにご注意ください。例えば、TRC を用いてカットされていない（頂点をもつ）円錐を定義する場合は、R_2 の値に小さな値を入れてください。

例 27: ELL の例題 (1)

```
1:   [surface]
2:     1   ELL    3 3 0  -3 -3 0  9
```

面記号 ELL を用いて、任意の位置に回転楕円体の面を定義できます。楕円の焦点の座標や中心の座標を指定することで楕円を定義し、その楕円を長軸周りに回転させることで、楕円面を定義します。ただし、この面記号は面定義数値の 7 番目に指定する R の符号によって、1 から 6 番目に指定する値の意味が変わります。例題 27 では R が正の場合を扱っており、この場合は、楕円を形作る 2 つの焦点と長軸の半径を指定します。面定義数値の 1,2,3 番目で楕円の第 1 焦点の座標 $P_1(x_1, y_1, z_1)$ を与え、4,5,6 番目で第 2 焦点の座標 $P_2(x_2, y_2, z_2)$、7 番目の値 R には長軸の半径を与えます。図 36 の右側に示したのが、例題 27 により定義された楕円面です。楕円の第 1,2 焦点の座標はそれぞれ $P_1(x_1, y_1, z_1) = (3, 3, 0)$, $P_2(x_2, y_2, z_2) = (-3, -3, 0)$ としており、xy 平面上に定義しています。また、長軸の方向は xy 平面上で x 軸から 45° 傾けた方向としており、図 36 の左側に示すような楕円となっています。長軸の長さは $2R$ = 18cm であり、この軸を回転軸とする回転楕円体の面が定義されます。

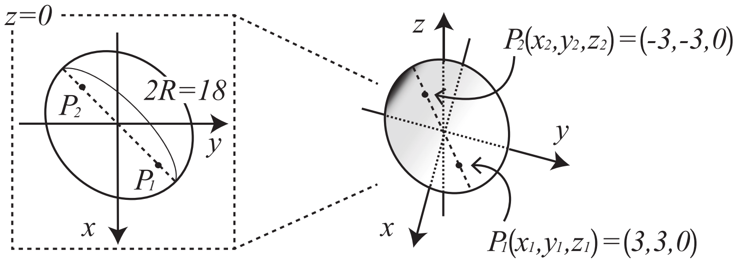


図 36: 例題 27 で定義した楕円面。

面記号 SQ で定義する楕円面との違いは、回転軸を傾けることが容易である点です。ただし、ELL は、楕円体の 2 つの軸の長さが等しい回転楕円体のみを定義する変数であることに注意してください。

例 28: ELL の例題 (2)

```
1:   [surface]
2:     1   ELL    0 0 0  6 6 0  -6
```

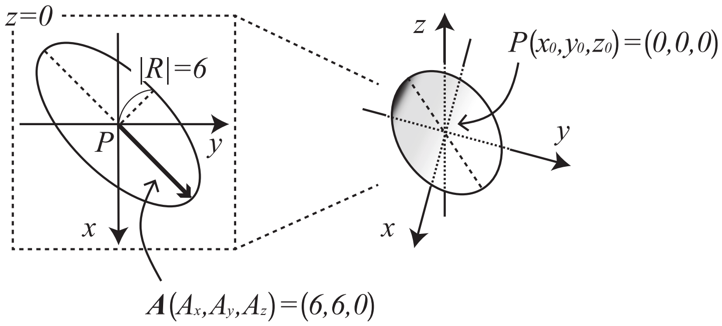


図 37: 例題 28 で定義した楕円面。

例題 28 では面記号 ELL の 7 番目の面定義数値 R が負の場合を扱っています。この場合は、楕円を定義するために、中心の座標 P と長軸ベクトル $\boldsymbol{A}$ および短軸の半径を指定します。例題では楕円の中心を原点としており、定義数値の 1,2,3 番目でその座標を $P(x_0, y_0, z_0) = (0, 0, 0)$ と与えています。4,5,6 番目では $\boldsymbol{A}$ を $(A_x, A_y, A_z) = (6, 6, 0)$ と与えており、図 37 の左側に示すような楕円を定義しています。7 番目の値 R はその絶対値が 6cm となっており、これが楕円の短軸の半径となります。図 37 の右側が、左側の楕円を長軸を回転軸として回転させて求めた楕円面です。

例 29: WED の例題

```
1:   [surface]
2:     1   WED    0 0 0  10 0 0  0 10 0  0 0 5
```

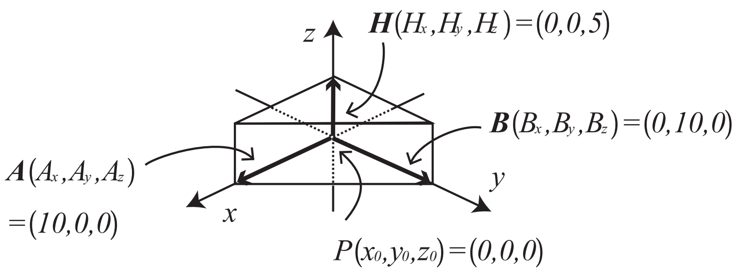


図 38: 例題 29 で定義したくさび形の面。

面記号 WED を用いて、図 38 に示すようなくさび形の面を定義できます。ただし、定義できるくさび形はその底面と上面が直角三角形となる三角柱のみです。例題 29 では、面定義数値の 1,2,3 番目で底面の 1 つの頂点の座標 $P(x_0, y_0, z_0) = (0, 0, 0)$ を指定しており、これを始点とする 2 つのベクトル $\boldsymbol{A}(A_x, A_y, A_z) = (10, 0, 0)$ と $\boldsymbol{B}(B_x, B_y, B_z) = (0, 10, 0)$ を 4,5,6 番目と 7,8,9 番目で指定することで、直角三角形を定義しています。面定義数値の 10,11,12 番目では、ベクトル $\boldsymbol{H}(H_x, H_y, H_z) = (0, 0, 5)$ を指定しており、三角柱の高さを決定しています。

5.5.3 マクロボディーの面定義

マクロボディーで定義した面をセル定義文で使うときは、マイナス記号がマクロボディーの内側、プラス記号が外側を表します。また、マクロボディーを構成するそれぞれの面もセル定義文で使うことができます。`[surface]` で定義した面番号の後にピリオド “.” を入れ、その後に表 84 にあるマクロボディー面番号を付けます。

各マクロボディー面番号は以下の通りです。

表 84: マクロボディー面番号

記号	マクロボディー面番号	解説
BOX	1	(A_x, A_y, A_z) の終点に垂直な面
	2	(A_x, A_y, A_z) の始点に垂直な面
	3	(B_x, B_y, B_z) の終点に垂直な面
	4	(B_x, B_y, B_z) の始点に垂直な面
	5	(C_x, C_y, C_z) の終点に垂直な面
	6	(C_x, C_y, C_z) の始点に垂直な面
RPP	1	x_{max} の面
	2	x_{min} の面
	3	y_{max} の面
	4	y_{min} の面
	5	z_{max} の面
	6	z_{min} の面
SPH		定義した面番号の面のみ
RCC	1	半径 R の円柱の側面
	2	(H_x, H_y, H_z) の終点に垂直な面
	3	(H_x, H_y, H_z) の始点に垂直な面
RHP or HEX	1	(A_x, A_y, A_z) の終点に垂直な面
	2	面 1 の対面
	3	(B_x, B_y, B_z) の終点に垂直な面
	4	面 3 の対面
	5	(C_x, C_y, C_z) の終点に垂直な面
	6	面 5 の対面
	7	(H_x, H_y, H_z) の終点に垂直な面
	8	(H_x, H_y, H_z) の始点に垂直な面
REC	1	楕円柱の側面
	2	(H_x, H_y, H_z) の終点に垂直な面
	3	(H_x, H_y, H_z) の始点に垂直な面
TRC	1	円錐面
	2	(H_x, H_y, H_z) の終点に垂直な面
	3	(H_x, H_y, H_z) の始点に垂直な面
ELL		定義した面番号の面のみ
WED	1	底面と上面の斜辺を含む面
	2	$\boldsymbol{B}$ と $\boldsymbol{H}$ を含む面
	3	$\boldsymbol{A}$ と $\boldsymbol{H}$ を含む面
	4	$\boldsymbol{A}$ と $\boldsymbol{B}$ および $\boldsymbol{H}$ の終点を含む面
	5	$\boldsymbol{A}$ と $\boldsymbol{B}$ および $\boldsymbol{H}$ の始点を含む面

5.6 [Cell] セクション

5.6.1 書式

このセクションでは、[surface] セクションで定義した面を使ってセル (小部屋) の定義を行います。これは GG(General Geometry) の考え方に基づいています。基本的にセルは閉じた空間として定義し、様々な形状のセルを組み合わせることにより、粒子を輸送する仮想空間を構築します。PHITS では、粒子が飛ぶことが可能な内部空間とそれ以外の空間を分けて定義する必要があり、後者を外部ボイド (真空) として明示的に指定しなければなりません。

コメント文字として C の他に$が使えます。このセクションではセルの定義に#を用いるため、PHITS の入力ファイルでのコメント文字#は利用できません。ファイルのインクルード文、定数定義などは、セクションの途中でも用いることが出来ます。継続行には、先頭に 5 桁の空白が必要です。

書式は以下の通りです。セル番号、物質番号、物質密度、セル定義文、キーワード形式のセルパラメータの順番で各セルを定義します。各項目の意味は表 85 を参照してください。

```
[ C e l l ]
セル番号    物質番号    物質密度    セル定義文    セルパラメータ
```

また、LIKE *n* BUT セルパラメータの表式や lattice 構造も使うことができます。似たような構造のセルを複数並べる場合に効率良く設定できます。利用方法の詳細は 5.6.5 節をご覧ください。セルパラメータの種類については表 86 にまとめています。

表 85: セル記述書式

項目	説明
セル番号	1 ~ 999999 まで使用できます。
物質番号	[material] セクションで指定した物質番号を入力します。ただし、そのセルが内部空間にあるボイド (真空) のときは 0 を、外部ボイドのときは −1 を指定します。
物質密度	そのセルに含まれる物質の密度を与えます。 単位は、正の場合は粒子密度 (10^{24} atoms/cm^3)、負の場合は質量密度 (g/cm^3) となります。ここで指定した粒子密度が計算に使用されますが、物質の組成比は [material] セクションで指定した密度を基に与えられます。従って、同じ組成で密度の異なる物質を定義可能となります。その際に、異なる物質番号を付加するパラメータ matadd を加えました。 セルがボイドのとき、すなわち物質番号が 0 か −1 の場合はこの項目は入力しません。
セル定義文	[surface] セクションで定義された面の面番号や集合代数の演算子を用いてセルの幾何形状を記述します。演算子の記号は ␣ (空白), :, #でそれぞれ積 (and), 和 (or), 否定 (not) を表します。(,) も使用できます。記述方法は 5.6.2 節をご参照ください。
LIKE *n* BUT	セルの内容がセル番号 *n* とほとんど同じで、BUT 以下に記述されるセルパラメータの値だけ違うセルを定義します。
セルパラメータ	書式は keyword=value です。キーワードとして VOL(体積), TMP(温度), TRCL (transform), U(universe), LAT(lattice), FILL が使用できます。また LIKE *n* BUT の場合は、これらに加えて MAT(物質番号), RHO (密度) が使えます。各キーワードの意味は表 86 を参照してください。

同じ物質番号を指定して別の密度をもつセルを複数定義した場合はご注意ください。2 番目以降のセル番号の物質番号として別の数字が割り振られます。

セル定義文において、積を意味する ⊔ (空白) の方が、和を意味する : より先に演算されるのでご注意ください。

表 86: セルパラメータ

項目	説明
`VOL`	そのセルの体積 (cm^3) を与えます。
`TMP`	そのセルの物質がもつ温度 (MeV) を与えます。
`TRCL`	そのセルを座標変換します。書式は `[transform]` セクションと同じです。
`U`	Universe 番号。1 ~ 999999 まで使用できます。Universe 構造を利用する場合に定義します。利用方法は 5.6.3 節を参照してください。
`LAT`	Lattice 番号。`LAT=1` の場合は四角柱、`LAT=2` の場合は六角柱を定義します。これらを単位とした繰り返し構造 (lattice 構造) を利用する場合に指定します。利用方法は 5.6.4 節を参照してください。
`FILL`	`U` で定義した universe 番号を入力し、そのセルを `U` で定義した宇宙 (universe) で満たします。利用方法は 5.6.3, 5.6.4 節を参照してください。
`MAT`	`LIKE` n `BUT MAT=`m の表式で用います。セル番号 n とは物質番号が違うセルを定義できます。
`RHO`	`LIKE` n `BUT RHO=`x の表式で用います。セル番号 n とは密度が違うセルを定義できます。密度の単位はセル番号 n と同じです。

5.6.2 セルの記述方法

任意の形状の構造物を仮想空間に構築する場合、最初にその構造物の表面を座標空間において定義し、次に対応する領域をその面で閉じることによってセル (小部屋) をつくる、という方法があります。この方法は General Geometry(GG) の考え方に基づいており、面の表側と裏側を区別して表現し、更に集合代数の演算子を利用することによって各セルを定義します。

考えている領域が定義した面の表側と裏側のどちらに属するかを区別するために、x, y, z で記述した面の方程式を利用します。具体的には、方程式 $f(x, y, z) = 0$ で定義された面があった場合に、ある領域の中の座標 (x_0, y_0, z_0) を代入した $f(x_0, y_0, z_0)$ が正であればその領域はプラス側、負であればマイナス側と表現します。例として、

例 30: `[cell]` セクションの例題 (1)

```
1:   [ Cell ]
2:     1   0  -10
3:     2  -1   10
4:   [ Surface ]
5:     10  SZ  3  5
```

を考えてみます。5 行目の SZ を用いて定義される面番号 10 の面は中心を (0,0,3)、半径を 5cm とする球面であるため、方程式は $f(x, y, z) = x^2 + y^2 + (z - 3)^2 - 5^2 = 0$ となります。球の内はマイナス側、外はプラス側と定義されますが、これらは球の中心の座標 (0,0,3) や球の外の適当な座標 (0,0,10) を代入して、$f(0, 0, 3) = -25 < 0$, $f(0, 0, 10) = 24 > 0$ となることから確認できます。`[cell]` セクションではこの様な領域の区切り方を行った上で、必要に応じて各領域にセル番号を付与します。例題 (1) の 2 行目では、セル番号 1 のセルがボイド (真空) であることと、その領域が球面のマイナス側であることが定義されており、図 39 で示す仮想空間を構築します。図 39 は、例題 (1) でつくられる領域を xz 平面で切った断面図として見た

ものです。本来は球ですが、2 次元平面で見た結果円となっています。例題 (1) の 3 行目では外部ボイドが明示的に定義されており、そのセル番号を 2 と指定しています。

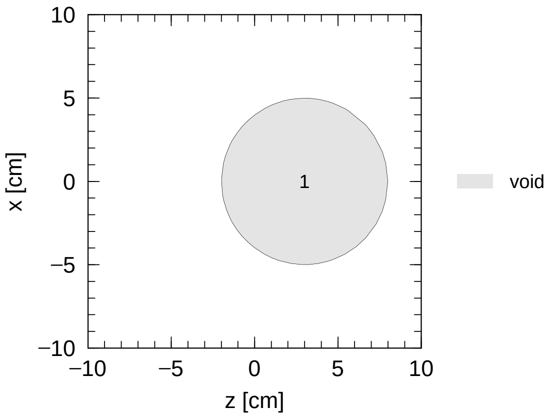


図 39: 例題 (1) の空間を *xz* 平面で切り取った断面図。セル番号 1 の領域はボイド (真空)。

例題 (1) の球面の様に 1 つの面だけで閉じた空間が定義できる場合は簡単ですが、`[surface]` セクションで定義できる面の多くは半無限領域をつくるため、幾つかの領域を組み合わせる必要があります。これを行うにあたって、PHITS では集合代数の考え方を導入しており、次の演算子を用いて閉じた領域を構成します。演算子は ␣ (空白), `:`, `#`の 3 つで、それぞれ積 (and), 和 (or), 否定 (not) を表します。また、`(　　)` で囲んだ範囲は 1 つの領域とみなします。ただし、␣ (空白) と`:`は面番号同士を演算するのに対して、`#`はセル番号に掛かります。また、`(　　)` を用いてまとめた領域に対しては、どの演算子でも掛けることができます。それでは、演算子を用いた次の例を考えてみましょう。

例 31: `[cell]` セクションの例題 (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
```

この例題では半無限領域をつくる `PX,PY,PZ` の面を使用して、一辺が 12cm の立方体をつくっています。`PX,PY,PZ` はそれぞれ yz, xz, xy 平面に平行な面を定義する面記号であるため、例えば 5,6 行目で定義している面番号 11,12 の面は、それぞれ座標 (−6,0,0), (6,0,0) を通り yz 平面に平行な面となります。これらの方程式はそれぞれ $x + 6 = 0$, $x - 6 = 0$ ですから、原点 (0,0,0) を含む領域は、面番号 11 に対しては $0 + 6 = 6 > 0$ によりプラス側、面番号 12 に対しては $0 - 6 = -6 < 0$ よりマイナス側となります。よって、面番号 11 と 12 で囲まれた領域は “11 のプラス側かつ 12 のマイナス側” となり、演算子 ␣ (空白) を用いて 11␣−12 と記述します。面番号が 13~16 の面も同様ですから、x, y, z の 3 方向に関して閉じた空間を表現するには、2 行目の様に 6 つの面を ␣ でつなげた書き方をします。例題 (2) ではこの閉じた領域をセル番号 1 でかつボイドであるとして定義しています。図 40 に、この例題でつくられる領域を *xz* 平面で切った断面図として示しました。本来は立方体ですが、2 次元平面で見た結果正方形となっています。また、2 行目ではセルに掛かる演算子`#`を用いて外部ボイドを定義しています。これはセル番号 1 の領域以外の範囲を対象にするという意味になります。

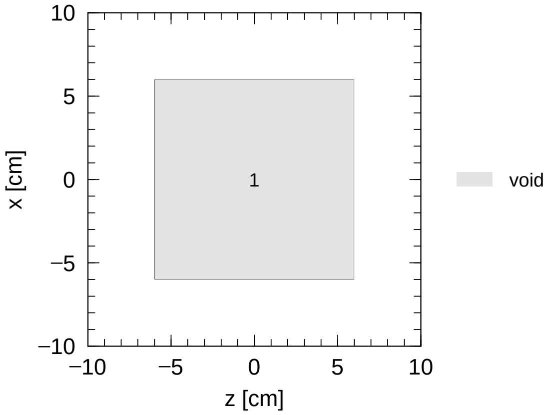


図 40: 例題 (2) の空間を *xz* 平面で切り取った断面図。セル番号 1 の領域はボイド (真空)。

次は ( ) と : の記号を使用した例です。例題 (1) で示した球と (2) の立方体を合わせた形状を考えています。

例 32: [cell] セクションの例題 (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
```

2 行目の ( ) で囲んだ部分が例題 (2) のセル番号 1 の領域に相当しており、これと例題 (1) の球面の内側を合わせた領域がこの例題ではセル番号 1 として定義されています。図 41 に示したのが例題 (3) の領域を *xz* 平面で切った断面図です。本来は立方体から球の一部が飛び出た形状をもつ立体ですが、2 次元平面では図の様になっています。本例題の様に、領域を足し合わせる場合に使用する演算子が : です。

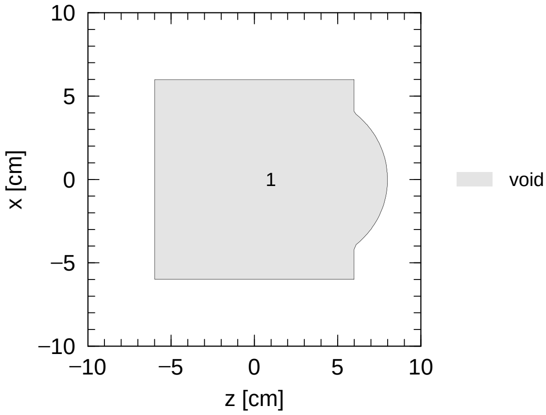


図 41: 例題 (3) の空間を *xz* 平面で切り取った断面図。セル番号 1 の領域はボイド (真空)。

例題 (4) では更に、立方体を球の内側と外側に分けた場合を考えます。

例 33: `[cell]` セクションの例題 (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
```

`[surface]` セクションは例題 (3) と同じです。5 行目で “立方体の内側でかつ球の外側” の領域を記号 ⊔ (空白) を用いて記述しており、これをセル番号 2 と定義しています。また、本例題では `[material]` セクションで “water” を物質番号 1 として定義しており、セル番号 2 の領域には粒子密度が 1.0×10^{24} atoms/cm^3 の “water” が満たされています。図 42 に結果を示します。ただし、*xz* 平面で切った断面図です。セル番号 1 は 4 行目で定義しており、この球の中はボイドとしています。外部ボイドの定義は 6 行目で行っており、セル番号 1 と 2 の領域を除いた他の全てが外部となっています。

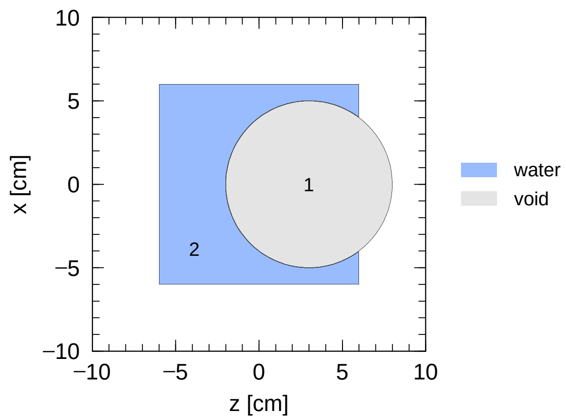


図 42: 例題 (4) の空間を *xz* 平面で切り取った断面図。セル番号 1 の領域はボイド (真空) とし、セル番号 2 の領域には水を満たした。

5.6.3 Universe 構造

PHITS では、セルパラメータ U を用いて、複数の宇宙 (universe) を定義できます。各宇宙に個別の仮想空間を構築することで、ある宇宙の一部として別の宇宙の内部構造を引用することが可能となります。この様に、多数の宇宙をそれぞれ定義し、相互に参照するための機能をここでは universe 構造と呼びます。ただし、実際に粒子輸送の舞台となる空間そのものは 1 つだけであり、この空間の一部を別の宇宙の内部構造で満たす (置き換える) 場合に universe 構造を利用します。単純に一部分だけを置き換える場合はあまり意味がありませんが、特に 5.6.5 節の繰り返し幾何形状を利用するときに有効となる機能です。

例題として、図 43 に示した 3 つの宇宙を組み合わせて仮想空間を構築する方法について考えます。図の (a) が実際に粒子輸送を行う空間で、一辺が 12 cm の立方体を *yz* 平面で分割した 2 つの直方体で構成されて

います。ただし、示した図は *xz* 平面で切った断面図です。図の (b) と (c) が、(a) とは違う内部構造をもつ universe の 1 と 2 で、座標の中心に半径 5 cm、高さ 12 cm の円柱を配置しています。これらは、内部構造は同じですが、各領域を構成する物質が違っています。Universe 1 では円柱内部に水が満たされ、その外はボイド（真空）となっているのに対し、universe 2 では円柱内部は鉄で、その外部は水で満たされています。

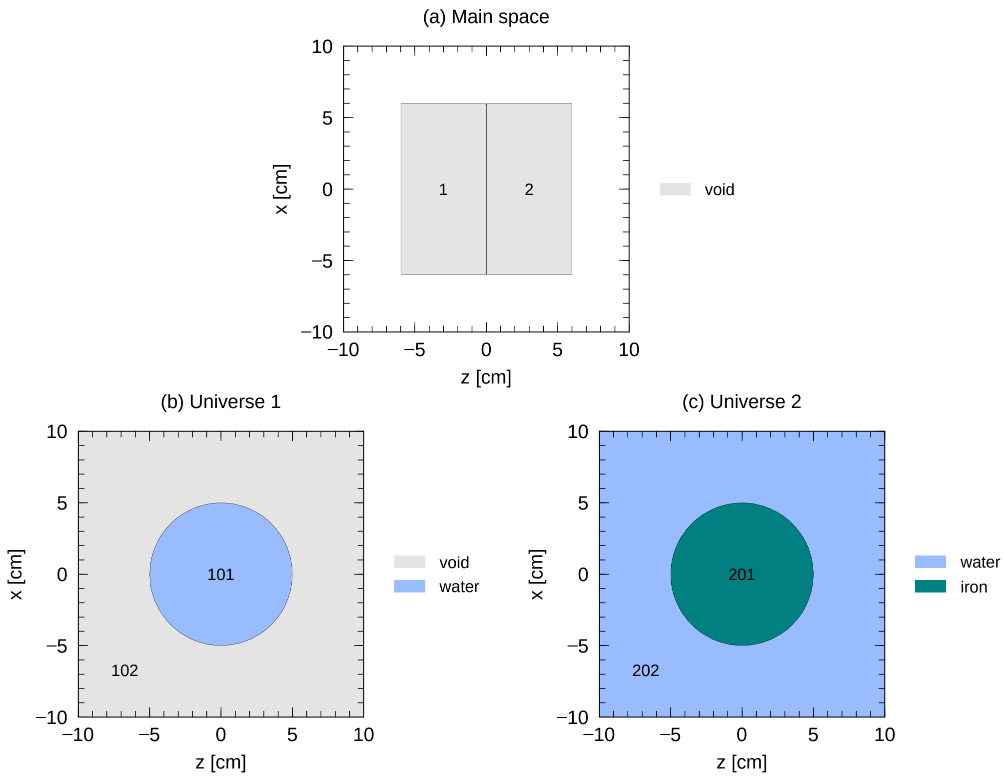


図 43: (a) 立方体を *yz* 平面で分割した 2 つの直方体。(b) 円柱内部に水を満たし、それ以外はボイドとした universe 1。(c) 鉄の円柱の周りを水で満たした universe 2。

(b) の universe 1 を (a) のセル番号 1 の領域に入れ、(c) の universe 2 をセル番号 2 の領域に入れた仮想空間を構築する例が以下の例題 (5) です。

例 34: [cell] セクションの例題 (5)

```
[ Material ]
  mat[1]  1H 2  16O 1
  mat[2]  Fe 1
[ Cell ]
    1  0          11  -12  13  -14  15  -17  FILL=1
    2  0          11  -12  13  -14  17  -16  FILL=2
  101  1  -1.0   -10  13  -14  U=1
  102  0        #101  U=1
  201  2 -10.0   -10  13  -14  U=2
  202  1  -1.0   #201  U=2
    9 -1        #1 #2
[ Surface ]
  10  CY   5
  11  PX  -6
  12  PX   6
  13  PY  -6
  14  PY   6
  15  PZ  -6
  16  PZ   6
  17  PZ   0
```

7,8 行目で universe 1 を、9,10 行目で universe 2 を定義しています。これらは共に面番号 10,13,14 を使って定義した円柱を中心に置いた宇宙となっています。Universe 1 では、セル番号 101 の円柱内部に物質番号 1 の水を入れ、その外部をボイドにしてセル番号 102 の領域と定義しています。Universe 2 では、円柱内部をセル番号 201 として鉄柱にし、セル番号 202 の外部には水を満たしています。7~10 行目の最後にある `U=1，2` がそのセルがどの宇宙に属するかを指定するパラメータです。5, 6 行目で、それぞれ図 43(a) のセル番号 1,2 を定義しており、最後の `FILL=1，2` によってそのセルをどの universe によって満たすか (置き換えるか) を決定します。この例題の結果を示したのが図 44 です。*xz* 平面の断面図ですが、セル番号 1 の領域が universe 1 に、セル番号 2 が universe 2 に置き換えられているのがわかります。

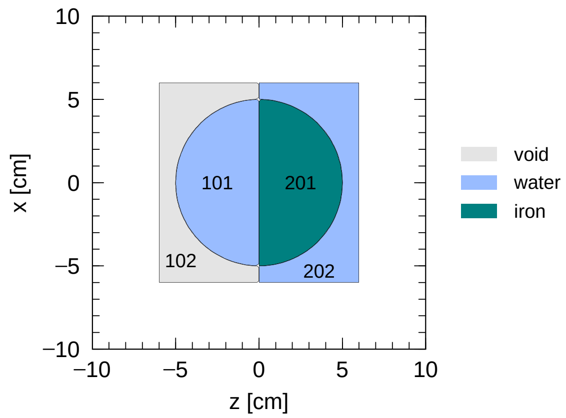


図 44: 例題 (5) の空間を *xz* 平面で切り取った断面図。セル番号 1 の領域に universe 1 のセル番号 101 と 102 が、セル番号 2 の領域に universe 2 のセル番号 201 と 202 が入っている。

Universe 構造を利用する際の注意点として、未定義の領域を引用しないことと、座標系が全ての宇宙で一致していることが挙げられます。例えば前者については、例題 (5) の 8 行目でセル番号 102 の領域をボイドとして定義する必要があり、これがないと 5 行目で引用した際にセル番号 102 の部分を適切に設定できません。各 universe の全ての領域を定義する必要はありませんが、引用する領域については何らかの物質 (かボイド) を指定しておかなければなりません。またその際、置き換える元の領域よりも引用する領域を少し広く定義しておかないと、エラーが出る場合がありますのでご注意ください。次に、後者については、座標原点の位置、x, y, z 軸の方向、空間のスケールがどの universe でも同じです。これは同じ `[surface]` セクションで定義した面を利用しているためです。別の宇宙を引用する場合は、対応する座標を確認してください。例えば、例題 (5) の 14,15 行目で `PX` の値を変えると、図 45 の様に円柱の一部が立方体の中に入らなくなります。

5.6.4 Lattice 構造

本節では、lattice 構造の仕組みと定義の仕方、および基本的な利用方法について説明します。この機能は似た形状を多数繰り返す場合に非常に有効です。より実際的な設定方法については 5.6.5 節をご覧ください。

Lattice 構造は、図 46 に示す四角形と六角形の 2 種類の形状を基本単位として、格子状に繰り返し並べる機能です。`U` パラメータで指定して、基本格子を無限に並べた宇宙をつくります。ただし、その内部は別の `U` パラメータを用いて定義した宇宙で満たす必要があります。パラメータ `LAT` を用いて、四角形は `LAT=1`、六角形は `LAT=2` として定義できます。図 46 に示した図形の各辺に付いた数字は各々の図形を定義する際の面の順番を意味しており、`[surface]` セクションで定義した面番号を図の数字の順に並べます。1 と 2、3 と 4、5 と 6 の面はそれぞれが平行な面として定義されている必要があります。もちろん、プラス側・マイ

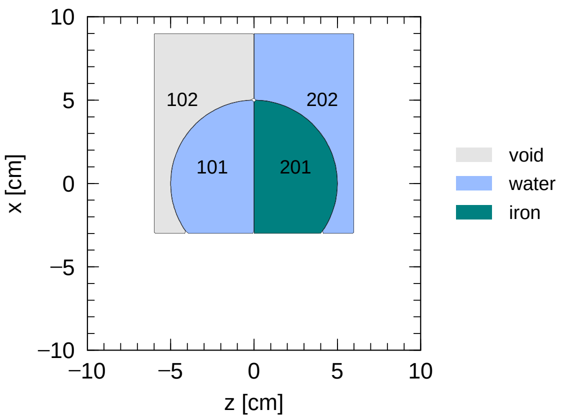


図 45: 例題 (5) において、セル番号 1 と 2 の範囲を x 方向に関してずらした場合。

ナス側のどちらであるかも指定します。ただし、1 と 2 の長さと 3 と 4 の長さを変えることは可能で、正方形や正六角形でなくても構いません。

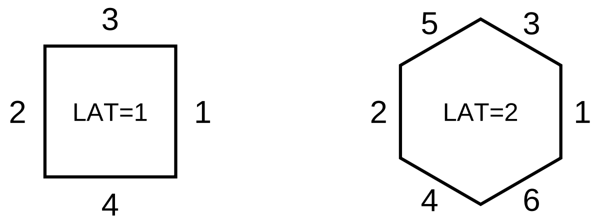


図 46: Lattice 構造の基本単位として使用できる四角形 (左) と六角形 (右)。各辺の番号は、LAT パラメータを定義する際の面の順番。

はじめに、四角形 (LAT=1) の lattice 構造を利用した例を考えます。

例 35: [cell] セクションの例題 (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
```

5 行目で lattice の基本格子を定義しています。LAT=1 で扱うことができる形状は四角形ですが、3 次元の形状を考えると実際には四角柱となります。図 46 で示した様に、定義には最低 4 面必要で、4 面の場合は高

さが無限大の四角柱となります。したがって、本例題で定義しているのは断面が一辺 4cm の正方形となる無限の高さをもつ四角柱となり、これが y 軸に関して平行かつ隙間無く並んでいる宇宙を universe 1 としています。この格子状の集まり全てがセル番号 101 の領域です。これとは別に 6 行目において、面記号 `BOX` を用いて一辺 20cm の立方体が原点にある宇宙を定義しており、これは universe 2 としています。このセル番号 201 の立方体には水が入っており、`FILL=2` を用いることにより、セル番号 101 中の格子一つ一つが水で満たされることになります。4 行目でセル番号 1 を universe 1 で満たした領域として定義しており、これが一辺 12cm の立方体であることから、本例題の仮想空間は図 47 の様になります。左図が 3 次元的に見た立体図で、右図がこれを xz 平面で切り取った断面図です。図を見てわかる様に、同じ四角柱が全部で 9 つ並んでいます。これらは、universe 1 においては高さが無限大ですが、セル番号 1 の領域が y 軸方向にも制限されているため、結果的に有限の高さをもつ四角柱となっています。もし、有限の高さをもった四角柱を lattice の基本単位としたい場合は、例えば面番号 23, 24 による領域の制限を 5 行目で加える必要があります。

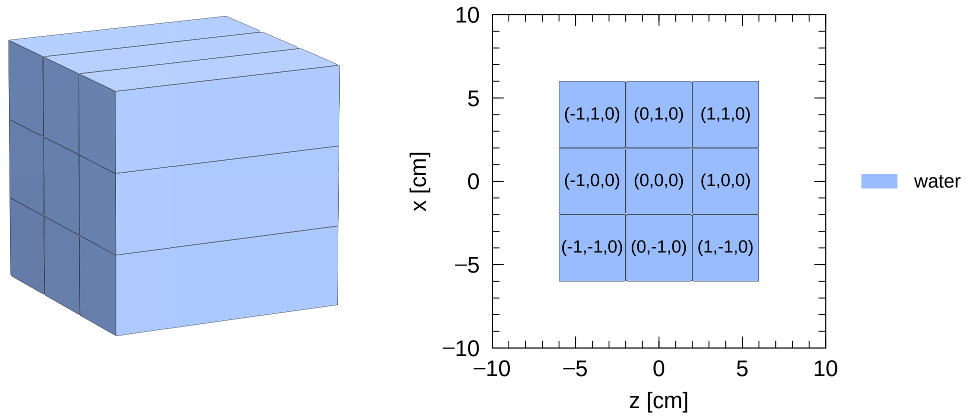


図 47: 例題 (6) の空間を立体的に見た図 (左) とこれを xz 平面で切り取った断面図 (右)。各格子は、中に示された座標によって区別される。

図 47 の右図で示した様に、各円柱には lattice 座標 (s, t, u) がふられており、各々の領域がセル番号 101 の中のどれであるかを区別できるようになっています。ただし、この座標は必ずしも通常の座標 (x, y, z) とは対応しておらず、単位格子を定義する際の面の順番で決まります。すなわち、図 46 の左図において、辺 2→1 の方向が s、4→3 が t の方向となります。残りの u の向きは、s, t と同様にセル定義文の順番で決定されます。例題 (6) の 5 行目では、z 軸方向、x 軸方向の順番で定義しているため、s は z 軸、t は x 軸に対応付けがなされています。実際に図 47 の右図でも、z が増える方向に s が増加し、x が増える方向に t の値が増えていることがわかります。各タリーにおいて `mesh=reg` により領域指定する場合は、`(201 < 101[-1 0 0] < 1)` の書式を用います。これは “セル番号 1 を満たすセル番号 101 の lattice 座標 $(-1, 0, 0)$ に含まれているセル番号 201 の領域” という意味です。この書式に関しては 6.1.2 節をご参照ください。なお lattice 座標 (s, t, u) は、`[t-gshow]` タリーにおいて `output=7 or 8` を指定することで表示させることができます。

次の例題は六角形 (`LAT=2`) の lattice 構造を利用した場合です。

例 36: `[cell]` セクションの例題 (7)

```
[ Material ]
  mat[1]  1H 2  16O 1
[ Cell ]
    1   0         11 -12  13 -14  15 -16  FILL=1
  101   0        -31  32 -33  34 -35  36 -24  23  LAT=2  U=1  FILL=2
  201   1  -1.0  -90  U=2
    2  -1        #1
[ 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
```

17~23 行目で正六角形をつくるための面の定義を行っています。これらを組み合わせると、xz 平面で見た場合に正六角形となり、y 軸に垂直な六角柱となります。ここで、正六角形の一辺の長さを 2cm とし、これを `c1` として定数定義しています。また、組み込み定数である `pi`(円周率 π) および面記号 `PZ`，`P` を用いて、60 度ずつ回転した平面を定義しています。Lattice 構造の基本格子は 5 行目で定義しており、セル定義文中の面番号の順番は図 46 の右図にある数字の通りとなっています。ただし、最後に面番号 23, 24 による y 軸方向の制限を加えています。したがって本例題では、高さが 4cm、一辺が 2cm の六角柱を無限に並べた宇宙が universe 1 となります。各六角柱には 6 行目で定義した universe 2 が入っており、実質的には水で満たされたセル番号 201 が入っていることになります。4 行目で、先の例題と同様に、セル番号 1 の領域を universe 1 で満たしており、本例題の結果は図 48 となります。左図が立体図、右図がこれを xz 平面で切った断面図です。左図からわかる様に、六角柱が詰められた universe 1 から立方体の領域だけを切り取った形状になっています。また、y 軸方向にも 3 列が並んでおりますが、その高さは揃えられており、いわゆる 3 次元のハニカム構造はつくりません。右図では各領域に lattice 座標 (s, t, u) が付けられています。図を見てわかる様に、s は左から右、t は左下から右上に向かって増加しており、これらはやはり 5 行目のセル定義文のところに並べる順番に依存します。図 46 の右図において、辺 2→1 の方向が s、4→3 が t の方向となります。u についても同様で、面番号 24→23 の順番であれば y 軸方向に、23→24 であれば $-y$ 軸方向に u の値は増加します。各タリーにおいて `mesh=reg` により領域指定する場合は、`(201 < 101[-2 0 0] < 1)` の書式を用います。これは “セル番号 1 を満たしたセル番号 101 の lattice 座標 $(-2, 0, 0)$ に含まれているセル番号 201 の領域” という意味です。この書式の詳細は 6.1.2 節をご覧ください。なお lattice 座標 (s, t, u) は、`[t-gshow]` タリーにおいて `output=7 or 8` を指定することで表示させることができます。

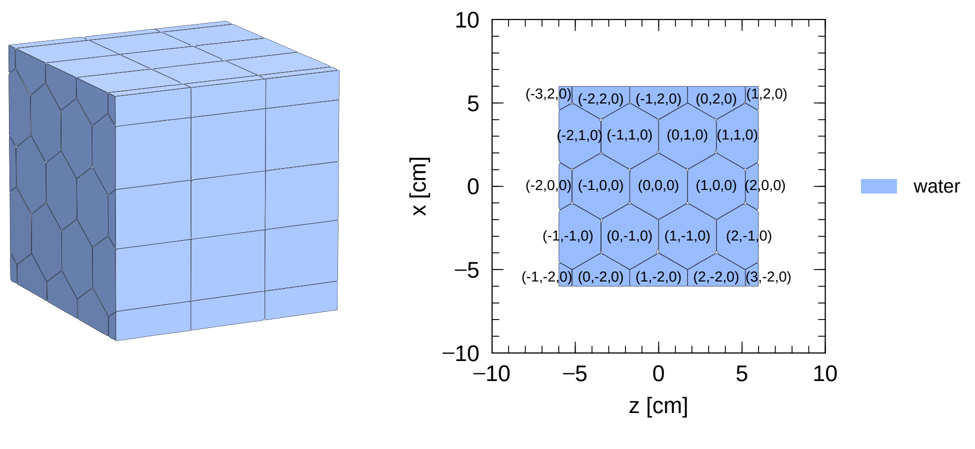


図 48: 例題 (7) の空間を立体的に見た図 (左) とこれを xz 平面で切り取った断面図 (右)。各格子は、中に示された座標によって区別される。

5.6.5 繰り返し幾何形状

多数の同じ、あるいは少し違うだけの構造のセルを並べる場合に、当然 1 つ 1 つのセルを定義することは可能ですが、基本となるセルを 1 つ用意しこれを繰り返すことで効率良く設定できます。5.6.4 節の lattice 構造の利用もその 1 つであり、他には `LIKE` *n* `BUT` セルパラメータ の表式も利用できます。

5.6.5.1 LIKE *n* BUT セルパラメータ

セル番号 *n* のセルと同様であるが、`BUT` 以下の内容だけが違うセルを定義する方法です。`BUT` の後に続くセルパラメータには表 86 に挙げたものがあります。以下の例題では、`TRCL` と `MAT` を使用した場合を考えています。

例 37: `[cell]` セクションの例題 (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
```

7, 8 行目で `LIKE` *n* `BUT` セルパラメータの表式を用いてセルを定義しています。この元となるセル番号 2 は 6 行目で定義しており、中に水が入った一辺 4cm の立方体です。セル番号 2 は原点を中心として配置されていますが、これを 19 行目で定義した座標変換番号 1 で座標変換したのがセル番号 3 です。座標変換番号 1 は、*x* 軸方向に 3cm、*z* 軸方向に −5cm 平行移動する変換ですから、*xz* 平面上で見ると図 49 の様になります。この図は本例題の結果を *xz* 平面で切り取った断面を示しています。次のセル番号 4 は `TRCL` と `MAT` のセルパラメータを使用しており、座標変換番号 2 による平行移動 (および回転) を行い、その上で物質番号 2 に中の物質を変更しています。回転は *xz* 平面上で 30 度行っており、図 49 でも確認できます。5 行目では、セル番号 1 を半径 10cm、高さ 4cm の円柱と定義しています。ただし、セル番号 2, 3, 4 の領域が含まれない様に、#を用いてこれらを除いています。また、これに伴い外部ボイドの定義も 9 行目に示す様な書き方が必要です。単純に #1 のみではセル番号 2, 3, 4 の領域も外部ボイドとなってしまいます。

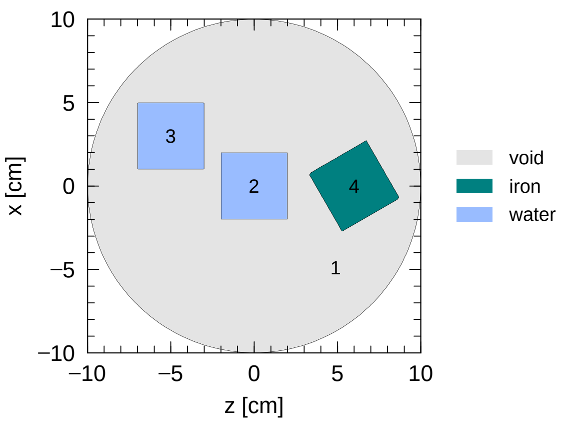


図 49: 例題 (8) の空間を *xz* 平面で切り取った断面図。セル番号 1 の円柱の中にセル番号 2, 3, 4 が入っている。

5.6.5.2 Lattice を用いた階層構造

5.6.4 節で見た四角柱や六角柱の lattice をより複雑なものにするために、universe を用いて階層構造を構築することができます。考え方は universe 構造に基づいており、universe 1 の一部を universe 2 で満たし、更に universe 2 の一部を universe 3 で満たし、といった手続きで各階層を定義していくことが可能です。PHITS では基本的に最大 10 (`param.inc` 中の `mxlv` で変更できます) の階層構造をつくることができます。

例 38: `[cell]` セクションの例題 (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 -10.0  -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
```

まず 5, 6 行目にある、四角柱の lattice 構造とそれを引用するセル番号 1 の定義は例題 (6) と同じです。ただし、7, 8 行目におけるセルパラメータ `FILL` の使い方が違っており、ここでは各格子の状態を個別に指定しています。7 行目では取り扱う範囲を決めており、各格子に割り振られた lattice 座標 (s, t, u) を参照して `FILL=`の後ろで決定しています。本例題では、s に関して -1 から 1 まで (3 つ)、t に関して -1 から 1 まで (3 つ)、u に関して 0 から 0 まで (実質は 1 つ) をセル番号 101 の lattice 構造と定義しています。合計 $3 \times 3 \times 1 = 9$ の格子が対象となり、それぞれを満たす宇宙は 8 行目で指定しています。この順番は lattice 座標 (s, t, u) で表現して、$(-1, -1, 0), (0, -1, 0), (1, -1, 0), (-1, 0, 0), \cdots$ の様になっており、2 であれば universe 2 で、3 であれば universe 3 でその格子を満たすことを意味します。

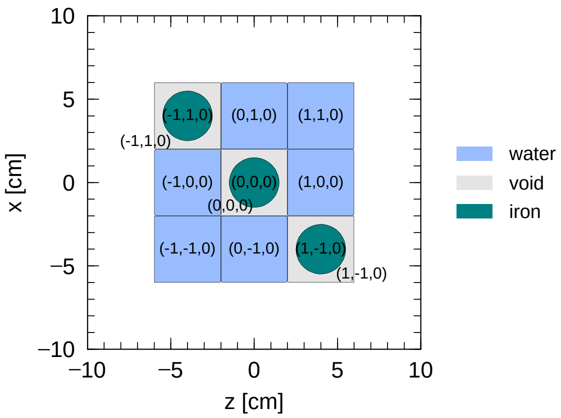


図 50: 例題 (9) の空間を xz 平面で切り取った断面図。9 つの四角柱の内、3 つが鉄柱を中心に配した内部構造をもっている。

Universe 2 を水が入った一辺 20cm の立方体を原点付近に配置した宇宙として、universe 3 を原点に半径 1.5cm の鉄柱を置いた宇宙として定義しており、9 つの四角柱の内 3 つが特別な構造をもつことになります。結果を示したのが図 50 で、xz 平面に関する断面図です。この図から $(s, t, u) = (1, -1, 0), (0, 0, 0), (-1, 1, 0)$ の格子が内部構造をもっていることがわかります。各タリーにおいて `mesh=reg` により領域指定する場合は、`(302 < 101[0 0 0] < 1)` の書式を用います。これは “セル番号 1 を満たしたセル番号 101 の lattice 座標 (0,0,0)

に含まれているセル番号 302 の領域” という意味で、図 50 では中心の四角柱にある鉄柱周辺のボイドの領域を指します。この様な書式については 6.1.2 節を参照してください。

次は、階層構造をより複雑にした例です。

例 39: `[cell]` セクションの例題 (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
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
```

この例題でつくられる仮想空間は、図 51 の様になります。ただし、*xz* 平面で切った断面を示しています。まず、一辺 12cm の立方体を 9 つの四角柱に分け、その内の 3 つに内部構造をもたせた点は例題 (9) と同じです。違うのは universe 3 として定義した宇宙を更に内部構造をもった格子で埋めたことで、9~11 行目でその状態を指定しています。ここでは $2 \times 2 \times 1 = 4$ の四角柱を定義しており、その内の 2 つが鉄柱をもつ格子となっています。また、8 行目の `(1 0 1)` は座標変換を表しており、引用する際に universe 3 を *x* 軸方向に 1cm、z 軸方向に 1cm 平行移動しなさい、という意味となります。セル番号 301 は、原点 (0,0,0) を中心にもつ四角柱を基本単位としているため、図の様な配置にする場合は少しずらす必要があります。また、lattice を用いて階層構造を作成した場合の `mesh=reg` による領域指定は、階層構造を増やした書式により行います。例えば図 51 において $x = -1, z = -1$ を中心とする微小四角柱のボイド部分は、`(402 < 301[-1 -1 0] < 101[0 0 0] < 1)` と記述します。もし対象とする階層が更に深くなった場合は、<と lattice 座標を用いて階層を増やすことで各領域を指定できます。この書式の詳細につ

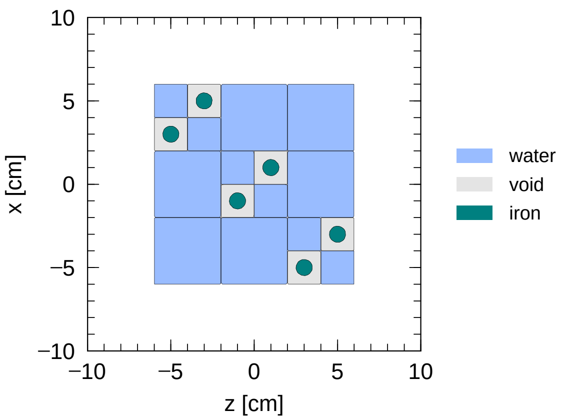


図 51: 例題 (10) の空間を *xz* 平面で切り取った断面図。9 つの四角柱の内 3 つが構造をもち、更に 4 つの四角柱で構成されている。

いては 6.1.2 節をご覧ください。

5.6.5.3 ボクセルファントム (voxel phantom) の利用

PHITS では lattice 構造を利用してボクセルファントム (voxel phantom) を仮想空間として設定できます。ボクセルファントムとは微小立方体を積み重ねて生物などの複雑な構造物を表現したものです。CT 等の画像データを元にして作成するボクセルデータを使用します。本節では、非常に簡単なボクセルデータを用意し、これを用いた設定方法について説明します。

基本的な考え方として、まず粒子輸送を行うためのある一定の大きさをもつ立方体 (直方体) を用意し、その中に微小立方体 (ボクセル) を並べます。その際、lattice 構造の `LAT=1` を利用し、各ボクセルの座標が指定できる様にセルパラメータ `FILL` を使います。`FILL` を使用した場合、各ボクセルがどの状態か、すなわちどの universe で満たされているか、が指定できるため、水や鉄 (あるいはボイド) で満たされた universe を引用することで、様々な構造物をつくることができます。したがって、ボクセルデータは各ボクセルの位置座標と構成物質の情報をもっている必要があり、ここでは $L_{000}\ L_{100}\ L_{200}\ \cdots\ L_{stu}\ \cdots$ の順番で並べたデータ群を想定しています。ここで、L_{stu} は lattice 座標 (s, t, u) にあるボクセルの構成物質 (universe 番号) です。

以下では、一辺が 10cm の立方体の中に一辺 2cm のボクセルを合計 $5 \times 5 \times 5 = 125$ 並べた場合を考えています。

例 40: `[cell]` セクションの例題 (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
```

5 行目でセル番号 1 の領域を一辺が 10cm の立方体として定義しています。これが次の行のセル番号 101 の lattice で満たされており、基本単位は面番号 20 で定義した一辺 2cm の立方体です。ここでは面記号 `BOX` を用いて微小立方体 (ボクセル) を定義しています。7 行目でボクセルを lattice 座標 (s, t, u) のそれぞれに 5 つずつ並べるとしており、8~12 行目で各ボクセルを満たす universe 番号を指定しています。ボクセルは $(-2, -2, -2), (-1, -2, -2), \cdots, (2, 2, 2)$ の順番で並んでおり、ボイドの場合は 2、鉄は 3、水は 4 の数字が対応しています。ボイドや鉄、水が入った空間の定義は 13~15 行目で行っており、それぞれが一辺 20cm の立方体に入った宇宙を universe 2, 3, 4 としています。本例題のボクセルで表現しているのは多少いびつな形状の鉄の箱に水が入っている構造物です。結果を図 52 に示しました。左図が立体的にそのまま見た結果、右図が表面の一部を透明にして見た結果です。少しずれていますが、図の左から右が s 軸、下から上が t 軸、奥

から手前が u 軸の方向に対応します。見てわかる様に、でこぼこした鉄の箱の中には水が入っており、この構造物を対象に粒子輸送計算を行うことができます。各タリーにおいて `mesh=reg` により領域指定する場合は、`(401 < 101[0 0 0] < 1)` の書式を用います。あるいは、セル番号 1 中の水の部分全てを指定したい場合は、`(401< 1)` とすることができます。これらの書式については 6.1.2 節をご覧ください。ただし、`(301 < 101[-2:2 -2:2 -2:2] < 1)` の様な指定方法はできません。セル番号 `101[-2:2 -2:2 -2:2]` の全てにセル番号 301 の領域が含まれているわけではないためです。

図 52: 例題 (11) の構造物を立体的に見た図 (左) と表面の一部を取り除き内部を見た図 (右)。

例 40 の 8 から 12 行目の universe の羅列について、バージョン 3.09 以降では短縮形式での記述が可能になりました。ボクセルファントムの様なデータでは、同じ universe 番号が連続することが多いため、この短縮形式を利用することで、大幅に記述を省略することができます。これにより、膨大なボクセルデータに対するファイルサイズの削減や読み書きに掛かる計算時間の短縮につながります。短縮形式では、連続する universe 番号について、一つ目の universe 番号はこれまでどおり記述し、二つ目には universe 番号ではなく、これ以降に同じ universe 番号が連続する数を負の符号を付けて記述します。これにより、例 40 の 8 から 12 行目は以下のように記述することができます。

例 41: `[cell]` セクションの例題 (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
```

例 40 では、universe 番号の 2 が 12 個並んだ後に、universe 番号の 3 が記述されているのに対し、例 41 では、universe 番号の 2 の後に同じ universe 番号がさらに 11 個並んでいるとの意味で、-11 の記述がされています。

PHITS では、最初に読み取ったインプットデータを一時的にバイナリ形式で書き出し、それを再度読み込んで粒子輸送計算を行います。例題 (11) で扱ったのは簡単なボクセルですが、一般的にボクセルデータは大規模なものとなり、それを PHITS の実行の度に読み書きするのは非常に時間がかかります。これを緩和するために、`[parameters]` セクションの `ivoxel` パラメータを使う方法があります。`ivoxel` は、ボクセルデータ (実質的には例題 (11) の 8 から 12 行目) をバイナリとして出力させて保存し、2 度目からはそのバイナリを読んで計算するという機能をもちます。PHITS 実行時に `ivoxel=2` とすると `file(18)` にデータを書き出してそのまま終了します (ただし、バージョン 2.30 以前では、終了せずに引き続き輸送計算を行います) ので、以降 `ivoxel=1` と設定することで該当ファイルからバイナリデータを読み込んで計算できます。また、大量のボクセルデータを PHITS のインプットファイルとは別のファイルに保存して計算するときは、`infl` を使用します。この場合の `ivoxel` の使い方も同じです。

5.6.5.4 連続四面体形状の利用

本節では、連続四面体形状の仕組みと定義の仕方、および基本的な利用方法について説明します。連続四面体形状はポリゴンの一種で、様々な大きさの四面体を組み合わせることで、複雑な形状をもつ物体を表現できるようになります。

PHITS では二種類の連続四面体形状の形式に対応しています。一つは四面体メッシュ分割ソフトウェア TetGen によって生成される Node ファイルと Element ファイルで四面体形状を表現する形式で、もう一つは有限要素法 CAE ソフトウェア NASTRAN のバルクデータ形式です。

TetGen での形式は、

- Node ファイル: 各四面体の各々の頂点の xyz 座標
- Element ファイル: 各四面体要素を構成する 4 つの頂点

の 2 つのファイルで表現されます。Node ファイルおよび Element ファイルは、共通のファイル名に.node および.ele という拡張子が付いた形を想定しています。PHITS のインプットファイルでは、この共通ファイル名を指定することになります。この機能の具体的な利用方法については、`\phits\utility\TetraGEOM` フォルダにある資料やサンプルインプットをご参照ください。

次に、Node ファイルおよび Element ファイルの例を示します。

例 42: Node ファイルの例

```
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
```

Node ファイルでは、まず始めにこのファイルに含まれる全ての頂点の数と扱っている次元数を設定します (2 行目)。PHITS で扱える次元は 3 次元のみです。2 行目の最後の 2 つの 0 は PHITS では使われずに無視されます。その下に (4 行目以下)、

```
[頂点番号] [x] [y] [z]
```

の順で頂点番号とその頂点の xyz 座標を指定します。

例 43: Element ファイルの例

```
1:   # Simple two elements
2:   2  4  1
3:   # elementID point(1:4) universe
4:   1      1  2  3  5    5001
5:   2      5  2  3  4    5002
```

Element ファイルでは、まず始めにこのファイルに含まれる全要素数と 1 つの要素を構成する頂点の数（PHITS では、1 次要素の四面体しか扱えないので、4 となります）と要素に付加する情報の数（PHITS では 1）を設定します (2 行目)。その下に (4 行目以下)、

```
[要素番号] [頂点番号 1] [頂点番号 2] [頂点番号 3] [頂点番号 4] [要素 universe 番号]
```

の順で各要素の情報を指定します。要素番号は各四面体の ID 番号、頂点番号 1~4 はその四面体を構成する頂点の番号で Node ファイル内で定義した番号です。最後の要素 universe 番号で、各四面体をどの universe 番号の空間で満たすかを指定します。各 universe の定義は PHITS のインプットファイルで行います (例 44 の 13,14 行目参照)。なお、Node ファイルおよび Element ファイル内では、任意の行で#から始まるコメント行を挿入することができます。

Node ファイルおよび Element ファイルは四面体メッシュ分割ソフトウェア TetGen を使って作成することができます。多少技術が必要ですが、一般的なポリゴンデータもこのソフトウェアを使って連続四面体に変換することが可能です。TetGen は次のウェブサイト `http://wias-berlin.de/software/tetgen/` から無料でダウンロード可能です。TetGen の使い方については、TetGen のマニュアルをご参照ください。

NASTRAN のバルクデータ形式は、構造解析計算や流体解析計算など様々なソフトウェアで一般的に出力可能な形式です。PHITS ではバルクデータ形式の中で、頂点情報 GRID と四面体ソリッド要素情報 CTETRA で構成される BDF ファイル（拡張子.bdf）を読み込むことができます。通常、何らかのソフトウェアにより体系を作成し、それをバルクデータ形式で出力することが想定されることから、バルクデータ形式に関する詳細は省略します。詳しくは NASTRAN バルクデータで検索してください。構造計算や流体計算と同一の形式を採用することで、これらの計算と放射線輸送計算をシームレスに繋いだ解析が可能となりました。具体的な利用方法については、`\phits\utility\FLUENT` フォルダにある資料をご参照ください。

次に、PHITS のインプットファイルにおける指定の仕方について説明します。

例 44: `[cell]` セクションの例題 (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=5000
14:   202   3 -7.874    -90  U=5001
```

例 45: `[cell]` セクションの例題 (13)

```
10:   101   1 -0.001205 -20  U=1  LAT=3  nfile=Tetra  TSFAC=1.0
```

PHITS で連続四面体形状を取り扱うためには、直方体（面記号 RPP）で指定する領域を定義し、その中に四面体連続形状を展開します。全ての四面体の頂点はこの直方体の中に含まれる必要があります。ただし、直方体の大きさを連続四面体形状の大きさからかけ離れて大きく取り過ぎると計算の都合で問題が生じるので、連続四面体形状が全て収まる適当な大きさの直方体としてください。10 行目では、セル番号 101 の領域として、面番号 20 で定義される 14cm×10cm×14cm の直方体の中にファイル名 Tetra で指定される連続四面体形状を展開しています。LAT=3 は連続四面体形状を使用するオプションの指定で、TFILE に与えられるファイル名に対応する Node ファイルおよび Element ファイルが読み込まれることになります。NASTRAN バルクデータ形式のファイ

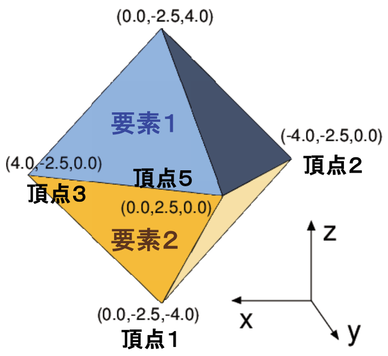


図 53: 例題 (12) で作られる連続四面体形状。

ルを使用する場合は、45 の様に `TFILE` の代わりに `NFILE` でファイル名を指定してください。ファイル名に拡張子.bdf がついたファイルが読み込まれます。Windows 以外の OS では、ファイル名に大文字小文字の区別があるので、ファイル名の指定はどちらの形式でも大文字小文字を正しく入力してください。`TSFAC` の指定により、Node ファイルで与えられる四面体頂点の座標に対する規格化定数を与えることができます。ここでは、`TSFAC=1.0` なので、Node ファイルで与えられる座標をそのまま使用します。セル番号 101 の領域に対するマテリアルの指定は領域 101 内で連続四面体を除く領域で使用されることになります。11 行目でセル番号 1 の領域を 10cm×6cm×10cm の直方体として定義しています。`Fill` の指定により、この領域が 10 行目のセル番号 101 で定義される連続四面体を含む領域によって満たされることになります。このように連続四面体形状 (`LAT=3`) を使用する場合も、Lattice 構造の指定と同じように、Universe と Fill を利用した入れ子構造を定義してください。Element ファイルで指定される各四面体の要素 universe に対応する領域の指定は Lattice 構造での指定と同様で 13,14 行目で行っています。この例題では、四面体要素 1 には物質番号 2 の水 (H_2O) が、四面体要素 2 には物質番号 3 の鉄 (Fe) が含まれることになります。

連続四面体形状の構成物質が増えた場合は、各四面体の要素 universe に対応する領域の数が増えます。この手作業の負担を減らすため、自動的にこの領域を作成するオプションを追加しました。この自動領域追加オプションを使用するには、パラメータセクションで `itetauto=1` と指定してください。連続四面体形状に含まれる全ての universe に対応する領域を自動で作成します。この際に、面番号 5000 番の非常に大きな球面とセル番号 5001 番から 5000+n 番を新たに追加します。ここで、n は連続四面体形状に含まれる univese の数です。ですので、このオプションを使う際にはこれらの面番号およびセル番号を使用しないようにしてください。NASTRAN バルクデータ形式ファイルを使用する場合は、バルクデータに含まれる PSOLID タブおよび MAT タブの情報を読み取り、自動的に追加領域の密度が設定されます。TetGen 形式ファイルを使用する場合は、これらの情報を外部ファイルで与える必要があります。同じファイル名で拡張子.txt のファイルを用意し、46 の様に各 universe 番号と密度を一行毎に記述してください。追加されたセル番号 5001 番から 5000+n 番の領域には自動的にセル番号と同じマテリアル番号が割り振られます。対応するマテリアルが存在しない状態で一度 PHITS を実行すると 47 の様なエラーメッセージが表示されますので、このメッセージを参考に連続四面体形状の universe に対応するマテリアルを 5001 番から 5000+n 番に定義してください。phits.out に含まれるインプットエコーに追加された面および領域が記述されるので、正しく指定が行えたかどうかを確認することができます。

例 46: txt ファイルの例

```
5000 -1.0
5001 -7.874
```

例 47: エラーメッセージの例

```
*** ERROR : undefined material
 TETRA material (MID):  5001
   should be defined as material ID number:  5001
*** ERROR : undefined material
 TETRA material (MID):  5002
   should be defined as material ID number:  5002
```

5.7 [Transform] セクション

5.7.1 書式

このセクションでは、[source], [surface], [cell] の各セクション、r-z, xyz メッシュの場合の各タリー、磁場や電場の定義の際に使用できる座標変換の定義を行います。コメント文字として$, C が利用できます。infl:によるファイルの挿入、set:によるユーザー定義定数の設定は、セクションの途中でも行うことができます。なお、[transform] を用いて可能な座標変換は回転と平行移動のみであり、拡大・縮小はできない点にご注意ください。

書式は以下の通りです。

```
[ Transform ]
    TRn   O1   O2   O3   B1   B2   B3   B4   B5   B6   B7   B8   B9   M
```

$*$TRn とすると、回転行列の成分 B_i を degree 単位の角度で指定できます。

表 87: 座標変換パラメータ

項目	説明
n	座標変換番号。1 ~ 999999 が使用できます。
O_1, O_2, O_3	平行移動の x, y, z 成分。
B_1 ~ B_9	回転行列の各成分。具体的には、式 (19)~(22) をご参照ください。行列式が 1 でない場合は、1 となるように規格化されます。
M	変換方法のオプション。 = 1 の場合、回転させた後に平行移動。 = −1 の場合、平行移動させた後に回転。 = ±2 は、B_1, B_2, B_3 にそれぞれ z, y, x 軸周りの回転角度のみを指定するオプションです。回転は、z, y, x 軸の順番で行います。指定する角度は、TRn の場合は radian 単位、$*$TRn の場合は degree 単位です。M = 2 と −2 の座標変換の定義は、それぞれ M = 1 と −1 と同じです。

デフォルト値は以下の通りです。

```
    TRn   0  0  0   1  0  0  0  1  0  0  0  1   1
```

また、M = ±2 の場合は、次のように入力してください。

```
    TRn   x0  y0  z0   θz  θy  θx   0  0  0  0  0  0   2
```

ここで、x_0, y_0, z_0 はそれぞれ平行移動の x, y, z 成分、$\theta_z, \theta_y, \theta_x$ はそれぞれ z, y, x 軸周りの回転角度です。B_4-B_9 の 0 の部分にはどんな数字でも良いので入れておく必要があります。

5.7.2 座標変換の定義

座標変換前の座標を (x, y, z)、変換後の座標を (x', y', z') とすると、タリーに関するもの以外は次の定義により変換します。$M = 1$ の時は、

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_4 & B_7 \\ B_2 & B_5 & B_8 \\ B_3 & B_6 & B_9 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} \tag{19}$$

となり、$M = -1$ の時は、

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_4 & B_7 \\ B_2 & B_5 & B_8 \\ B_3 & B_6 & B_9 \end{pmatrix} \begin{pmatrix} x - O_1 \\ y - O_2 \\ z - O_3 \end{pmatrix} \tag{20}$$

です。変換を行う対象を、$M = 1$ の場合は回転してから平行移動させ、$M = -1$ の場合は平行移動してから回転させせます。回転は、どちらの場合も、xyz 座標系の原点に対して行われます。ただし、平行移動は $M = 1$ と -1 で逆向きとなるのでご注意ください。

タリーに関しては、変換前の座標を (x, y, z)、変換後の座標、すなわちタリーで表示する座標、を (x', y', z') とすると、次の変換式をとります。$M = 1$ の時は、

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_2 & B_3 \\ B_4 & B_5 & B_6 \\ B_7 & B_8 & B_9 \end{pmatrix} \begin{pmatrix} x - O_1 \\ y - O_2 \\ z - O_3 \end{pmatrix} \tag{21}$$

となり、$M = -1$ の時は、

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_2 & B_3 \\ B_4 & B_5 & B_6 \\ B_7 & B_8 & B_9 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} \tag{22}$$

となります。タリーの場合は変換後の座標系で結果を描画することから、タリー以外の場合とは変換前後の座標系が式の上では逆になります。

$M = 2, -2$ はそれぞれ $M = 1, -1$ と同じ変換式を使用しますが、パラメータの入力方法を簡略化しています。

5.7.3 [transform] の例題

例 48: `[transform]` セクションの例題 (1)

```
[ Transform ]
set: c10[90]  $ angle of around Z (degree)
set: c20[30]  $ angle of around Y (degree)
set: c30[0]   $ angle of around X (degree)

tr1   0 0 0
       cos(c10/180*pi)*cos(c20/180*pi)
       sin(c10/180*pi)*cos(c30/180*pi)+cos(c10/180*pi)*sin(c20/180*pi)*sin(c30/180*pi)
       sin(c10/180*pi)*sin(c30/180*pi)-cos(c10/180*pi)*sin(c20/180*pi)*cos(c30/180*pi)
      -sin(c10/180*pi)*cos(c20/180*pi)
       cos(c10/180*pi)*cos(c30/180*pi)-sin(c10/180*pi)*sin(c20/180*pi)*sin(c30/180*pi)
       cos(c10/180*pi)*sin(c30/180*pi)+sin(c10/180*pi)*sin(c20/180*pi)*cos(c30/180*pi)
       sin(c20/180*pi)
      -cos(c20/180*pi)*sin(c30/180*pi)
       cos(c20/180*pi)*cos(c30/180*pi)
       1
```

上の例題は、z 軸周りに c10 度、次に y 軸周りに c20 度、最後に x 軸周りに c30 度回転させる回転行列を書き下したものです。c10, c20, c30 の定数に値を入れることにより、各軸周りの回転を定義できます。

例 49: [transform] セクションの例題 (2)

```
[ Transform ]
*tr1   0 0 0  90   30   0  0 0 0 0 0 0  2
tr2    0 0 0  pi/2 pi/6 0  0 0 0 0 0 0  2
```

これは $M = 2$ の例題です。例題 (48) と同じ回転の座標変換を簡易に入力できます。∗tr1 は degree 単位、∗ の付かない tr2 は radian 単位で回転角度を指定します。

5.8 [Temperature] セクション

このセクションでは、GG の cell の Free-Gas Thermal Temperature を定義します。単位は MeV です。時間の定義はできません。この量は、GG の場合、`[cell]` セクションのセルパラメーターでも定義できますが、領域が重なった場合は、この `[temperature]` で定義した値が優先します。両者で定義されない領域は、2.53×10^{-8}MeV にセットされます。なお、一般的に温度は熱中性子の放射線挙動にのみ影響を与え、それ以外の放射線挙動には物理的に影響を与えません。

書式は、

```
    [ T e m p e r a t u r e ]
        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
        ....           ........
        ....           ........
```

同じ値の領域をまとめて書く、`( { 2 - 5 } 8 9 )` という書式も使えます。ただし、単一の数字で無い場合は必ず `(    )` で括ってください。しかし、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造は、使えません。

領域番号 (`reg`) と温度 (`tmp`) の順番を変えたいときは、`tmp reg` とします。読み飛ばしコラム用の non も使えます。GG の場合も、`cell` でなく `reg` を使ってください。

5.9 [Mat Time Change] セクション

このセクションでは、ある特定の物質をある時間で異なる物質に変化させる機能を定義します。時間の単位は nsec です。書式は、

```
[ Mat Time Change ]
            mat       time      change
              1       50.0        11
              2      100.0        12
              3     1000.0         0
            ....     ........
            ....     ........
```

物質番号 (mat) と (time change) の順番を変えたいときは、mat change time の様に定義行で変えます。読み飛ばしコラム用の non も使えます。この 3 つのコラムは全て必須です。mat で定義された物質が、time で定義された時間で change で定義された物質に変化します。時間によってシャッターの開閉がある場合などを想定しています。

5.10 [Magnetic Field] セクション

5.10.1 荷電粒子

PHITS では、指定した領域において磁場を設定することができます。電子・陽電子についてこの機能を使用する場合は、EGS5 を利用してください。このセクションでは、磁場を及ぼす region もしくは cell の番号 (reg) と、磁場のタイプ (typ)、磁石の間隔の半分 (gap)、磁場の強さ (mgf)、座標変換 (trcl)、時間変化 (time) を定義します。このセクションで定義された磁場を有効にするには、[parameters] において imagnf=1 を指定してください。

書式は、

```
[ 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
   ...              ...   ........       ........      ...      ...
   ...              ...   ........       ........      ...      ...
```

ここで、time は、ユーザー定義の磁場の時間変化のパラメータです。non が、時間変化無しです。ユーザー定義サブルーティンは、usrmgt1.f と usrmgt2.f がサンプルルーティンとして含まれています。前者が Wobbler 磁石、後者が中性子用パルスマグネットです。これらを切り替えるには、パラメータセクションで、usrmgt=1,2 で指定します。Wobbler の時の time パラメータは、位相を指定し、パルスマグネット時は、立ち上がり時間を指定します。pi は、円周率の定数として使用できます。

reg は、region、typ は磁場のタイプ、二重極電磁石が 2、四重極電磁石が 4、gap は、磁石の間隔の半分 (cm)、mgf は磁場の強さ (kG)、trcl は、座標変換の定義番号です。座標変換は、[transform] セクションで定義します。

同じ値の領域をまとめて書く、({ 2 - 5 } 8 9) という書式も使えます。また、(6 < 10[1 0 0] < u=3) などの lattice, universe 構造も指定できます。 ただし、単一の数字で無い場合は必ず (　　) で括ってください。

この書式を用いれば lattice のひとつひとつに 異なる magnet field を定義することも可能です。上の定義で、同じセルが二重に定義された時は、初めに定義された値が採用されます。

これらの順番を変えたいときは、定義の文字列の順番を変えます。読み飛ばしコラム用の non も使えます。GG の場合も、cell でなく reg を使ってください。

二重極電磁石の場合、磁石の間隔 gap は意味がありませんが、何らかの数値を入力して下さい。磁場の働く領域は、真空の領域、即ち、mat が 0 のときに 限りません。磁場中でも減速、散乱が起こります。

trcl は、省略できます。trcl コラムがある場合で、座標変換のないときはゼロを書いてください。磁場中心の軸は z 軸上、磁場の方向は、二重極電磁石では、y 軸正方向、即ち正荷電粒子が z 軸正方向に進むと

きに、x 軸負方向に偏極します。四重極電磁石では、正荷電粒子が z 軸正方向に進むときに、x 軸方向に収束、y 軸方向に分散の配置です。これらの位置関係を幾何形状に合わせるには、trcl による座標変換が必要です。座標変換で、z 軸を磁石の幾何形状の中心軸に変換します。なお、電子・陽電子の場合は、座標変換が有効になりませんのでご注意ください。

[source] セクションのパラメータ izst=を用いて入射粒子の電荷価数を指定した場合、磁場中の運動はその価数に応じたものになります。通常の *PHITS* では、荷電粒子、特に原子核は、磁場の中で電荷価数 Z の粒子として振る舞います。しかしながら、加速原子核では、電荷価数が Z、即ち全ての電子が剥ぎ取られた状態以外に、幾つかの電荷状態が存在するときがあります。izst を用いることで、その状態にある粒子の運動を模擬できるようになります。ここで指定された価数は、輸送の途中では変化しません。ただし、核反応を起こし、生成される原子核には影響を及ぼしません。即ち、原子核反応で生成された原子核の価数は、Z に戻ります。

磁場中の軌跡を [t-track] で表示すると、通常、タリーは境界面横断時及び核反応時にしか呼ばれないので、境界間は直線で表現されてしまいます。磁場中の粒子の軌跡を滑らかに表示したい場合は、[parameters] において deltm で軌跡を表示したいステップ長を定義する必要があります。

5.10.2 中性子

中性子光学のための磁場の書式は、ほぼ荷電粒子の場合と同じですが、細かい点が異なりますので、ここにまとめて説明します。まず書式の例題を下に示します。

```
[ 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
   ...   ...    ........       ........      ...       ...      ...
   ...   ...    ........       ........      ...       ...      ...
```

タイプは上の 7 種類あります。60、61 は、最もシンプルな 6 極磁場で、重力効果や追加 2 極磁場は入れられません。60 は、スピンとの相互作用なし、即ちスピンは磁場に平行か反平行かで記述されます。61 は、スピンとの相互作用が入ります。磁場の弱いところでのスピンの反転などが起こります。磁場の強さは、mgf のコラムに [T/m^2] の単位で指定します。

次に 100 番台は、全てスピンとの相互作用を考慮しています。また、重力効果、追加の 2 極磁場が入れられます。106 が 6 極、104 が 4 極、102 が 2 極磁場です。z 方向の追加 2 極磁場の強さは、gap のコラムに記述します。単位は T(テスラ) です。

101 は、磁場の値をユーザー定義ファイル usrmgf1.f で定義します。このユーザープログラムでは、原研の中性子光学グループで測定された 6 極磁場の値が格納された 4 つのファイルからデータを読み込み、それらを内挿して計算に用います。磁場の強さは、mgf で定義した値に規格化します。gap のコラムで与えた 2 極磁場は、定義した磁場領域全域に作用します。

103 も、ユーザー定義ファイルを参照するオプションです。103 では、usrmgf3.f を参照します。usrmgf3.f には、6 極磁場の式が書いてありますので、このまま動かせば、106 を指定した時と同じになります。ユーザー定義磁場を書く場合は、添付の usrmgf1.f と usrmgf3.f を参考にして下さい。

中性子のスピンは、ソースセクションで sx，sy，sz で、定義すれば、その方向で磁場に入射します。ソースセクションで定義しないか、sx，sy，sz 全てがゼロの場合は、最初に磁場に入った時の磁場の方向で初期化されます。その時、このセクションの polar のコラムで定義される偏極率によりスピンの向きが磁場に平行か反平行かの比率が決定されます。non は、偏極率ゼロと同義です。偏極率 P の定義は、

$$P = \frac{\phi_+ - \phi_-}{\phi_+ + \phi_-}$$

ここで、ϕ_+ と ϕ_- は、スピンが磁場に平行な粒子数と反平行な粒子数です。

5.10.3 磁場マップの読込方法

Version 3.10 より，xyz グリッドもしくは r-z グリッドで表現された任意の磁場構造を読み込むことが可能となりました。その書式の例題を以下に示します。

```
[ 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
```

磁場マップは 4 つの形式で定義でき，それぞれ typ=−1, · · · , −4 で指定します。中性子に対して磁場を考慮したい場合は typ=−101, · · · , −104 となります。1 つのインプットファイルで同タイプの磁場マップを 2 つ以上指定することができません。各形式は表 88 のようになります（詳細は 5.10.4 節参照)。

表 88: 利用できる磁場マップの形式

typ	説明
−1 or −101	xyz グリッド，データリスト型
−2 or −102	r-z グリッド，データリスト型
−3 or −103	xyz グリッド，データマップ型
−4 or −104	r-z グリッド，データマップ型

磁場マップの場合，gap と mgf が通常の磁場の場合と比べて意味が異なり，gap は飛程の分割数（磁場がない場合の飛程を gap×100 等分し，ルンゲ・クッタ法を用いて飛跡を解析します)，mgf は磁場マップで指定した磁場強度の倍率となります。例えば，磁場マップで指定した磁場の単位が T（テスラ）の場合，mgf=10 とすることにより kG 単位に変換して PHITS で読み込むことができます。file は磁場マップファイル名です。なお，電子・陽電子は EGS5 モードを用いた場合のみ磁場が有効となり，その飛程の分割数は gap とは無関係となります。また、電子・陽電子の場合は、通常の磁場と同じく座標変換 (trcl) が有効になりませんのでご注意ください。

5.10.4 磁場マップの形式

全ての磁場マップファイルは，ヘッダー部分とデータ部分に分けられます。ヘッダー部分は#や$を使ってコメント文を挿入することができますが，データ部分にはコメントを書くことができません。各磁場マップのサンプルは \phits\utility\magmap フォルダに格納されていますので，そちらもご参照ください。

5.10.4.1 ヘッダーの指定方法

ヘッダー部分では，PHITS インプットファイルと同じく，スペースは無視され，大文字・小文字の区別はありません。ヘッダーパラメータは，1 行に 1 つずつ指定します。指定可能なヘッダーパラメータを表 89 に示します。

表 89: 磁場マップで指定可能なヘッダーパラメータ

パラメータ	説明
nx	x 方向のグリッド数（typ=-1 & -3 で有効，省略不可）
ny	y 方向のグリッド数（typ=-1 & -3 で有効，省略不可）
nz	z 方向のグリッド数（全ての typ で有効，省略不可）
nr	r 方向のグリッド数（typ=-2 & -4 で有効，省略不可）
xmin	x グリッドの最小値（typ=-3 で有効，D=0）
xmax	x グリッドの最大値（typ=-3 で有効，省略不可）
ymin	y グリッドの最小値（typ=-3 で有効，D=0）
ymax	y グリッドの最大値（typ=-3 で有効，省略不可）
zmin	z グリッドの最小値（typ=-3 & -4 で有効，D=0）
zmax	z グリッドの最大値（typ=-3 & -4 で有効，省略不可）
rmin	r グリッドの最小値（typ=-4 で有効，D=0）
rmax	r グリッドの最大値（typ=-4 で有効，省略不可）
ibxmap	x 方向磁場マップデータの有 (=1) 無 (=0)。（typ=-3 で有効，D=1）
ibymap	y 方向磁場マップデータの有 (=1) 無 (=0)。（typ=-3 で有効，D=1）
ibzmap	z 方向磁場マップデータの有 (=1) 無 (=0)。（typ=-3 & -4 で有効，D=1）
ibrmap	r 方向磁場マップデータの有 (=1) 無 (=0)。（typ=-4 で有効，D=1）
extendx	x 負方向への磁場の拡張（typ=-1 & -3 で有効，省略した場合は拡張しない）
extendy	y 負方向への磁場の拡張（typ=-1 & -3 で有効，省略した場合は拡張しない）
extendz	z 負方向への磁場の拡張（全ての typ で有効，省略した場合は拡張しない）
data	この行以降がデータ部分となります（全ての typ で有効，省略不可）

extend を指定し，磁場の向きを反転させて拡張したい場合は，同じ行に flip=br，bz のように反転したい磁場を書きます。例えば

```
extendx flip = bz
extendy flip = bz
```

とした場合は，図 54 のように z 方向の磁場の向きが反転します（x,y 方向に関しては，同じ向きの磁場が拡張されます)。

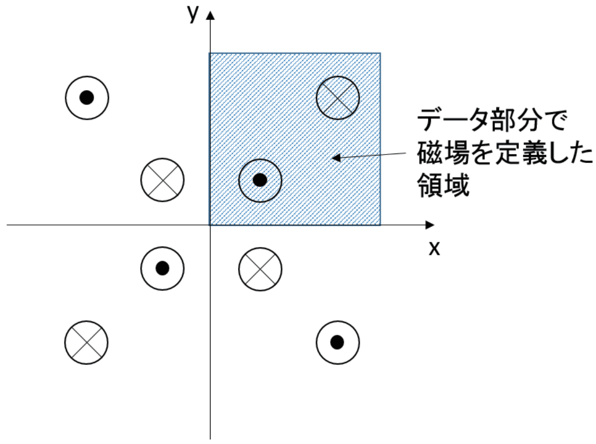


図 54: `flip` を使用し、磁場の向きを反転させて拡張した例。

5.10.4.2 磁場強度の定義方法

データ部分では，スペース，タブ，もしくはコンマ区切りで数値データを指定します。文字やコメントは入れられませんのでご注意ください。ただし，数値データの後ろのカラムであれば自由にコメントを入れることができます。距離の単位は cm，磁場の単位は kG です。粒子位置における磁場強度はグリッド上の値を線形内挿して導出しますので，入力する磁場強度は，グリッドの中点ではなく各グリッド上の値を入力してください。以下，`typ` 別の磁場指定方法を示します。

- `typ=-1`: xyz グリッド，データリスト型

$$
\begin{array}{cccccc}
x_1 & y_1 & z_1 & B_{x_{1,1,1}} & B_{y_{1,1,1}} & B_{z_{1,1,1}} \\
x_1 & y_1 & z_2 & B_{x_{1,1,2}} & B_{y_{1,1,2}} & B_{z_{1,1,2}} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
x_1 & y_1 & z_{nz} & B_{x_{1,1,nz}} & B_{y_{1,1,nz}} & B_{z_{1,1,nz}} \\
x_1 & y_2 & z_1 & B_{x_{1,2,1}} & B_{y_{1,2,1}} & B_{z_{1,2,1}} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
x_1 & y_{ny} & z_{ny} & B_{x_{1,ny,nz}} & B_{y_{1,ny,nz}} & B_{z_{1,ny,nz}} \\
x_2 & y_1 & z_1 & B_{x_{2,1,1}} & B_{y_{2,1,1}} & B_{z_{2,1,1}} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
x_{nx} & y_{ny} & z_{ny} & B_{x_{nx,ny,nz}} & B_{y_{nx,ny,nz}} & B_{z_{nx,ny,nz}}
\end{array}
$$

- `typ=-2`: rz グリッド，データリスト型

$$
\begin{array}{cccc}
r_1 & z_1 & B_{r_{1,1}} & B_{z_{1,1}} \\
r_1 & z_2 & B_{r_{1,2}} & B_{z_{1,2}} \\
\vdots & \vdots & \vdots & \\
r_1 & z_{nz} & B_{r_{1,nz}} & B_{z_{1,nz}} \\
r_2 & z_1 & B_{r_{2,1}} & B_{z_{2,1}} \\
\vdots & \vdots & \vdots & \\
r_{nr} & z_{nz} & B_{r_{nr,nz}} & B_{z_{nr,nz}}
\end{array}
$$

- `typ=-3`: xyz グリッド，データマップ型

$$
\begin{array}{ccccc}
B_{x_{1,1,1}} & B_{x_{1,2,1}} & B_{x_{1,3,1}} & \cdots & B_{x_{1,ny,1}} \\
B_{x_{2,1,1}} & B_{x_{2,2,1}} & B_{x_{2,3,1}} & \cdots & B_{x_{2,ny,1}} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
B_{x_{nx,1,1}} & B_{x_{nx,2,1}} & B_{x_{nx,3,1}} & \cdots & B_{x_{nx,ny,1}} \\
B_{x_{1,1,2}} & B_{x_{1,2,2}} & B_{x_{1,3,2}} & \cdots & B_{x_{1,ny,2}} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
B_{x_{nx,1,nz}} & B_{x_{nx,2,nz}} & B_{x_{nx,3,nz}} & \cdots & B_{x_{nx,ny,nz}}
\end{array}
$$

（同様のマップを B_y, B_z に対しても繰り返す）

- `typ=-4`: rz グリッド，データマップ型

$$
\begin{array}{ccccc}
B_{r_{1,1}} & B_{r_{1,2}} & B_{r_{1,3}} & \cdots & B_{r_{1,nz}} \\
B_{r_{2,1}} & B_{r_{2,2}} & B_{r_{2,3}} & \cdots & B_{r_{2,nz}} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
B_{r_{nr,1}} & B_{r_{nr,2}} & B_{r_{nr,3}} & \cdots & B_{r_{nr,nz}}
\end{array}
$$

（同様のマップを B_z に対しても繰り返す）

データの順番は変更することはできません。ただし，`typ=-3` & -4 で `ibxmap` などのパラメータを 0 にした場合は，その方向に対する磁場データを省略することができます（その場合，その方向に対する磁場強度は全て 0 となります）。また，`typ=-1` & -2 で指定する座標は，必ず昇順となるようにしてください。計算時間は，同じ磁場であればデータマップ型の方が短くなりますが，データリスト型でも各グリッド間隔が全て等しければ，データマップ型とほぼ同等になります。

5.11 [Electro Magnetic Field] セクション

このセクションでは、指定した空間に電場と磁場を設定するためのパラメータを定義します。[parameters] セクションで ielctf=1 とすることでこのセクションで定義したパラメータが有効となり、荷電粒子の電磁場中の運動を記述できるようになります。電子・陽電子についてこの機能を使用する場合は、EGS5 を利用してください。電場と磁場は任意に併存させることができます。

5.11.1 一様な電場と磁場の利用

次の書式により、一様な電場や磁場を設定できます。ただし、[Magnetic Field] セクションで定義できる 4 極磁場は設定できません。

```
[ Electro Magnetic Field ]
  reg    elf    mgf    trcle    trclm
    1    100      1        1        2
    2    100      1        1        2
```

パラメータは、電磁場を指定する region もしくは cell の番号 (reg)、電場の強さ (elf)、磁場の強さ (mgf)、電場の方向 (trcle)、磁場の方向 (trclm) です。場の強さ elf と mgf の単位は、それぞれ kV/cm と kGauss です。電場と磁場の方向 trcle と trclm は、座標変換番号 trcl を用いて定義します。これらを 0 とした場合は、電場の向きが x 軸の +、磁場の向きが y 軸の+の方向となります。trcle と trclm は省略できません。場の強さを 0 とした場合や trcl が必要ない場合も 0 と入力してください。なお、電子・陽電子の場合は、座標変換が有効になりませんのでご注意ください。

[source] セクションのパラメータ izst を用いて入射粒子の電荷価数を指定した場合、電磁場中の運動はその価数に応じたものになります。通常の PHITS では、荷電粒子、特に原子核は、電場や磁場の中で原子番号 Z の電荷をもつ粒子として振る舞います。しかしながら、加速原子核では、電荷価数が Z、即ち全ての電子が剥ぎ取られた状態以外に、幾つかの電荷状態が存在するときがあります。izst を用いることで、その状態にある粒子の運動を模擬できるようになります。ここで指定された価数は、輸送の途中では変化しません。ただし、核反応を起こして生成される原子核には影響を及ぼしません。即ち、原子核反応で生成された原子核の価数は、その原子番号 Z に戻ります。

磁場中の軌跡を [t-track] で表示すると、通常、タリーは境界面横断時及び核反応時にしか呼ばれないので、境界間は直線で表現されてしまいます。磁場中の粒子の軌跡を滑らかに表示したい場合は、[parameters] において deltm で軌跡を表示したいステップ長を定義する必要があります。

5.11.2 電場マップの利用方法

Version 3.20 より、xyz グリッドもしくは r-z グリッドで表現された任意の電場構造を読み込むことが可能となりました。基本的には書式は 5.10.3 節「磁場マップの読込方法」と同じですが、ファイル名は filee で指定します。また、磁場マップと共存させる場合は、そのファイル名を filem で指定します。その書式の例題を以下に示します。

```
[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 及び typm は、電場及び磁場マップの形式を表し、表 88 で示す typ と同じ値が指定できます。ただし、中性子に対しては動作しませんので、−101 から −104 を指定することはできません。1 つのインプットファイルで同タイプの電場マップもしくは磁場マップを 2 つ以上指定することができません。elf 及び mgf は、電場及び磁場マップの強度倍率で、それぞれ 1 とした場合の単位は kV/cm と kG となります。gap はルンゲ・クッタ法を用いて飛跡を解析する際に飛程の分割数を決定するパラメータであり、そのデフォルト値は 10 で、大きくするほど刻み幅が小さくなります。なお、電子・陽電子は EGS5 モードを用いた場合のみ電磁場が有効となり、その飛程は gap とは無関係になります。trcle と trclm は、電場・磁場それぞれの座標変換番号を示しており、通常の電磁場と同じく、電子・陽電子の場合は有効になりませんのでご注意ください。

電場マップの定義方法は、単位が kG ではなく kV/cm である点を除いて磁場マップの定義方法と同じですので、5.10.4 節「磁場マップの形式」をご参照ください。

5.12 [Delta Ray] セクション

このセクションでは、物質中を荷電粒子が通過した場合にその飛跡周辺に発生するノックアウト電子 (δ 線) を 2 次粒子として発生させる機能をコントロールします。荷電粒子が通過する物質に与えるエネルギーは、通常 LET(dE/dx) として評価され、その軌道上にのみ付与されます。しかし、高エネルギーの δ 線が発生した場合などは、その輸送によって 1 次粒子の軌道から離れた位置にエネルギーが付与されることが知られており、本セクションを利用することでその影響を調べることが可能となります。δ 線生成断面積は、Butts と Katz の式[71] より計算し、そのエネルギーや角度を決定する際は相対論を考慮しています

本機能と `[t-sed]` タリーは併用できませんのでご注意ください。

真空 (void) 以外の領域毎に、δ 線を発生させるしきい値エネルギー $E_{\rm th}$（単位は MeV）を決めることができ、この値より高いエネルギーの δ 線を実際に 2 次粒子として発生させます。$E_{\rm th}$ 以下の δ 線の寄与は通常の LET で評価しており、設定できる $E_{\rm th}$ の最小値は 0.001 MeV(=1 keV) です。ただし、$E_{\rm th}$ を 10 keV 以下に設定したり、δ 線を発生させる領域の厚さをおおよそ 10 μg/cm^2 以下に設定すると、荷電粒子の挙動が多少変化しますのでご注意下さい。具体的には、荷電粒子の実効的な阻止能が、δ 線を多く放出しすぎることにより高くなってしまいます。また、1 MeV 以下の荷電粒子による δ 線生成を計算したい場合は `emin(1)`, `emin(i=15-19)` を 1 keV に設定する必要があります。`[delta ray]` セクションを利用しない場合は、$E_{\rm th}$ にデフォルトの値として $1.e+10$ が入っており、事実上 δ 線は発生しません。領域番号と $E_{\rm th}$ はそれぞれ `reg`, `del` で指定します。その書式は以下の例の通りです。

```
[ delta ray ]
    reg     del
      1     0.1
     11     1.0
    ....   ....
    ....   ....
```

同じ値の領域をまとめて書く、`( { 2 - 5 } 8 9 )` という書式も使えます。ただし、単一の数字で無い場合は必ず `(      )` で括ってください。しかし、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造は使えません。領域番号 (`reg`) としきい値エネルギー (`del`) の順番を変えたいときは、`del reg` とします。読み飛ばしコラム用の non も使えます。GG の場合も、`cell` でなく `reg` を使ってください。

[71] 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] セクション

1 keV 以下のような低エネルギー電子・陽電子について、飛跡構造計算[72] を実施すると、電子・陽電子が電離・励起もしくは分子振動・回転のような衝突イベントによる減速過程を模擬することができます。現状では、衝突断面積は液体水に対してのみ整備されており、それ以外の物質に対しては、液体水の衝突断面積を簡易的に電子密度スケーリングすることで対処しています。この計算機能を利用すると、10 eV 程度の低エネルギー電子数が多数生成されるため、従来アルゴリズムと比較して計算時間が膨大に掛かります。そのため、通常の大きさの体系（cm オーダー）における放射線挙動解析には適しません。また、飛跡構造解析モードは [t-track] と [t-deposit] タリーのみにしか対応しておりませんのでご注意ください。

このセクションでは、飛跡構造解析を行うセルを指定します。mID は利用する断面積データの ID 番号で、現状では 0（= 飛跡構造解析を行わない）か 1（= 液体水）しか指定できません。水以外の標的物質に対し mID =1 を指定すると、液体水の断面積データから電子密度スケーリングにより、標的物質の平均自由行程が算出されます。今後、様々な物質に対する断面積データを順次整備していく予定です。

```
[track structure]
reg  mID
  1    1
  2    0
```

同じ値の領域をまとめて書く、(2 - 5 8 9) という書式も使えます。ただし、単一の数字で無い場合は必ず ( ) で括ってください。しかし、(6 < 10[100] < u=3) などの lattice, universe 構造は使えません。領域番号 (reg) と利用する断面積データの ID 番号 (mID) の順番を変えたいときは、mID reg とします。読み飛ばしコラム用の non も使えます。

以下は重要なパラメータの設定です。[parameters] セクションに etsmax, etsmin を設定します。これらは飛跡構造解析を実施する電子・陽電子エネルギーの上限・下限値で単位は (MeV) です。上限値を上げる、もしくは下限値を下げると計算時間がより長くなりますので、ご注意ください。飛跡構造解析モードを利用する場合は、必ず EGS5 モード (negs=1) を用いて emin(12-13) を 1keV に設定する必要があります。

```
[ P a r a m e t e r s ]
emin(12) = 1.E-03
emin(13) = 1.E-03
negs     = 1
etsmax   = 1.E-2
etsmin   = 1.E-6
```

[72] 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).

バージョン 3.20 より、陽子・炭素イオンの飛跡構造解析モードも、KURBUC[73]の実装により、利用可能になりました。利用方法は電子の場合と同様です。ただしその他に `tsmax` (D=1e-3 MeV/n) を `[parameters]` セクションに設定してください。`tsmax` は本機能が有効となるエネルギー上限値です。ただし、利用できる飛跡構造用断面積ライブラリの上限エネルギーは、現在のところ、陽子・炭素イオンのそれぞれの場合について 300 MeV と 10 MeV/n ですので、それ以上大きい `tsmax` が設定された場合は、自動的にライブラリのある範囲内で飛跡構造解析を実施します。エネルギー下限値は、陽子・炭素イオンのそれぞれの場合について `emin(1)` と `emin(19)` (D=1e-3 MeV/n) で調整可能です。

[73] 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

5.14 [Super Mirror] セクション

このセクションでは、低エネルギーの中性子のスーパーミラーによる反射の機能を定義します。ここでは、以下のような半経験的なスーパーミラーの反射率を仮定します。

$$R = \begin{cases} R_0 & \text{if } Q \leq Q_c \\ \frac{1}{2} R_0 \left(1 - \tanh\left[(Q - mQ_c)/W\right]\right)\left(1 - \alpha(Q - Q_c)\right) & \text{if } Q > Q_c \end{cases}$$

ここで Q は、散乱ベクター (Å^{-1}) で、次のように定義されます。

$$Q = |\mathbf{k}_i - \mathbf{k}_f| = \frac{4\pi \sin\theta}{\lambda}$$

m は、ミラーの物質とレイヤーの数などに依存するパラメーターです。Q_c は、一層のレイヤーによる臨界散乱ベクター、これ以上の Q で、反射率は α の傾斜で直線的にカットオフ値 $Q = mQ_c$ まで減少する。そのときのカットオフの幅は W で表される。

これらのパラメーターは、このセクションで次のように定義される。

```
[ 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
   ....           ....     ..   ...   ....     ...   ...
   ....           ....     ..   ...   ....     ...   ...
   ....           ....     ..   ...   ....     ...   ...
```

反射の面は、`r-in` が入射領域、`r-out` が反射体領域の間の面で定義されます。

同じ値の領域をまとめて書く、`( { 2 - 5 } 8 9 )` という書式も使えます。また、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造も 指定できます。

上の標識のその他のパラメーターは、m を `mm` で、R_0 を `r0`、Q_c を `qc` (Å^{-1})、α を `am` (Å)、W を `wm` (Å^{-1}) で定義します。

このスーパーミラーの反射は、10eV 以下の中性子、また、sin$\theta > 0.001$ の時に制限されます。後者は、表面の粗さに因るものです。

5.15 [Elastic Option] セクション

このセクションでは、核データを用いる低エネルギー中性子の弾性散乱について、ユーザー定義の角分布を与えるパラメータを指定します。ユーザー定義のサブルーティンのサンプルとして、usrelst1.f と usrelst2.f が用意してあります。これらは、パラメータセクションで、usrelst=1, 2 で切り替えることができます。

書式は、以下のように、このオプションを適用する領域番号と、ユーザー定義サブルーティンで用いる 4 つの定数を定義することができます。

```
[ 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
           ....      ...      ...       ...       ...
           ....      ...      ...       ...       ...
```

物質番号 (reg) と (c1 c2 c3 c4) の順番を変えたいときは、reg c3 c2 c1 c4 の様に定義行で変えます。読み飛ばしコラム用の non も使えます。同じ名前をまとめて書く、{ 4 - 7 } という書式も使えます。ただし、({ 4 - 7 } 9 10) という表式は使えません。

現在、サンプルサブルーティンとして入っている、usrelst1.f と usrelst2.f は、それぞれ、前者がデータベースを用いた Bragg 散乱の角分布、後者が、任意の関数形を用いた角分布が記述できるようになっています。

5.16 [Data Max] セクション

このセクションでは、マテリアルの核種ごとにライブラリー利用の上限エネルギー dmax を設定することができます。このセクションは、6 つまで定義することができます。

```
[Data Max]
  part = neutron proton
  mat   nucleus  dmax
   all   Fe      20
    5    all     50
    3    56Fe    150
```

最初の行に part=の書式で輸送粒子を指定します。省略はできません。また、ここで定義できる粒子は、version 2.86 において、neutron と proton だけです。

マテリアルごとに指定するときは mat コラムを用います。ここにはマテリアル番号、もしくは、all を使うことができます。nucleus コラムで核種を指定します。56Fe、もしくは、26056 の書式で指定します。Fe、26000 などの質量を指定しない場合は、その核種の全ての同位体を指定することになります。また、all も使うことができます。dmax のコラムでその核種のライブラリー利用の上限エネルギー (MeV) を指定します。(mat) (nucleus) (dmax) の順番は可変です。また、読み飛ばしのコラム non も使えます。

同じ核種が重ねて定義されている場合は、後出が優先されます。

[parameters] セクションで定義される、dmax(1) と dmax(2) は、このセクションで定義される dmax の最大値とすることが望まれます。

dmax の値がライブラリの上限エネルギーよりも高い値に設定された場合は、上限エネルギーの断面積の値を dmax まで使います。例えば、通常の JENDL-4.0 を使う場合、dmax=200 と設定すると、20 MeV から 200 MeV までは 20 MeV の断面積が使用されます。また、ライブラリによって上限が違う場合は本セクションで個別に定義してください。

[parameters] セクションで、kmout=1 を指定すると、file(6) (D=phits.out) にマテリアル毎に各核種の dmax が出力されます。

5.17 [Frag Data] セクション

このセクションでは、ユーザー指定断面積読み込み機能について定義できます。この機能を使用することにより、特定の粒子間の核反応イベントをユーザーが指定した断面積データを用いて模擬できます。ただし、本機能で指定した粒子間、エネルギー領域にある反応に対しても、核データライブラリを使用する設定になっている場合は、データライブラリにある情報が優先的に使用されます。また、指定できる入射粒子は、陽子や中性子、原子核のみです。光子や電子は指定できないのでご注意ください。他に、核子の弾性散乱の寄与は PHITS のモデルにより別途考慮されます。

本機能を用いて生成される粒子は、その断面積に応じてウエイト値が変わります。場合によっては非常に小さい値となりますので、ウエイトカットオフを避けるために、`[parameters]` セクションにおいて `wc2(i)` の値を粒子 `i` 毎に変えてください。粒子の反応計算打切エネルギー `cmin(i)` は、考慮したい最小の入射エネルギーに応じて設定してください。

書式は、以下のようになります。

```
[ Frag Data ]
  opt     proj     targ     file
    0     12C      16O      DDX_12C-16O.dat
    1     proton   63Cu     DDX_p-63Cu.dat
```

opt が 0 の場合は指定した断面積を使用しません。これが 1 の場合は、proj と targ で指定した粒子間の核反応を file で指定したファイルにある断面積データを基に模擬します。opt が 2,3 の場合は整備中です。opt が 4 の場合は、入射エネルギーや放出角度、放出エネルギーについて、与えられたデータを基に単純な外挿を行います。ただし、この外挿オプションは、下記にある neo が正で、nag が 0 でない場合しか使用できません。また、放出エネルギーや角度の微分断面積データが群形式ではなく点形式で与えられた時は、opt を 5 としてください。この場合、線形内挿を行って放出粒子のエネルギーと角度の依存性を再現します。

file で指定する断面積データファイルの書式は、以下のようになります。

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)

（続き）

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)*

. . . .
. . . .

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)*

最初に、projectile と target において入射粒子と標的粒子のそれぞれを指定します。nei は入射エネルギーの分点数で、その次の行に nei+1 点の入射エネルギー（ein: 単位は MeV/n）を指定し、更にその次の行に同数の全反応断面積（totxs: 単位は mb）を指定します。ここで totxs が 0 の場合は、`icxsni` や `icrhi` で指定する全反応断面積モデルにより得られた値が使われます。neo は放出粒子のエネルギー分点数で、その次の行に neo+1 点の放出エネルギー（eout: 単位は MeV/n）を指定します。ただし、opt が 5 の場合は neo 点

の eout を指定してください。nag は放出角度の分点数で、その次の行に nag+1 点の角度（angle）を指定します。ただし、opt が 5 の場合は nag 点の angle を指定してください。nag>0 の場合はラジアン単位 (rad)、nag<0 の場合は度数単位 (degree) で angle を与えます。nag=0 の場合は等方分布を仮定します。nfrg は放出粒子の数で、その次の行に nfrg 個の放出粒子を指定します。proxs は各放出粒子のある入射エネルギーに対する生成断面積 (mb) で、その次の行から 2 重微分断面積（ddx: 単位は mb/MeV/sr）を並べます。1 つの proxs に対して neo×nag 点の 2 重微分断面積のデータが必要で、それを 1 単位として nei+1 個のデータを並べます。放出粒子を複数指定する場合 (nfrg>1) は、proxs から始まるデータ群が nfrg で指定した数だけ必要となります。

また、neo を書くべき行で `model` と書いた場合は、PHITS に組み込まれている核反応モデルが使用されます。その場合は、微分断面積のデータを与える必要はありません。

例 50: 断面積データファイルの例

```
 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
```

例 50 に断面積データファイルの一例を示します。1,2 行目にそれぞれ入射粒子として proton、標的粒子として ^{63}Cu を指定しています。3 行目がエネルギー分点数で、この例では nei=1 としています。4,5 行目において入射エネルギーと全反応断面積を nei+1 点ずつ与えています。6 行目は放出粒子のエネルギー分点数です。この例では neo=3 としており、7 行目で放出エネルギーとして、1.0, 10.0, 50.0, 100.0MeV の値を指定しています。8 行目は角度の分点数です。これをマイナスにし、9 行目で与える角度の情報を度数単位で与えています。10 行目で放出粒子として取り扱いたい数を決めており、11 行目でその粒子 (=neutron) を指定しています。12 行目に入射エネルギーが 10.0MeV の場合の neutron 生成反応断面積を与えており、13,14,15 行目でこの入射エネルギーにおける 2 重微分断面積を与えています。各行は 7 行目で決めたエネルギービンに対応しており、各列は 9 行目で決めた角度ビンに対応しています。基本的に、12 行目の生成反応断面積には、13,14,15 行目の微分断面積を積分した量を与えてください。一致しない場合は、生成反応断面積の値を使って規格化されます。16,17,18,19 行目は、入射エネルギーが 100.0MeV の場合の生成反応断面積と 2 重微分断面積です。16 行目の生成反応断面積は 0 にしていますが、この場合、生成反応断面積として、17,18,19 行目の微分断面積を内挿して積分した値を使用します。

neo をマイナスで指定することで、放出粒子のエネルギーを離散的に与えることができます。この場合、eout の個数は neo としてください。

neo を 0 にすると、放出粒子のエネルギースペクトルをガウス分布で与えることができます。この場合は、ddx の箇所にガウス分布の平均値と分散の値を MeV で与えてください。

ddx の箇所で、0 の値が並ぶ場合は次のような省略形を利用できます。

```
0 -9
```

この例では、0 が合計 10 個並ぶことを意味します。

5.18 [Importance] セクション

このセクションでは、GG の cell の importance を定義します。定義されない領域もしくは粒子は、1.0 にセットされます。

このセクションは、6 つまで定義することができます。書式は、

```
   [ I m p o r t a n c e ]
      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
       ....           ........
       ....           ........
```

最初の行に `part =` の書式で粒子を指定します。省略した時のデフォルトは、`part = all` で全粒子です。`part =` の書式の詳細はタリーの粒子指定と同じ書式です。しかし、指定できるのは `ityp` としての区別だけです。それぞれの原子核などは、個別には指定できません。

領域番号 (`reg`) と (`imp`) の順番を変えたいときは、`imp reg` とします。読み飛ばしコラム用の non も 使えます。GG の場合も、`cell` でなく `reg` を使ってください。

同じ値の領域をまとめて書く、`( { 2 - 5 } 8 9 )` という書式も使えます。また、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造も 指定できます。ただし、単一の数字で無い場合は必ず `(   )` で括ってください。

階層構造のある場合の importance の指定の仕方は、 各階層の importance の積を最下層の importance とします。ただし、最下層の特定のセルの importance の指定が、上の書式で可能です。この書式を用いれば lattice のひとつひとつに 異なる importance を定義することも可能です。上の定義で、同じセルが二重に定義された時は、初めに定義された値が採用されます。

透過性のある粒子に、具体的にはニュートリノですが、importance を強く設定されていると、計算時間が膨大になります。この危険を避けるために、part=all で指定してもニュートリノは省かれていますので、注意して下さい。

階層のあるセルの importance の指定は、幾つかの書き方が とり得ます。例えば、最下層のセル 5 6 7 が共に上位のセル 11 12 13 に含まれる時、

```
[ Importance ]
   reg           imp
 ( 5 6 7 < 11 )  2.0
 ( 5 6 7 < 12 )  4.0
 ( 5 6 7 < 13 )  8.0
 ( 11 12 13 )    1.0
```

で定義される importance と、

```
[ Importance ]
   reg           imp
 ( 5 6 7 )       1.0
   11            2.0
   12            4.0
   13            8.0
```

は、同じ結果を与えます。ただし、後者の場合、importance のサマリーでセル 5 6 7 の importance の値は、1.0 と表示されます。

5.19 [Weight Window] セクション

このセクションでは、ウエイトウインドウを定義します。このセクションは、6 つまで定義することができます。書式は、以下のような形式となります。

```
        [ 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
                           ........   ........
```

最初の行に、mesh = reg もしくは xyz を指定して形状メッシュを指定してください。もし mesh の指定を省略した場合は、デフォルトである reg が形状メッシュとして選択されます。mesh=xyz を指定した場合は、その後に x-type，y-type，z-type でメッシュの大きさや数を定義してください。詳しい定義方法は、“6. タリー共通パラメータの書式” を参照してください。なお、mesh = r-z は定義できませんのでご注意ください。

そして、part =の書式で粒子を指定します。省略した時のデフォルトは、part = all で全粒子です。part =の書式の詳細はタリーの粒子指定と同じ書式です。しかし、指定できるのは ityp としての区別だけです。それぞれの原子核などは、個別には指定できません。

次に、エネルギーもしくは、時間のメッシュを定義します。eng =もしくは tim =でメッシュの個数を、次行に、メッシュの値 $(e_1, e_2, e_3, ...)$ を定義します。このメッシュ毎のウエイトウインドウの下限値を以下に定義します。各下限値は、ww1, ww2, ww3, ... でエネルギーもしくは時間メッシュを指定します。ここで wwi は、$e_{i-1} < E < e_i$ のメッシュのウインドウ下限値です。$e_0 = 0, t_0 = -\infty$ を仮定しています。eng =もしくは tim =の定義がない場合は、エネルギーもしくは、時間のメッシュを仮定せず、ひとつの領域とします。この場合は、ww1 だけを定義します。

領域（reg もしくは xyz）は、最初のコラムに固定です。上の例題の様に、ひとつの表に、ww1, ww2, ... が書き切れない時は、次の表に書き足すことができます。この時、最初の領域（reg や xyz）のコラムは、省略できます。また、領域の並びは、上の表と同じと仮定しています。読み飛ばしコラム用の non は使えます。

`reg` のコラム指定する際は、同じ値の領域をまとめて書く `( { 2 - 5 } 8 9 )` という書式も使えます。そして、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造も指定できます。ただし、単一の数字で無い場合は必ず `(    )` で括ってください。また、`xyz` の場合は `(ix iy iz)` の順番で指定してください。

透過性のある粒子に、具体的にはニュートリノですが、importance を強く設定されていると、計算時間が膨大になります。この危険を避けるために、`part=all` で指定してもニュートリノは省かれていますので、注意して下さい。

5.20 [WW Bias] セクション

このセクションでは、[weight window] で指定したウエイトの下限値に対して、特定の領域にバイアス（傾斜）をかけることができます。特に、[t-wwg] を用いて自動的に [weight window] を作成した際、更に特定の領域にウエイト値のバイアスをかけたい場合に利用できます。図 55 に、[weight window] と [ww bias] を組み合わせた輸送計算を実行する際の流れを示しました。最初に [t-wwg] を用いて [weight window] を作成します。これで十分な場合は不要ですが、注目しているタリー領域が限定的である場合は、[ww bias] を作成して特定の領域にバイアスをかけることにより、効率的に統計量を増やすことが可能となります。[ww bias] を作成する方法は 2 通りあり、Weight Window Bias Generator (WWBG) である [t-wwbg] を利用して自動的に作成する方法と、本節でご紹介する手動で作成する方法です。[t-wwbg] の利用方法については、7.17 節 [T-WWBG] セクションをご覧ください。作成した [ww bias] と [weight window] を組み合わせた輸送計算を実行する場合は、[parameters] において icntl=0 および iwwbias=1 としてください。

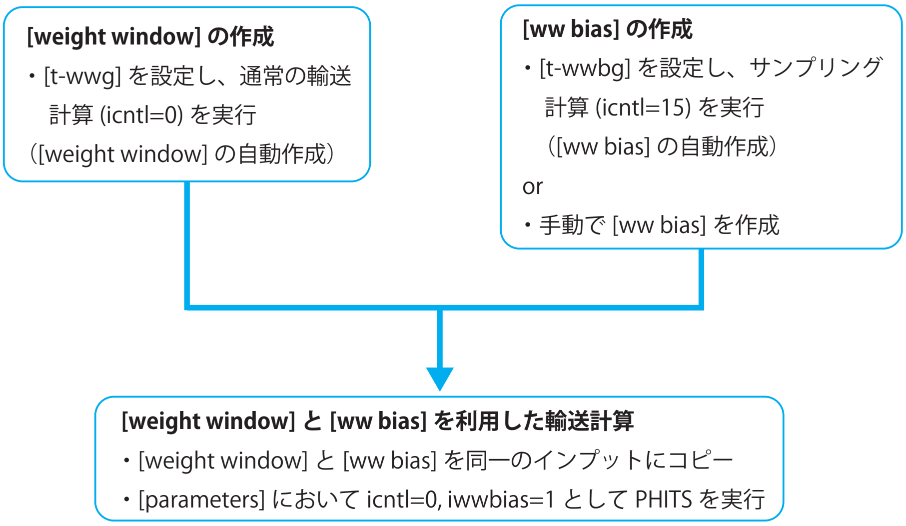


図 55: [weight window] と [ww bias] を組み合わせた輸送計算の概要。

[ww bias] の書式は、以下の様になります。バイアスをかける予定の [weight window] と同じ粒子、エネルギーメッシュ、領域番号を指定してください。

```
[ 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
   ....   ........    ........
```

最初に part=の書式で粒子を指定します。part=all がデフォルトなので、省略した場合は全粒子が対象となります。part=の書式の詳細はタリーの粒子指定と同じです。ただし、指定できるのは ityp としての区別だけです。核種のそれぞれを個別に指定することはできません。次に、エネルギーメッシュを定義します。eng=でメッシュの個数を指定し、その次の行に、メッシュの値 $(e_1, e_2, e_3, ...)$ を定義します。更にその次の行に reg, wwb1, wwb2, ... と記載し、reg の列には領域番号、wwb*i* の列には各領域にかけたいバイアスの値を与えます。その際、読み飛ばしコラム用の non が使えます。wwb*i* は、eng=で指定したエネルギーメッシュの数だけ用意してください。各 wwb*i* の列は、エネルギー E が $e_{i-1} < E < e_i$ となる粒子を対象としています。ここで $e_0 = 0$ です。領域番号の指定では、同じ値の領域をまとめて書く ({ 2 - 5 } 8 9) という書式も使えます。また、(6 < 10[1 0 0] < u=3) などの lattice, universe 構造も指定できます。ただし、単一の数字で無い場合は必ず (   ) で括ってください。

[parameters] において iwwbias=1 とすると、[ww bias] で与えたバイアスの値の逆数が [weight window] で指定したウエイトの下限値にかかります。この場合、input echo の [weight window] には [ww bias] で定義した値を考慮した結果が書き出され、[ww bias] には off が付きます。もし、[ww bias] が定義されていない場合は、input echo において全ての値が 1 となった [ww bias] が書き出されます。

[ww bias] を用いた計算の 1 例を以下に示します。

例 51: [ww bias] の例題

```
[ WW Bias ]
   part = neutron
    eng =  1
         1.00000E+05

   reg       wwb1
   1         1/7
   2         1/6
   3         1/5
   4         1/4
   5         1/3
   6         1/2
   7         1
   8         2
   9         3
   10        4
   11        5
   12        6
```

考慮する粒子は中性子とし、エネルギーメッシュは 100GeV 以下の領域 1 つを指定しています。領域番号 1 から 12 に対してバイアスをかけており、領域 7 を中心にして、少ない方には弱く、大きい方には強くバイアスをかけています。

図 56(a) が、例題 51 を適用した体系の xz 断面図です。体系には z 軸を中心とした半径 100cm のコンクリートの円柱が配置されており、$x = 0, y = 0, z = 90$cm の点から 14MeV 中性子を等方線源として発生させました。最初に `[t-wwg]` により `[weight window]` を作成し、そのウエイト値をそのまま用いて計算して求めた中性子のフルエンス分布が図 56(b)、例題 51 を適用して求めた結果が図 56(c) です。図 (b) を見てわかるように、バイアスをかけずに `[weight window]` を利用して求めた結果は、z 軸のプラス側とマイナス側に同様に粒子数が広がっています。それに対し、図 (c) では、例題 51 に示す値が示すように、z 軸のプラス側では粒子が増加し、マイナス側では減少しています。このように、マイナス側が重要でなく、プラス側の領域について詳細に調べたい場合は、`[ww bias]` によってバイアスをかけることにより、効率的にシミュレーションを実行することができます。

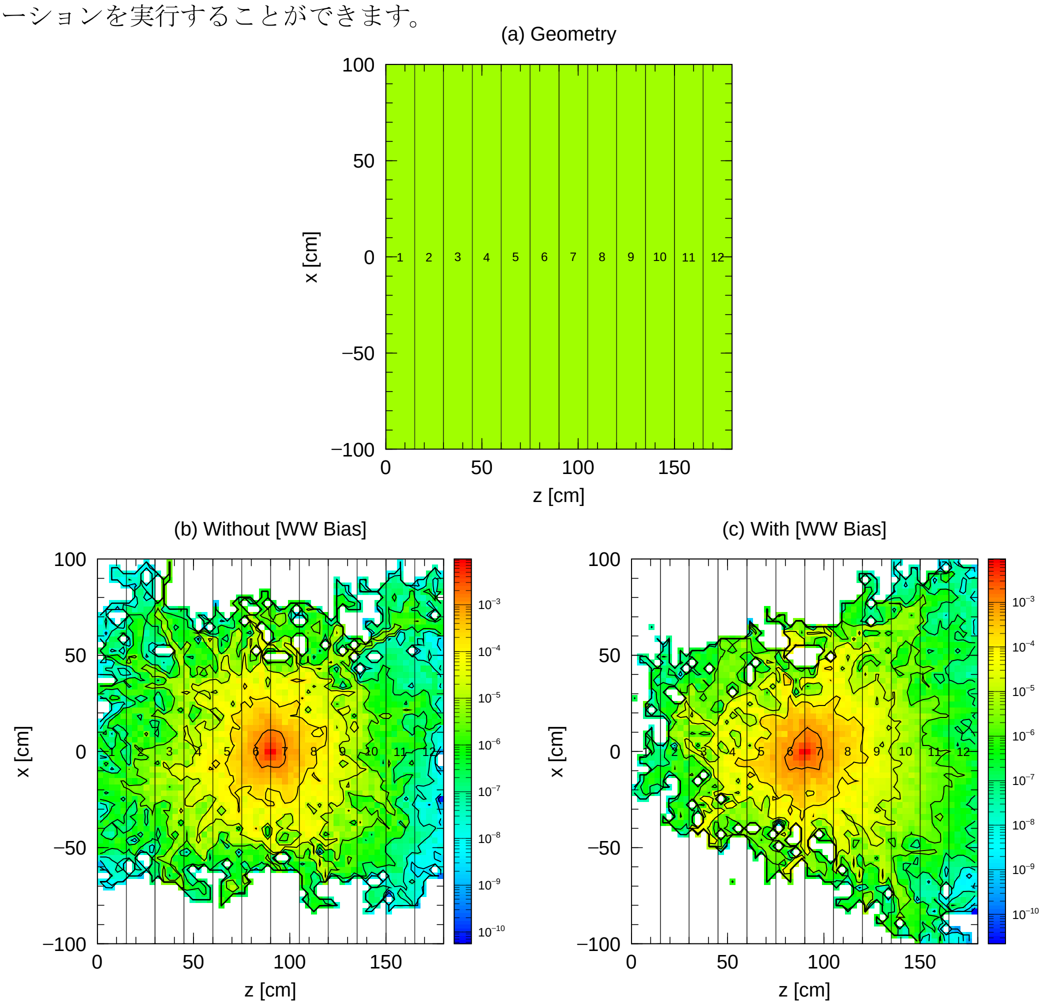


図 56: (a) 体系の xz 断面図。(b)`[ww bias]` を適用せずに計算した結果。(c) 例題 51 に示した `[ww bias]` を適用して計算した結果。

5.21 [Forced Collisions] セクション

強制衝突 (forced collisions) は、衝突の確率を人為的に上げてタリーの効率を上げたり、衝突の確率の低い薄いターゲットなどを解析するときに用います。強制衝突の領域に入った指定された粒子は、ふたつに分離されます。ひとつはその領域の透過確率をその重みにかけた粒子で、次の領域まで透過させます。もうひとつは、(1 - 透過確率) を乗じた重みを持った粒子で、この強制衝突の領域で強制的に衝突を起こさせます。衝突位置は、断面積に従って確率的に決定します。

強制衝突の領域と強制衝突係数をこのセクションで指定します。定義されない領域の強制衝突係数は、ゼロにセットされます。

このセクションは、6 つまで定義することができます。その書式は、

```
    [ 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
        ....           ........
        ....           ........
```

最初の行に `part =` の書式で粒子を指定します。省略した時のデフォルトは、`part = all` で全粒子です。ただし、電子、陽電子は強制衝突はできません。`part =` の書式の詳細はタリーの粒子指定と同じ書式です。しかし、指定できるのは `ityp` としての区別だけです。それぞれの原子核などは、個別には指定できません。

領域番号 (`reg`) と (`fcl`) の順番を変えたいときは、 `fcl reg` とします。読み飛ばしコラム用の non も使えます。GG の場合も、`cell` でなく `reg` を使ってください。

同じ値の領域をまとめて書く、`( { 2 - 5 } 8 9 )` という書式も使えます。

また、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造も 指定できます。ただし、単一の数字で無い場合は必ず `(   )` で括ってください。

この書式を用いれば lattice のひとつひとつに 異なる forced collision factor を定義することも可能です。上の定義で、同じセルが二重に定義された時は、初めに定義された値が採用されます。

強制衝突の係数 `fcl` は次の様な意味を持ちます。まず、ゼロは強制衝突をさせません。$|$`fcl`$| > 1$ はエラーです。$|$`fcl`$| \le 1$ の時は、強制衝突の確率を $|$`fcl`$|$ 倍します。その分、強制衝突粒子の重みを $1/|$`fcl`$|$ 倍します。

強制衝突領域での多重散乱と weight cutoff との関係では、次のふたつのオプションがあります。`fcl` < 0 の時、強制衝突による生成粒子は通常の衝突をさせます。その時 weight cutoff は行いません。`fcl` > 0 の時、強制衝突による生成粒子も更に強制衝突をさせます。この時、そのたびに weight cutoff を行います。また、タリーは weight cutoff の前に行います。この関係を良く理解して weight cutoff と強制衝突の係数を決めないと、強制衝突はさせたが出てきた粒子は全て殺されてしまうということもあり得ます。例えば、強制衝突で発生させた 2 次粒子の軌跡や強制衝突領域から離れた場所における粒子の情報をタリーする場合が該当します。この場合、`[parameters]` セクションにおいてウェイトカットオフ `wc1(i)` を低く設定する必要があります。

5.22 [Repeated Collisions] セクション

繰返し衝突 (repeated collisions) は、非弾性散乱の核反応（重イオン反応を含む核反応モデルを用いた高エネルギー核反応）が起こった時に、核反応のシミュレーションを繰り返し、生成物の統計的揺らぎを平均化する機能です。核反応チャンネルに確率は小さいが生成物の影響が大きい核分裂等があるときに有効です。生成物は繰返しの回数によりそのウェイトを変化させ、生成後は通常のウェイトカットオフ等の機能が有効になります。ただし、生成物のタリーは、ウェイトカットオフ等の処理の前に行われます。

繰返し衝突の領域と繰り返し係数等を以下のように指定します。このセクションは、6 つまで定義することができます。その書式は、

```
[ 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
```

最初の行に part=の書式で入射粒子を指定します。省略した時のデフォルトは、part=all で全ての粒子です。指定できる核反応は非弾性散乱の核反応（重イオン反応を含む核反応モデルを用いた高エネルギー核反応）です。核データを用いた核反応等は利用できません。次に、mother で核反応の標的核種を指定します。省略した時は、そのセルの全核種です。まず、個数を指定し、次の行で核種をその個数分指定します。質量数のない Pb 等の指定も可能です。次の emin と emax で、入射粒子のエネルギー範囲を指定します。省略した場合は全エネルギー領域が対象となります。次の行は、reg n-coll n-eval です。まず、reg の列で対象のセルを指定します。次の n-coll では核反応モデルでの繰り返し回数、n-evap では蒸発モデルでの繰り返し回数を指定します。

5.23 [Volume] セクション

このセクションでは、GG の cell の体積 (cm^3) を定義します。この量は、以下のタリーセクションで用いられます。両者で定義されない領域は、1.0 cm^3 にセットされます。書式は、

```
     [ V o l u m e ]
         reg                vol
           1              1.000000
          11              5.000000
  ( { 2 - 5 } 8 9 )       2.000000
     ( 11 12 15 )         3.000000
          16              6.000000
         ....             ........
         ....             ........
```

同じ値の領域をまとめて書く、`( { 2 - 5 } 8 9 )` という書式も使えます。ただし、単一の数字で無い場合は必ず `(      )` で括ってください。しかし、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造は、使えません。タリーで異なる universe でのセルの体積を細かく定義する時は、タリーセクションの中の volume 定義を用いてください。

領域番号 (`reg`) と体積 (`vol`) の順番を変えたいときは、 `vol reg` とします。読み飛ばしコラム用の non も使えます。GG の場合も、`cell` でなく `reg` を使ってください。

5.24 [Multiplier] セクション

このセクションで定義した係数は、[t-track], [t-cross], [t-point] の multiplier サブセクションで利用することができます。エネルギーの関数として係数を定義し、タリー結果にその係数を乗じます。例えば、任意の線量換算係数を用意し、その値を用いた場合の線量を評価する、といったことに利用できます。

ひとつの [multiplier] セクションにひとつの係数のセットを定義でき、用意できる [multiplier] セクションの数の上限は 100 です。書式は以下の様になります。

```
[ Multiplier ]
  number = -201
  interpolation = log
  part = all
  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
```

最初に、この係数をタリーの中で使うときの ID 番号を number で指定します。指定できる ID は、-200 から-299 までの負整数です。対象となる粒子は part で指定します。定義方法は、タリーの part 定義方法と同じで、all も使えます。同じ multiplier ID で様々な粒子に対するデータを定義したい場合は、それぞれの粒子に対して [multiplier] セクションを定義してください。データの内挿方法は interpolation で定義し、log-log 内挿の場合は log, linear-linear 内挿の場合は lin、内挿せず群定数として使う場合は glow もしくは ghigh と設定します。データ数は ne で指定し、その下に ne の数だけエネルギー点と係数の値をそれぞれ定義します。なお、エネルギー点は昇順で書く必要があり、glow あるいは ghigh の場合は、エネルギー点として群の下限あるいは上限値をそれぞれ設定します。

Version 3.03 より、いくつかの ID に対してデフォルトの値が設定されています。それらのデータは、/phits/data/multiplier/フォルダに格納されており、m+‘‘ID の絶対値’’+.inp というファイル名でそれぞれまとめられています（例: ID が-200 の場合は、m200.inp）。設定されているのは線量換算係数データ（単位 pSv·cm^2）と半導体ソフトエラー発生率（単位は (FIT/Mbit)/(/cm^2/s)）であり、表 90 に各 ID との対応関係をまとめています。このフォルダ内のデータは、ユーザーが自分で追加することも可能です。これらの ID をインプットファイルの [multiplier] セクションで再定義した場合は、再定義した値が優先されます。

半導体ソフトエラー発生率を評価するための mutiplier(ID=-299）には、仮想的な素子に対して PHITS とデバイスシミュレーターで評価した中性子 SEU 断面積を格納しています[74]。この素子に対するバックグランドソフトエラー発生率[75] はおおよそ 400 FIT/Mbit（2MB の SRAM で 18 年に 1 回ソフトエラーが発生す

[74] しきい電荷量を 0.6 fC と仮定し，中性子のエネルギー範囲として 1 MeV~1 GeV を考慮しています。詳細な計算方法は、S. Abe and Y. Watanabe, IEEE Trans. Nucl. Sci. 61, 3519-3526 (2014)、をご参照ください。

[75] 東京地表面における宇宙線フラックスを PARMA モデルで計算して推定しています。PARMA モデルは、http://phits.jaea.go.jp/expacs よりダウンロード可能です。

る頻度）となります。なお、FIT はソフトエラー率を表す単位で、1 FIT = 1e-9 err / hour です。SEU 断面積は半導体製品によって異なるため、実在の電子機器のソフトエラー率を正確に算出する目的には使用できません。あくまで一般的な地上環境におけるソフトエラー率との比較や、おおよその目安を把握する目的に使用してください。

表 90: 格納されている換算係数データの一覧

Multiplier ID	格納されているデータ
-200	$H^*(10)$[76]
-201	ICRP60 定義に基づく実効線量（前方照射）[77]
-202	ICRP103 定義に基づく実効線量（前方照射）[78]
-203	ICRP103 定義に基づく実効線量（等方照射）[78]
-204	新しい実用量 H^*（全ての照射条件で最大となる実効線量）[78]
-210	男女平均した実効線量当量（等方照射）[79]
-211	男性に対する実効線量当量（等方照射）[79]
-212	女性に対する実効線量当量（等方照射）[79]
-213	男性に対する造血組織線量当量（等方照射）[79]
-214	女性に対する造血組織線量当量（等方照射）[79]
-215	男性に対する皮膚線量当量（等方照射）[79]
-216	女性に対する皮膚線量当量（等方照射）[79]
-299	半導体ソフトエラー発生率（等方照射）[80]

5.24.1 Multiplier サブセクション

定義した係数をタリーで利用する場合は、[t-track], [t-cross], [t-point] の multiplier サブセクションを使います。基本的な表式は $(C \quad k)$ です。C を規格化定数、k を ID 番号（ただし $k < 0$）として指定します。具体的には、各タリーのセクションにおいて、以下の様な書式で使用します。

```
multiplier =  number of material
      part = neutron
      emax = 1000
    mat     mset1         mset2
      1   ( 1    -201 )  ( 2 -202 )
      2   ( 1.2  -201 )  ( 3 -202 )
    ....   ....      ....   ....
    ....   ....      ....   ....
```

multiplier =で指定する物質の数を決定します。all も可能ですが、この時は、下の mat の項でも all を用います。part =で係数を乗じる粒子を指定します。6 個まで複数指定可能で、all も利用できます。省略すれば all を指定したことになります。ただし、指定した粒子以外の寄与はゼロになります。emax =で係

[76] EXPACS (http://phits.jaea.go.jp/expacs/) よりデータを抜粋。
[77] 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) 他。
[78] ICRP Publication 116, Ann. ICRP 40(2-5), 2010.
[79] ICRP Publication 123, Ann. ICRP 42(4), 2013.
[80] S. Abe and Y. Watanabe, IEEE Trans. Nucl. Sci. 61, 3519-3526 (2014).

数を乗じるエネルギーの上限を設定します。省略した場合は、[multiplier] セクションで指定したエネルギーの最大値、もしくはライブラリを使う場合はその上限値がセットされます。それ以上では、その最大値に対する値が入ります。mat が係数を乗じる物質番号です。mset1, mset2 は multiplier セットの指定です。このセットは全部で 6 つまで指定可能で、それぞれのセット毎に結果が出力されます。また、ひとつのタリーセクション内に複数の multiplier サブセクションを指定できます。ただし、その時の multiplier セットの数は等しくなければなりません。

[multiplier] セクションで定義せずに使用できる係数もあります。$k = -1$ の場合は 1/weight を乗じ、モンテカルロ粒子の（すなわち、weight を常に 1 とした時の）結果を出力します。$k = -2$ の場合は 1/velocity を乗じます。$k = -120$ の場合は、物質の密度を乗じますので、icntl=5 と組み合わせれば領域内の質量が計算できます。また、phits/data/multiplier/フォルダ内にある ID 番号もインプットで定義せずに利用することができます。現在、整備されている ID 番号とデータの種類は表 90 をご参照ください。様々な種類の線量換算係数が格納されています。これらの単位は pSv·cm^2 ですので、フルエンスを (/cm^2/sec) に規格化すれば、(pSv/sec) 単位で被ばく線量を推定することができます。なお、重イオンに対する線量換算係数のエネルギー単位は核子あたりの MeV である MeV/n ですので、重イオンによる線量を計算したい場合は [parameters] セクションで iMeVperu=1 と設定してください。

この他、$k = -101, -102, -112, -114$ でそれぞれ陽子、中性子、電子、光子のセットが選択されます。これらの換算係数は、AP(前方-後方) 照射条件で評価された実効線量換算係数です[81]。これらの値は、ID 番号が −201 の換算係数とほぼ同等ですが、単位が (μSv/h)/(n/sec/cm^2) に規格化されている点が異なりますのでご注意ください。すなわち、フルエンスを (/cm^2/sec) で規格化した場合に、(μSv/h) 単位で被ばく線量を導出するように調整されています。(PHITS ver.2.00 から、線量換算係数のデータの内挿の仕方を linear-linear から log-log に変えましたので、注意してください。)

また、核データライブラリに含まれる断面積やカーマ係数などを読み込んで乗じることも可能です。その場合は、各 mset の書式が

```
( C  m  MT1  MT2  ... )
```

のようになります。ここで、C は規格化定数、m は [material] で定義した物質番号、MT1, MT2 は乗じたい係数の MT 番号です。2 つ以上の MT 番号を入力した場合は、基本的にはそれら MT 番号のデータの積が乗せられます。和を乗じたい場合は、加算したい MT 番号を:で繋ぎます。MT 番号とは、核反応の種類毎に決められた番号です。詳細は、https://wwwndc.jaea.go.jp/cgi-bin/ENDFfig?help=yes や https://www.oecd-nea.org/dbdata/data/manual-endf/endf102.pdf の Appendix B をご覧ください。その代表的なものを以下に示します。

MT 番号		説明
1	:	(z,total) 全反応断面積 [cm^2]
2	:	(z,z0) 弾性散乱断面積 [cm^2]
3	:	(z,nonelas.) 非弾性散乱断面積 [cm^2]
4	:	(z,n) 1 中性子生成断面積 [cm^2]
16	:	(z,2n) 2 中性子生成断面積 [cm^2]
17	:	(z,3n) 3 中性子生成断面積 [cm^2]
18	:	(z,fission) 核分裂断面積 [cm^2]
22	:	(z,nα) 中性子・α 粒子生成断面積 [cm^2]
27	:	(z,abs) 吸収断面積 [cm^2]
28	:	(z,np) 中性子・陽子生成断面積 [cm^2]

[81] 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)

102 : (z,γ) γ 線生成捕獲断面積 [cm^2]
103 : (z,p) 1 陽子生成断面積 [cm^2]
104 : (z,d) 1 重陽子生成断面積 [cm^2]
105 : (z,t) 1 三重陽子生成断面積 [cm^2]
106 : (z,^{3}He) 1^{3}He 生成断面積 [cm^2]
107 : (z,α) 1α 粒子生成断面積 [cm^2]
108 : (z,2α) 2α 粒子生成断面積 [cm^2]
109 : (z,3α) 3α 粒子生成断面積 [cm^2]

また、MT 番号の箇所に負値の特殊番号を入力することにより、下記に示す係数を乗じることも可能です。

中性子の場合

特殊番号		説明
−1	:	熱補正しない全断面積 [cm^2]
−2	:	吸収断面積 [cm^2]
−3	:	熱補正しない弾性散乱断面積 [cm^2]
−4	:	反応あたりの平均発熱量 [MeV/collision]
−5	:	γ 線生成断面積 [barn]
−6	:	全核分裂断面積 [cm^2]
−7	:	核分裂による平均中性子放出数 ν [個/fission]
−8	:	核分裂による平均発熱量 Q [MeV/fission]

光子の場合

特殊番号		説明
−1	:	コンプトン散乱（incoherent scattering）断面積 [cm^2]
−2	:	レイリー散乱（coherent scattering）断面積 [cm^2]
−3	:	光電効果断面積 [cm^2]
−4	:	電子対生成断面積 [cm^2]
−5	:	全反応断面積 [cm^2]
−6	:	反応あたりの平均発熱量 [MeV/collision]

以下、`mset` の例題をいくつか示します。

- `(100.0 5 1)`: 物質番号 5 番の全反応断面積を 100 倍して乗じる
- `(1.0 3 1 −4)`: 物質番号 3 番の中性子に対するカーマ係数（全反応断面積 x 平均発熱量）を乗じる
- `(1.0 4 −5 −6)`: 物質番号 4 番の光子に対するカーマ係数を乗じる（`part=photon` の場合）
- `(1.0 5 4:16:17)`: 物質番号 5 番の (n,n),(n,2n),(n,3n) 反応断面積の和を乗じる
- `(1.0 1 1 −4:−6 −8)`: 物質番号 1 番のカーマ係数（全反応断面積 x 平均発熱量）と核分裂生成熱（核分裂断面積 x Q 値）の和を乗じる。`[Cell]` の定義と同じく、積が和よりも優先されるため `(1 −4):(−6 −8)` と解釈されます。

5.25 [Mat Name Color] セクション

このセクションでは、gshow タリー、3dshow タリーで領域を表示する時の、物質番号毎に物質の名前と色、表示の文字の大きさを指定します。デフォルトは、名前は物質番号、色は物質の出現順に割り振ります。書式は、

```
[ 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
      ....      ........
      ....      ........
```

物質番号 (mat) と (name size color) の順番を変えたいときは、mat color size name の様に定義行で変えます。読み飛ばしコラム用の non も使えます。name color の何れかが定義されなければなりません。省略された場合は、デフォルトと見なされます。同じ名前、色の物質をまとめて書く、{ 4 - 7 } という書式も使えます。ただし、({ 4 - 7 } 9 10) という表式は使えません。文字の大きさは、デフォルトを 1 とした相対値で指定します。

名前に空白が入る時は、{ } で括ってください。また、名前の中で括弧 (,) を使うときは、¥(, ¥) としてください。名前の中で括弧 { , } は使えません。ただし、¥{ , ¥} とすることで、LaTeX 表記による上付きや下付き文字を使用できます。例えば、^¥{ 208 ¥}Pb とすると、^{208}Pb と表示されます。名前として定義できる文字数の上限は 80 です。

[cell] セクションにおいて同一の物質番号で密度の違う領域を複数定義した場合、デフォルト設定 (matadd=1) では、最初のもの以外の密度の物質に対して新しい物質番号を付与します。この番号は、file(6) (D=phits.out) の [material] セクションに出力されるので、その値を mat としてご使用ください。また、[parameters] セクションにおいて matadd=0 とすることで、同じ物質番号が割り振られるようになりますが、この機能は本セクションで指定する物質番号には影響がありません。この番号は file(6) の最初に Warning として出力されますので、その値をご参照ください。

色の指定は、ANGEL の書式に準じます。記号 (r bbb yy)、名前 (red orange blue)、また、HSB 数値 *H*(色相) *S*(彩度) *B*(明度) で指定できます。HSB 数値指定の時は { } で括ってください。HSB 数値が 1 個しか指定されない場合は、彩度、明度は 1 に設定されます。

次頁に、色指定の記号、名前、HSB 数値の表を示します。

表 91: グレースケール

記号	HSB定義	出力	名前
W	–1.0		white
O	–0.8		lightgray
K	–0.6		gray
J	–0.4		darkgray
F	–0.2		matblack
E	–0.0		black

表 92: 記号による色指定

記号	HSB定義	出力	名前
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

表 93: 名前、数値による色指定

名前	出力	HSB定義		
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] セクション

このセクションでは、[t-gshow]、[t-rshow]、[t-3dshow] で出力されるセル番号（数字）の表示を任意の名前に変更し、またその文字の大きさも変更できます。 他のタリーで gshow や rshow オプションを使用した場合の出力結果にも反映されます。 書式は、

```
[ Reg Name ]
          reg       name        size
            1       cover          1
            2        body        0.5
            3     {cell 2}         2
            4     {cell 3}         2
    { 5 - 8 }       tube           3
  ....            ........
  ....            ........
```

です。セル番号 (reg) と名前 (name) や文字の大きさ (size) の順番を変えたいときは、reg size name の様に定義行で変えます。読み飛ばしコラム用の non も使えます。name と size の何れかが定義されなければなりません。省略された場合は、デフォルトと見なされます。セル番号をまとめる { 4 - 7 } という書式も使えます。ただし、({ 4 - 7 } 9 10) という表式は使えません。名前に空白が入る時は、{ } で括ってください。名前の中で括弧 {, } を表示することはできませんが、¥{ ¥} とすることで、LaTeX 表記による上付きや下付き文字を使用できます。例えば、^¥{ 208 ¥}Pb とすると、^{208}Pb と表示されます。名前の中で括弧 (,) を使うときは、それぞれ ¥(, ¥) としてください。また、名前として定義できる文字数の上限は 80 です。文字の大きさは、デフォルトを 1 とした相対値で指定します。

5.27 [Counter] セクション

このセクションでは、タリーで用いるカウンターの設定をします。カウンターは 3 つ用意され、それぞれのカウンターで領域ごとに、その動作を定義します。カウンターの動作契機は大きく分けると 4 つあり、領域に入った時 (in)、領域を出た時 (out)、境界で反射を起こした時 (ref)、反応を起こした時 (coll) です。それぞれに、カウンターの進度 (-9999~9999)、もしくは、ゼロセット (10000) を定義できます。カウンターは粒子に付随して、散乱によって生成される粒子は、親のカウンターを引き継ぎます。カウンターの容量は、-9999 から 9999 まででそれ以下、以上になる時は変化しません。カウンターは、`part =` を使って粒子毎に指定できます。また、`*part =` とすれば、カウンターを動作させない粒子を指定できます。

書式は以下のようになります。、

```
[ C o u n t e r ]
   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
    ....            ........
    ....            ........
```

領域番号 (`reg`) と (`in out coll ref`) の順番を変えたいときは、 `reg coll in out ref` の様に定義行で変えます。読み飛ばしコラム用の non も使えます。`in out colls ref` の内、少なくともひとつは定義されなければなりません。省略された場合は、ゼロ、つまりカウントなしと見なされます。数字は、カウンターの進度を表します。この数だけカウンターに加えられます。10000 は、ゼロセットを意味します。ソース粒子のカウンター初期値はゼロです。

同じ値の領域をまとめて書く、`( { 2 - 5 } 8 9 )` という書式も使えます。また、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造も 指定できます。ただし、単一の数字で無い場合は必ず `(     )` で括ってください。

`part=`での粒子指定では、粒子は 20 個まで指定できます。原子核は、`208Pb` のように質量数を指定すればその核、`Pb` のように質量数を指定しなければ、`Pb` の同位体全体を指定することになります。

バージョン 2.90 以降、反応を起こした時 (coll) に属するイベントを細かく分類し、カウンターの動作契機を増やしました。まず、coll は、原子核反応を起こした時 (nucl)、原子相互作用により粒子を生成した時 (atom)、崩壊を起こした時 (dcay) の 3 つに分類されます。そして、nucl と atom は更にそれぞれ 3 つと 10 つに分かれます。表 94 に coll) に属する各種のイベントを示します。これらの反応イベントを詳細に分析したい場合は、カウンターの動作契機として各反応のキーワードを指定してください。ただし、例えば、coll と nucl を同時に指定した場合は、重複してカウントされますので、ご注意ください。

fiss を設定すると、散乱（核反応）が起こり、更に核反応として核分裂が起こった時だけカウンターが呼ばれます。核分裂が起こったかどうかの判定は、核データを使う時は核データの fission チャンネルが選択された時、反応モデルを使う時は統計崩壊過程において fission チャンネルが選ばれた時です。

表 94: coll に属するイベントの分類

相互作用	反応イベント	説明
nucl		原子核反応を起こした時。
	elst	弾性散乱を起こした時。
	iels	非弾性散乱を起こした時。
	fiss	核分裂を起こした時。
atom		原子相互作用により粒子を生成した時。
	delt	デルタ線を生成した時。
	knoe	電子と陽電子による knock on electron 生成が起こった時。
	fluo	蛍光 X 線を生成した時。
	auge	Auger 電子を生成した時。
	brem	制動放射を起こした時。
	phel	光電効果を起こした時。
	cmpt	コンプトン散乱を起こした時。
	pprd	電子陽電子対を生成した時。
	anih	陽電子が消失した時。
	rayl	レイリー散乱を起こした時。
dcay		崩壊を起こした時。

5.28 [Timer] セクション

このセクションでは、各粒子が持っている時計のリセット、ストップをコントロールします。タイマーは、領域ごとに、その動作を定義します。タイマーの動作契機は 4 つあり、領域に入った時、領域を出た時、散乱を起こした時、境界で反射を起こした時です。それぞれに、ゼロリセット (-1)、ストップ (1)、何もしない (0) の動作が定義できます。書式は、

```
[ T i m e r ]
    reg     in     out    coll     ref
      1      0      -1       0       0
     11      1       0       0       0
    ....   ....    ....    ....    ....
    ....   ....    ....    ....    ....
    ....   ....    ....    ....    ....
```

領域番号 (reg) と (in out coll ref) の順番を変えたいときは、 reg coll in out ref の様に定義行で変えます。読み飛ばしコラム用の non も使えます。in out coll ref の内、少なくともひとつは定義されなければなりません。省略された場合は、ゼロ、つまり何もしません。

同じ値の領域をまとめて書く、({ 2 - 5 } 8 9) という書式も使えます。また、(6 < 10[1 0 0] < u=3) などの lattice, universe 構造も 指定できます。ただし、単一の数字で無い場合は必ず (　　) で括ってください。

6 タリー共通パラメータの書式

タリーセクションには、以下のセクションがあります。

表 95: タリーセクションの種類

name	説明
[t-track]	粒子の飛跡長 (track length) やフルエンスを導出するタリー。
[t-cross]	粒子の面横断回数やフルエンスを導出するタリー。
[t-point]	ある点や線上のフルエンスを導出するタリー。
[t-deposit]	物質におけるエネルギー付与を導出するタリー。
[t-deposit2]	2 つの領域でのエネルギー付与の相関を出力するタリー。
[t-heat]	物質におけるエネルギー付与を導出するタリー。（非推奨[82]）
[t-yield]	残留核の生成量を導出するタリー。
[t-product]	線源や核反応による生成粒子を導出するタリー。
[t-dpa]	原子あたりのはじき出し数 (DPA) を導出するタリー。
[t-let]	LET の関数として飛跡長や線量を導出するタリー。
[t-sed]	微小領域におけるエネルギー付与分布を導出するタリー。
[t-time]	Energy cut off と escape 粒子の個数を導出するタリー。
[t-interact]（従来の [t-star]）	反応数を導出するタリー。
[t-dchain]	DCHAIN 用入力ファイルを作成するタリー。
[t-wwg]	[weight window] のパラメータを出力するタリー。
[t-wwbg]	[ww bias] のパラメータを出力するタリー。
[t-volume]	体積自動計算機能のためのタリー。
[t-userdefined]	ユーザー定義による任意の物理量を導出するタリー。
[t-gshow]	ジオメトリ（仮想空間）を 2 次元で表示するタリー。
[t-rshow]	ジオメトリを 2 次元で物理量による色分けをして表示するタリー。
[t-3dshow]	ジオメトリを 3 次元で表示するタリー。

以下では、これらのタリーセクションで共通に使われるパラメーターの定義の仕方を説明します。

6.1 形状メッシュ

上に示した各タリーでは、データを収集する幾何形状の種類として、GG 領域メッシュ(reg)、r-z スコアリングメッシュ(r-z)、xyz スコアリングメッシュ(xyz) が全て使えます。他に [t-track]、[t-desposit]、[t-yield]、[t-product]、[t-dpa]、[t-dchain] でのみ使用可能な四面体メッシュ(tet) があります。この 4 種類のメッシュを

```
mesh =  [ reg,  r-z,  xyz,  tet ]
```

で定義します。

[82] バージョン 3.04 までは [t-deposit] によるカーマ近似計算ができなかったため、この計算が可能な [t-heat] との使い分けを行っていました。

6.1.1 領域メッシュ

領域番号、もしくはセル番号で指定する領域メッシュは次の様に定義します。

```
mesh = reg
 reg = 1 2 3 4 5 ( 10 11 ) 50
```

領域番号、もしくは、セル番号は、空白区切りで指定します。

幾つかの領域をまとめたい時は、`(  )` を用います。このまとめた領域の識別番号はエコーに表示されます。

連続した領域番号では、次の様な書式も使えます。

```
mesh = reg
 reg = { 1 - 5 } ( 10 11 ) ( 6 < 10[1 0 0] < u=3 )
```

`( n1 - n2 )` は、使えません。`( { } )` は可能ですが、`{ ( ) }` は不可です。`( all )` も可能です。また、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造も 指定できます。この書式を用いれば lattice のひとつひとつを 個別にタリーすることも可能です。また、`reg = all` という書式で領域を定義すると全ての領域を対象とします。ただし、最下層でないセルは除かれます。

6.1.2 階層構造の領域と体積の定義

階層構造を含む領域を指定する際は、必ず `(  )` で書式を括らなければなりません。階層は `<` で表します。中間の階層が lattice 構造を持つ場合は、そのセル番号の後に `[  ]` を用いてどの lattice か指定することができます。例えば、`160[1:2 3:6 1:1]` の様に指定した場合は、lattice 座標 (s,t,u) で表現した $1 \le s \le 2, 3 \le t \le 6, u = 1$ の範囲に含まれる $2 \times 4 \times 1 = 8$ の lattice を意味します。個別に幾つか指定する時はコンマを用い、`160[1 3 4, 2 3 4, 3 3 4]` とします。ひとつの階層の中で `(  )` を用いると、それは括弧内の領域をまとめることを意味します。次の例題を見てみましょう。

例 52: `mesh = reg` の例題 (1)

```
mesh = reg
reg = (all)
      ({ 201 - 205 })
      ( 161 < 160[1:2 3:6 1:1] )
      ( (201 202 203 204) < (161 162 163 ) )
      ( ( 90 100 ) 120 < 61 ( 62 63 ) )
```

このメッシュ定義文をタリーで用いてエコーを取ると次のものが得られます。

例 53: `mesh = reg` の例題 (2)

```
mesh =  reg             # mesh type is region-wise
 reg = ( all ) ( { 201 - 205 } ) ( 161 < 160[ 1:2 3:6 1:1 ] ) ( (
       { 201 - 204 } ) < ( { 161 - 163 } ) ) ( ( 90 100 ) 120 < 61
       ( 62 63 ) )
volume                  # combined, lattice or level structure
```

```
         non   reg      vol     # reg definition
           1  10001   8.1000E+01 # ( all )
           2  10002   5.0000E+00 # ( { 201 - 205 } )
           3  10003   1.0000E+00 # ( 161 < 160[ 1 3 1 ] )
           4  10004   1.0000E+00 # ( 161 < 160[ 2 3 1 ] )
           5  10005   1.0000E+00 # ( 161 < 160[ 1 4 1 ] )
           6  10006   1.0000E+00 # ( 161 < 160[ 2 4 1 ] )
           7  10007   1.0000E+00 # ( 161 < 160[ 1 5 1 ] )
           8  10008   1.0000E+00 # ( 161 < 160[ 2 5 1 ] )
           9  10009   1.0000E+00 # ( 161 < 160[ 1 6 1 ] )
          10  10010   1.0000E+00 # ( 161 < 160[ 2 6 1 ] )
          11  10011   4.0000E+00 # ( ( { 201 - 204 } ) < ( { 161 - 163 } ) )
          12  10012   2.0000E+00 # ( ( 90 100 ) < 61 )
          13  10013   1.0000E+00 # ( 120 < 61 )
          14  10014   2.0000E+00 # ( ( 90 100 ) < ( 62 63 ) )
          15  10015   1.0000E+00 # ( 120 < ( 62 63 ) )
```

入力では 5 つの領域の様に見えますが、エコーを見るとタリーを取る領域として 15 の領域が定義されていることが分かります。それぞれ単一の領域番号で表されないので、新たに 10000 番台の番号を付けてその体積と領域の定義文がエコーされます。この場合、[volume] セクションが無いので各セルの体積は 1 と 定義されています。

まず、`( all )` は、この例題では 81 個の最下層のセルが定義されているので、この領域の体積は 81 となっています。もし最下層のセルの体積が正確に `[volume]` セクションで 定義されていれば、この領域の体積は入力しなくても正確に与えられます。次に `({ 201 - 205 })` は、201 から 205 までの領域をまとめたものですから、体積として 5 がエコーされています。これも `[volume]` セクションで正確に体積が 定義されていれば、この領域の体積は入力しなくてもかまいません。

次に、`( 161 < 160[1:2 3:6 1:1] )` ですが、これは、160 の領域の中に 161 の領域が lattice として組まれています。ここでは、lattice 座標の s に関して 1 から 2、t に関して 3 から 6、u に関して 1 の各 lattice に含まれているセル番号 161 の計 8 個の領域でタリーを取ることを意味します。エコーでは、最下層の領域の数 1 が体積としてエコーされています。この様な場合は、次に示す体積定義文で体積を与えなければなりません。

次は、`( (201 202 203 204) < (161 162 163 ) )` ですが、これは、各階層に幾つかの領域が記述されていますが、全て括弧で括られてひとつの領域にされているので、全体としてもひとつの領域を表します。エコーの体積は、最下層の領域の和としているので正しくはありません。これも volume 定義文で与えなければなりません。

最後の `( ( 90 100 ) 120 < 61 ( 62 63 ) )` は、各階層に 2 つずつ独立な領域があるので計 4 個の領域に分割されます。それぞれの体積はやはり volume 定義文で与える必要があります。

volume 定義文は、インプットエコーの書式をそのまま使えます。書式は、

```
  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
```

上の例題では、`reg` として、1 から 4 まではそのまま、`( 5 < 12 ) ( {13 - 17} )` は、10001 と 10002 を用います。この番号は、一度インプットエコーを取ると得られます。上の例題より分かるように、インプットエコーの部分をそのままコピーし、体積の値だけを変更すれば済みます。

領域番号 (`reg`) と体積 (`vol`) の順番を変えたいときは、 `vol reg` とします。読み飛ばしコラム用の `non` も使えます。インプットエコーでは、`non` として通し番号が付けられています。`axis = reg` の時の タリーの出力の際にはこの番号が X 軸の値として出力されます。GG の場合も、`cell` でなく `reg` を使ってください。

最下層の領域である場合、その領域を指定する時は、`( 3000 < 3000[1:2 3:6 1:1] )` の様に、同じ領域を階層構造の様に 2 度記述します。

6.1.3 r-z メッシュ

r-z スコアリングメッシュの場合、まず、円柱の中心の x, y 座標のオフセットを

```
mesh = r-z
  x0 = 1.0
  y0 = 2.0
```

の書式で定義します。これは省略可能でデフォルト値はゼロです。

次に、

```
mesh   = r-z
r-type = [1-5]
   ..........
   ..........
z-type = [1-5]
   ..........
   ..........
```

で始まるメッシュ定義文によって r と z とのメッシュを定義します。メッシュ定義文は後述します。

`[t-track]` タリーに関しては、既存の r, z メッシュに加え、θ メッシュを指定することができます。ここで、r, θ, z はそれぞれ円柱座標系における

r: 円の半径

θ: 円面における角度

z: 円柱の中心軸上の位置

であり、θ を定義する方位角の基線は x 軸です。`[t-track]` において θ に関するメッシュを指定する際は、`a-type` を用いてください。`a-type=1,2,-1,-2` のタイプがありますが、これらの数字が正の場合は角度の単位を余弦で与え、負の場合は degree 単位で与えることになります。ただし、円柱の定義であるので、0 度から 360 度までしか指定できません。また、角度のメッシュを加えたことにより、角度を横軸にして出力できるようになりました。そのために `axis` として、`rad` と `deg` を指定できます。`rad` は radian で、`deg` は degree の単位で出力します。

6.1.4 xyz メッシュ

xyz スコアリングメッシュの場合、

```
mesh   = xyz
x-type = [1-5]
   ..........
   ..........
y-type = [1-5]
   ..........
   ..........
z-type = [1-5]
   ..........
   ..........
```

で始まるメッシュ定義文によって x, y, z のメッシュを定義します。メッシュ定義文は後述します。

6.1.5 四面体メッシュ

連続四面体（四面体メッシュ）形状 5.6.5.4 を使用している場合にのみ有効で、この体系内の各四面体で物理量をタリーすることができます。但し、現状では [t-track]、[t-deposit]、[t-yield]、[t-product]、[t-dpa] でのみ使用可能です。書式は、

```
mesh   = tet
 reg   = 100
```

のように mesh の指定の次の行で領域番号 (reg) を指定します。ここで領域番号として、連続四面体形状を収めた直方体領域を指定してください。こうすることで、この直方体領域に定義した全ての四面体に対して、個別に物理量がタリーされます。一部の四面体のみを抽出したり合計を求めたりすることは、四面体メッシュではできません。代わりに領域メッシュ(mesh=reg) で四面体要素の Univese を構成するセルを指定することにより、一部の目的を達することができます。また、四面体の一部をくり抜いて使用している場合、四面体の体積を自動で計算しているため、正しい物理量が導出されない恐れがあります。この場合も領域メッシュ(mesh=reg) を使用して回避してください。

6.2 エネルギーメッシュ

タリーのエネルギーメッシュは

```
e-type = [1-5]
   ..........
   ..........
```

で始まるメッシュ定義文によって定義します。DEPOSIT2 では、e1-type, e2-type で二つのエネルギーメッシュを定義します。メッシュ定義文は後述します。

6.3　LET メッシュ

タリーの LET メッシュは

```
l-type = [1-5]
   ..........
   ..........
```

で始まるメッシュ定義文によって定義します。メッシュ定義文は後述します。

6.4　時間メッシュ

タリーの時間メッシュは

```
t-type = [1-5]
   ..........
   ..........
```

で始まるメッシュ定義文によって定義します。メッシュ定義文は後述します。

6.5　角度メッシュ

cross, product タリーで用いる角度メッシュは

```
a-type = [1, 2, -1, -2]
   ..........
   ..........
```

で始まるメッシュ定義文によって定義します。この定義により、余弦 (cos) もしくは角度 (degree) を定義します。a-type を正で定義したときは余弦 (cos)、負で定義したときは角度 (degree) を表します。メッシュ定義文は後述します。

6.6 メッシュ定義文

メッシュ定義文には、e-type，t-type，x-type，y-type，z-type，r-type，a-type，l-type で始まる 8 種類がありますが、最初の一文字を除くと全て共通なので、ここでは e-type を例に解説します。ne を nx，ny，nz，...、emin を xmin，ymin，zmin，...、の様に読み換えてください。

定義したメッシュの最小のビンに emin の値自身は含まれます。一方、最大のビンに emax の値自身は含まれません。

6.6.1 メッシュタイプ

メッシュの指定の仕方は 5 種類あり、それを e-type = [1-5] で指定します。1 から 5 番までのそれぞれのメッシュタイプは以下のようなもです。

表 96: メッシュタイプ

メッシュタイプ	内容
1	群数、分点をデータで与える。
2	群数と最小値、最大値を与え、線形で等分点が与えられる。
3	群数と最小値、最大値を与え、対数で等分点が与えられる。
4	メッシュ幅と最小値、最大値を与え、線形で等分点が与えられる。このとき、分点の最大値は与えた最大値もしくは最大値を超える最小の値となるように群数が決定される
5	最小値、最大値、メッシュ幅の対数値とを与え、対数で等分点が与えられる。このとき、分点の最大値は与えた最大値もしくは最大値を超える最小の値となるように群数が決定される

ただし、a-type 即ち、角度のメッシュでは、1, 2 (-1, -2) のメッシュタイプしか使えません。

[t-cross] で z-type=1 とした場合のみ、群数 nz を 0 としてデータ点を 1 つだけ指定することが可能です。この場合、z =（データ点）とする面のみがタリー面として設定されます。

以下に、それぞれのメッシュタイプのデータの定義の書式を示します。

6.6.2 e-type = 1 の場合

書式は、

```
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)
```

です。この際、継続行のシンボル無しに複数行にまたがっても自動判別します。

6.6.3 e-type = 2, 3 の場合

この次に群数、最小値、最大値を次の様な書式で与えます。

```
e-type = 2, 3
    ne =  number of group
  emin =  minimum value
  emax =  maximum value
```

6.6.4 e-type = 4 の場合

この次にメッシュ幅、最小値、最大値を次の様な書式で与えます。

```
e-type = 4
  edel =  width of mesh
  emin =  minimum value
  emax =  maximum value
```

6.6.5 e-type = 5 の場合

この次にメッシュ幅、最小値、最大値を次の様な書式で与えます。

```
e-type = 5
  edel = log(  width of mesh  )
  emin =  minimum value
  emax =  maximum value
```

この場合、メッシュ幅は対数上での幅、即ち edel = log(M_{i+1} / M_i) です。

6.7 他のタリー定義文

6.7.1 粒子定義文

タリーの中で粒子を指定する時は、

```
part = proton neutron pion+ 3112 208Pb
```

のように空白で区切って定義するか、

```
part = proton
part = neutron
part = pion+
part = 3112
part = 208Pb
```

のように、定義文を繰り返すことも出来ます。粒子名の表式は表 4 を参照してください。kf コード番号での指定も可能です。

```
part = all
```

と定義すると、粒子すべての和を表わします。ひとつのタリーで粒子 6 個まで指定できます。それ以上の粒子を指定したい時は、複数のタリーセクションを用いてください。

複数の粒子をひとつのグループとしてタリーしたいときは、次のように `( )` が使えます。`( )` の中の粒子は、最大 6 個まで指定できます。

```
part = ( proton neutron ) all pion+ 3112 208Pb
```

この場合、proton と neutron を合わせたものが、最初のグループとして出力されます。2 番目は、全ての粒子の和です。全体として 5 種類の粒子についての出力になります。原子核は、`208Pb` のように質量数を指定すればその核、`Pb` のように質量数を指定しなければ、`Pb` の同位体全体を指定することになります。

6.7.2 axis 定義文

出力データの x 軸を定義します。`axis` の種類は、タリーの種類とタリーの形状メッシュによりますが、

```
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
```

があります。

```
axis = eng
```

のように定義します。

ひとつのタリーで複数の `axis` が定義できます。

```
axis = eng  x  y
```

とするか、

```
axis = eng
axis = x
axis = y
```

のように表わすことも出来ます。ひとつの axis に対して、ひとつのファイルに結果が 出力されます。従って、複数の axis を定義した場合は、次のファイル定義文で同数のファイル名を定義する必要があります。なお、[t-yield] では、1 つのタリーに 1 つの axis しか指定できません。複数の axis に対して残留核収率を計算したい場合は、複数の [t-yield] タリーを作成してください。これは、統計誤差を正しく計算するためにバージョン 2.50 以降についた制限です。

6.7.3 file 定義文

出力ファイル名を定義します。書式は、

```
file = file.001  file.002  file.003
```

のように、パス名を含めてファイル名を書きます。ただし、“.eps” や “.vtk” の拡張子をもつ名前は指定しないようにしてください。axis を複数指定したときは、その数だけファイル名を指定します。このとき

```
file = file.001
file = file.002
file = file.003
```

のように一行にひとつずつ書くこともできます。

6.7.4 resfile 定義文

再開始計算時の過去タリーファイル名を指定します。書式は

```
resfile = file.001
```

のようにパス名を含めてファイル名を書きます。file と違うのは、複数タリーを指定しても、resfile は 1 つで十分です。2 つ以上指定した場合、先に書いたファイルが優先されます。resfile のデフォルトは、file です。その場合、過去タリーファイルから結果を読み込んで、新しい結果を加えて同じファイルに上書きします。

6.7.5 unit 定義文

出力の単位を定義します。通常番号で次の様に定義します。

```
unit = number
```

番号とその単位の内容は、各タリーの説明で解説します。

6.7.6 factor 定義文

出力の規格化定数を定義します。[t-gshow] タリーでは、境界線の太さを指定します。次の様に定義します。

```
factor = number
```

出力の物理量にこの定数が掛けられます。

6.7.7 output 定義文

出力する情報の種類を定義します。次の様に定義します。

```
output = name of output
```

詳細は、各タリーの説明で行います。

6.7.8 info 定義文

出力するタリー出力で、詳細情報を出力するかどうかのオプションです。通常、0 か 1 で定義します。

```
info = 0, 1
```

6.7.9 title 定義文

タリー出力に表示されるタイトルを定義します。

```
title = title of the tally
```

省略可能です。省略された場合、デフォルト値が入ります。

6.7.10 ANGEL パラメータ定義文

タリー出力で、ANGEL のパラメータを追加します。

```
angel = xmin(1.0) ymin(1.3e-8)
```

ここで定義したパラメーターは、タリー出力の中で

```
p: xmin(1.0) ymin(1.3e-8)
```

と記載され、横軸と縦軸の最小値をそれぞれ指定した値に変更します。
ANGEL については、A をご覧ください。

6.7.11 2d-type 定義文

タリー出力で、`axis = xy` などの 2 次元表示を選択したときの、2 次元データの表示のオプションです。番号で指定します。ただし、rshow オプションがあるタリーでは、意味を持ちません。

```
2d-type =  1, 2, 3, 4, 5, 6, 7
```

- `2d-type = 1, 2, 3, 6, 7`
  これは、データの並びが FORTRAN の書式で書くと

  ```
  ( ( data(ix,iy), ix = 1, nx ), iy = ny, 1, -1 )
  ```

  で、1 行に 10 個のデータが入り、ANGEL 用のヘッダーが付きます。ヘッダーは、1 は、等高線、2 は、クラスタープロット、3 は、カラープロットです。6 は、クラスターと等高線です。7 は、カラープロットと等高線です。

- `2d-type = 4`
  これは、データの並びが FORTRAN の書式で書くと

  ```
  do iy = ny, 1, -1
  do ix = 1, nx
     ( x(ix), y(iy), data(ix,iy) )
  end do
  end do
  ```

  で、1 行に `x(ix), y(iy), data(ix,iy)` の 3 個のデータが入ります。

- `2d-type = 5`
  これは、データの並びが FORTRAN の書式で書くと

```
y/x  ( x(ix), ix = 1, nx )
  do iy = ny, 1, -1
     ( y(iy), data(ix,iy), ix = 1, nx )
  end do
```

で、1 行に nx + 1 個のデータが並び、全部で ny + 1 行です。Excel 等に取り込むデータとして便利です。

6.7.12 gshow 定義文

[t-gshow], [t-rshow] 以外のタリーで用いることができます。タリーのメッシュが xyz メッシュ、axis が xy, yz, xz で、かつ 2d-type = 1,2,3,6,7 の ANGEL の出力を 想定したタリーで、これを指定すると、出力の画面に領域の境界線、また物質番号、領域番号、LAT 番号が表示されます。

```
gshow =  0, 1, 2, 3, 4
```

0 で表示無し、1 は境界線の表示、2 は境界線と共に物質番号、3 は境界線と共に領域番号、4 は境界線と共に領域番号と LAT 番号、を表示します。物質番号、領域番号、LAT 番号を表示させるときに、resol を使って分解能を resol 倍にすると、番号の表示が乱れますので、番号を表示させるときは、resol でなくメッシュの個数を増やして分解能を上げてください。パラメータセクションで、icntl = 8 とすることにより、 実際の計算をすることなしに、タリーを設定した領域を事前に見ることができます。icntl = 8 とすると、輸送計算はせずに、xyz メッシュ、xy, yz, zx, axis のタリーで gshow = 1, 2, 3, 4 が指定してあるものの領域を指定のファイルに書き出します。その時、物質毎に色分けします。この機能を使い、大きい計算の前に、タリーの領域の確認、xyz メッシュの分解能の適否を確認することをお勧めします。

6.7.13 rshow 定義文

[t-cross] と [t-gshow] 以外のタリーで用いることができます。タリーの mesh が reg メッシュの場合に利用でき、領域毎に求めた物理量の大きさに応じた色を幾何形状表示させた体系に付与します。axis は xy, yz, xz のどれかを指定する必要があり、更に rshow 定義文の下に xyz のメッシュ定義文が必要となります。rshow の値を変えることで、境界線、物質番号、領域番号を表示させることができます。

```
rshow =  1, 2, 3
x-type = [2,4]
   ..........
   ..........
y-type = [2,4]
   ..........
   ..........
z-type = [2,4]
   ..........
   ..........
```

0 で表示無し、1 は境界線の表示、2 は境界線と共に物質番号、3 は境界線と共に領域番号、を表示します。0 の時は、xyz メッシュ定義文は、不必要なのでコメントアウトして下さい。 物質番号、領域番号を表示させるときに、resol を使って分解能を resol 倍にすると、番号の表示が乱れますので、番号を表示させるときは、resol でなくメッシュの個数を増やして分解能を上げてください。

reg メッシュでこのオプションを付けると、領域ごとの出力はありません。従って、計算の後に図の体裁や最大値最小値の調整などは、元のデータがありませんからできません。このオプションを付ける時は、axis = xy,yz,zx で用いるわけですが、その他に axis = reg も加えて 別ファイルに領域の物理量のデータを保存することをお勧めします。そのデータと、[t-rshow] タリーを用いることにより、再度加工したデータを元に表示することが可能となります。

パラメータセクションで、icntl = 10 とすることにより、実際の計算をすることなしに、タリーを設定した領域を事前に見ることができます。icntl = 10 とすると、輸送計算はせずに、reg メッシュ、xy, yz, zx, axis のタリーで rshow = 1, 2, 3 が指定してあるものの領域を 指定のファイルに書き出します。その時、物質毎に色分けします。この機能を使い、大きい計算の前に、タリーの領域の確認、xyz メッシュの分解能の適否を確認することをお勧めします。

6.7.14 x-txt, y-txt, z-txt 定義文

ANGEL 表示の x, y, z 軸のテキストをデフォルトから変えたい時に用います。これらのテキストは、ANGEL パラメータ定義文では、変えられません。

```
x-txt = テキスト
y-txt = テキスト
z-txt = テキスト
```

6.7.15 volmat 定義文

volmat パラメータは、xyz メッシュでメッシュが領域境界を またいでいる時の体積補正をするものです。これが有効になるのは、xyz メッシュでかつ material の指定がある場合です。領域をまたいだメッシュの体積を、メッシュサイズから与えるのではなく、指定された material を含む体積をモンテカルロ的に計算します。その時のスキャンは、各軸に平行な 1 辺当たり volmat 数の軌跡で計算します。この数をあまり大きく取るとメッシュ数にもよりますが、計算時間が膨大になることがありますので注意して下さい。volmat を負の数で指定すると、強制的に全ての xyz メッシュをスキャンします。正の場合は、メッシュの 8 頂点が同じ物質ならスキャンしません。

6.7.16 epsout 定義文

`epsout = 1` を指定すると、出力ファイルを ANGEL で処理した eps ファイルを作成します。ファイル名は出力ファイルの拡張子を eps に変えたファイル名です。パラメータセクションで `itall = 1` を指定して、出力の eps ファイルを ghostview を 用いて画面に表示し、`w` キーを押して eps ファイルの更新に伴う画面の書き換えを指定していれば、バッチ毎にタリーの結果の変化を自動的に画面上で確認できます。

6.7.17 カウンター定義文

[counter] セクションで定義したカウンターを用いて、タリーで集める物理量に制限を加えることができます。各カウンターごとに、最小値と最大値を ctmin(i), ctmax(i) で定義します。i は、カウンター番号、1 から 3 です。デフォルト値は最小値が −9999、最大値が 9999 です。複数のカウンターを用いる時は、それらの条件の共通部分となります。

また、あるヒストリー中に発生する全粒子のカウンター値の最大値を使って制限を加えることも可能です。その場合は、chmin(i), chmax(i) を使います。この機能を使えば、特定のイベントが発生した場合に線源発生まで遡ってタリーすることが可能となります。詳しくは "/phits/sample/misc/history_counter" フォルダをご参照ください。ただし、この最大カウンター制限機能は、バッチ分散モード（istdev=1）では動作しません。また、version 3.24 において、[t-track] と [t-deposit] でのみ利用可能です。

6.7.18 resol 分解能、width 線太さ定義文

resol を使い、gshow、rshow、3dshow の表示の時、指定した xyz のメッシュを固定したまま、境界線を求める分解能を上げることができます。デフォルトは 1 で、xyz のメッシュの分解能と同じです。resol = 2 とすると各辺 2 倍のメッシュになります。xyz メッシュのタリーで、境界線の精度だけ上げるのに便利です。また、3dshow の時も、荒い精度で表示を確認してから、resol を大きくして最終的なきれいな図を得ることができます。resol を大きくしてもメモリーは変化しません。ただし、物質番号、領域番号、LAT 番号を同時に表示させるときには、resol を使って分解能を resol 倍にすると、番号の表示が乱れますので、番号を表示させるときは、resol でなくメッシュの個数を増やして分解能を上げてください。

width は、gshow、rshow、3dshow の表示の時の線の太さを定義します。デフォルトは、0.5 です。

6.7.19 trcl 座標変換

r-z、xyz メッシュの座標を trcl により座標変換します。書式は以下のように 2 通りあります。

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

最初の書式は、[transform] セクションで定義した座標変換番号です。次の書式は、この定義の中で座標変換を定義します。[transform] と同じように 13 個の数字で定義します。一行に収まらない場合は、複数の行にわたって定義できます。その時は自動認識しますので、行末の継続行の記号 ¥は必要ありません。ただし、継続行は先頭に 12 個以上の空白が必要です。

r-z、xyz メッシュの他に、3dshow のタリーで box を定義するときにも、trcl による座標変換が使えます。書式は [t-3dshow] のところで示します。

6.7.20 dump 定義文

[t-cross], [t-time], [t-product] タリーで、粒子の情報をファイルにダンプすることができます。その時、ダンプする物理量とそのデータ並びを dump 定義文で指定できます。

dump データの情報は、まず、dump = でデータの個数を指定します。正で与えた時は、バイナリーファイル、負で与えた時は、アスキーファイルへの書き込みをします。次の行にひとつのレコードのデータの並びを指定します。その時の番号と物理量の関係は、以下の通りです。

表 97: dump データの種類と番号 (1)

物理量	kf	x	y	z	u	v	w	e	wt	time	c1	c2	c3	sx	sy	sz
番号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

表 98: dump データの種類と番号 (2)

物理量	name	nocas	nobch	no
番号	17	18	19	20

ここで、kf は粒子を識別する kf-code (表 4 を参照してください)、x, y, z は座標 (cm)、u, v, w は運動量の単位ベクトル、e はエネルギー (MeV)、原子核の場合は核子当たりのエネルギー、wt は粒子ウエイト、time は時間 (nsec)、c1, c2, c3 はカウンターの値、sx, sy, sz はスピンの方向ベクトルです。name は粒子の衝突回数、nocas はバッチの中の現在のイベント数、nobch は現在のバッチ数、no は現在のイベントでのカスケード ID です。これらはバイナリーファイルの時は real*8 のデータで、アスキーファイルの時は、*n*(1p1d24.15) のフォーマットで格納します。

例えば、9 つのデータを次の順番で書き出させたいとき、

```
kf  e  wt  x  y  z  u  v  w
```

このデータを書き込みたいときは、

```
dump = 9
  1  8  9  2  3  4  5  6  7
```

と指定します。

dump 定義文を用いてダンプさせる時は、axis、file 数ともにひとつに制限されます。また、unit は 1 に固定されます。dump するデータを書き込むファイルは、file 定義文で指定した名前に “_dmp” を付けたファイルです。file 定義文で定義された file 名には、通常のタリーの結果が書き出されます。この結果からダンプされたデータの規格化定数を読み取ってください。(バージョン 2.66 以前では、file 定義文で指定したファイルに dump するデータが書き込まれ、通常のタリー結果を出力するファイルとして`.cfg` をつけたファイルが作成されていました。) メモリ分散型並列計算時には使用する並列 PE (Processor Element) 数 −1 個のファイルを作成します。データを dump するファイルの名前の最後に PE 番号をつけたファイルが作られ、各 PE は対応するファイルにのみデータの書き出しと読み込みを行います。

6.8 複数のタリー結果の統合機能

Sumtally 機能を使うことにより、複数のタリー結果を足し合わせることができます。足し合わせの方法は 2 種類あり、手動で並列計算を実行するために各タリー結果のヒストリー数を考慮して足し上げる方法と、各タリー結果に任意の重み付けをして加重平均を求める方法があります。前者は、並列計算が利用できず、限られた計算資源を有効に利用したい場合に役立ちます。後者は、例えば複数の線源による計算を行う際、それらの強度を任意に変えた場合の結果を、計算をやり直さずに求めることができます。

Sumtally 機能が動作する条件は次の 3 つです。

- [parameters] セクションにおいて `icntl=13` とする
- 足しあわせたいタリー結果の条件（`mesh`，`axis`，`part` など）が一致していること
- 足しあわせたいタリー結果の条件が書かれたタリーセクションにおいて、sumtally サブセクションを設定する[83] [84]

Sumtally サブセクションが設定されていても、`icntl=13` となっていなければその内容は無視されます。（バージョン 2.88 以降では、全てのタリーで本機能を利用できます。）この機能の具体的な利用方法については、`\phits\utility\sumtally` フォルダにある資料やサンプルインプットをご参照ください。

Sumtally サブセクションは、“sumtally start” と “sumtally end” で挟んだ領域で設定します。本サブセクション内のユーザー定義定数 c*i* (i = 1 ~ 99) は無視されるのでご注意ください。

Sumtally サブセクションで使用できるパラメータは表 99 の通りです。

表 99: sumtally サブセクションパラメータ

name	値	説明
`isumtally =`	`1`(省略時), `2`	`Sumtally` 機能の計算方法 `1:` 手動並列計算用 `2:` 加重平均計算
`nfile =` `(次行)`	数 `file name` 数値	タリーファイル数 タリーファイル名、重み付けの値 `isumtally=2` の場合は自動的に規格化されます
`sfile =`	`file name`	`sumtally` 機能で足しあわせた結果の出力ファイル名
`sumfactor =`	（省略可、`D=1.0`）	規格化定数

`isumtally=1` の場合は手動で並列計算を実行することができます。例えば、`maxcas=100`，`maxbch=10` で計算した結果 result-1.out と `maxcas=100`，`maxbch=20` で計算した結果 result-2.out がある場合、次のように sumtally サブセクションを設定することで `maxcas=100`，`maxbch=30` の条件で計算した結果 result-s.out を得ることができます。

例 54: sumtally 機能を用いた例題（`isumtally=1`）の場合）

```
sumtally start
isumtally = 1        $(D=1) sumtally option, 1:integration, 2:weighted sum
nfile = 2            $ number of tally files
   result-1.out  1.0
```

[83] バージョン 2.81 以前では、sumtally サブセクションが設定されたタリーセクションにおいて `file` パラメータが設定されていなくても動作しましたが、バージョン 2.82 以降ではこのパラメータがないとエラーになりますのでご注意ください。

[84] バージョン 2.85 以前では、1 つのインプットファイルにおいて 1 つの sumtally サブセクションしか設定できませんのでご注意ください。

```
    result-2.out  1.0
sfile = result-s.out $ file name of output by sumtally option
sumfactor = 1.0      $ (D=1.0) normalization factor
sumtally end
```

ただし、乱数の重複を避けるために、result-2.out の計算を実行する際は [parameters] セクションにおいて irskip=-1000 を設定してください。また、足し合わせるファイル名の右で設定する重み付けの値は、基本的に 1 としてください。isumtally=1 の場合、重み付けの値を 1 より変更することで、各タリー結果を求めた際のソースウエイトの平均値を変えることになります（[source] セクションで wgt を変更した場合と同じ）。なお、isumtally=1 で得られたタリー結果に対しては再開始計算を実行することが可能です。この場合は nfile 以下で指定するタリーファイルの内、一番下のファイルにある乱数が用いられます。

isumtally=2 の場合、次の計算式により加重平均値 $\bar{X}$ を求めます。

$$\bar{X} = F \sum_{j=1}^{N} \frac{r_j}{r} \bar{X}_j \tag{23}$$

ここで、F が sumfactor で指定する規格化定数、N が nfile で指定するタリーファイル数、$\bar{X}_j$ が j 番目のタリーファイルの結果、r_j が j 番目のタリーファイルの重み付けの値です。r は $r = \sum_{j=1}^{N} r_j$ で計算しており、重み付けの値 r_j を変えることで、各タリー結果の相対的な強度を任意に与えることができます。標準偏差 σ_X は、j 番目のタリー結果の標準偏差を σ_{X_j} とし、

$$\sigma_X = F \sqrt{\sum_{j=1}^{N} \left(\frac{r_j}{r}\right)^2 \sigma_{X_j}^2} \tag{24}$$

により求めています。isumtally=2 は、例えば、ある標的に対して左右から違う量の放射線を照射した結果を求めたい時に利用できます。左から照射した結果が result-l.out、右から照射した結果が result-r.out であった場合、2:3 の割合で照射した結果は次のような設定で計算することができます。

例 55: sumtally 機能を用いた例題（isumtally=2）の場合）

```
sumtally start
isumtally = 2        $(D=1) sumtally option, 1:integration, 2:weighted sum
nfile = 2            $ number of tally files
   result-l.out  2.0
   result-r.out  3.0
sfile = result-s.out $ file name of output by sumtally option
sumfactor = 5.0      $ (D=1.0) normalization factor
sumtally end
```

この条件と同じシミュレーションはマルチソースの機能を利用することで実行可能です。しかし、照射割合を変えるなど、色々な条件の計算を調べたい場合は、isumtally=2 を利用するのが便利です。なお、重み付けの値は、足し合わせるタリー結果の相対的な割合にしか影響しませんのでご注意ください。絶対値を変える場合は sumfactor の値を変更します。また、isumtally=2 の場合は、乱数等の再開始計算に必要な情報は出力されず、再開始計算ができないようになっています。

6.9 複数のタリー結果の解析機能

前セクションに記載した sumtally 機能は複数のタリー結果の単純な加重和を計算する機能ですが、より複雑な解析を行いたい場合は anatally 機能を使う必要があります。例えば、粒子線治療における生物学的線量を吸収線量の絶対値を考慮して計算したり、物質密度がもつ誤差が計算結果に与える影響を系統的不確かさとして評価することが可能です。

Anatally 機能が動作する条件は次の 4 つです。

- [parameters] セクションにおいて `icntl=17` とする
- 解析したいタリー結果の条件（`mesh`，`axis`，`part` など）が一致していること
- 解析したいタリー結果のファイル名等をまとめた `[anatally]` セクションを設定する
- 上のタリー結果の出力したタリーセクションにおいて、`anatally start` と `anatally end` のみの anatally サブセクションを設定する

`[anatally]` セクションは、`icntl=17` の時のみ有効となります。その際、anatally サブセクションは `[anatally]` と `[t-track]` などのタリーセクションの 2 箇所に必要です。前者では解析したいファイル名等を記載し、後者では中身のない（`anatally start` と `anatally end` のみの）サブセクションを記載します。（なお、“autorun” スクリプトを利用する場合は、`icntl=16` とし、`[anatally]` は記載せず、解析したいファイル名等はタリーセクション中の anatally サブセクションにまとめてください。）

Anatally サブセクションは、“anatally start” と “anatally end” で挟んだ領域で設定します。本サブセクション内のユーザー定義定数 c*i* (*i* = 1 ~ 99) は無視されるのでご注意ください。

Anatally サブセクションで使用できるパラメータは表 100 の通りです。なお、現在のバージョンでは、`icntl=16,17` の場合に、3.3 節にある `itall=3` の場合に指定できる `ix,iy,iz,ipart` 等のパラメータは利用できません。

表 100: anatally セクションパラメータ

name	値	説明
manatally =	0, 1(省略時)	anatally 機能の計算方法 0: ユーザー定義 anatally(‘‘usranatal.f’’) に書かれたプログラムに従う（6.9.1 節参照） 1: 分散分析モード（6.9.2 節参照）
nfile = (次行)	数 file name	タリーファイル数 タリーファイル名
sfile =	file name	anatally 機能で解析した結果の出力ファイル名

例 56: anatally 機能を用いた例題（`manatally=0`）の場合）

```
1:  [t-track]
       .........
       .........
2:  anatally start
3:  anatally end
       .........
       .........

4:  [anatally]
```

```
anatally start     1
manatally = 0        $(D=1) anatally option, 0:user-defined anatally, 1:anova
nfile = 2            $ number of tally files
   result-1.out
   result-2.out
sfile = result-a.out $ file name of output by anatally option
anatally end     1
```

この例では、[t-track] で得られた result-1.out と result-2.out の結果を使用し、ユーザー定義タリーによって計算した結果を result-a.out に出力します。[anatally] セクション中の anatally start と anatally end の後にある数字は anatally サブセクションの ID 番号です。解析したいタリー結果の条件（mesh，axis，part など）が書かれたタリーセクションが、そのインプットファイルにおいて、何番目のタリーセクションなのかを ID 番号として書いてください。上の例で、もし [t-track] より上に 1 つ別のタリーセクションが書かれていた場合は、[anatally] セクションにおいて、anatally start と anatally end の後には 2 と記載してください。

6.9.1 ユーザー定義 anatally 機能

ユーザー定義 anatally(“usranatal.f”) に書かれたプログラムに従って計算した値を出力します。既定プログラムとして、ユーザー定義定数に従って各タリー値を重み付けして加算するモードと、確率論的マイクロドジメトリ (SMK) モデル[85]に従って BNCT における光子等効果線量を計算するモードが組み込まれています。その具体的な使用方法は、“/phits/utility/usranatal” をご参照ください。また、PHITS の再コンパイルが必要となりますが、同フォルダには、粒子線治療における生物学的線量を SMK モデルに従って計算するサンプルプログラムも含まれていますので、そちらもご活用下さい。

6.9.2 分散分析機能

スクリプトを利用して、計算条件を変えながら実行した輸送計算の結果を分析することができます。例えば、物質の密度に誤差が含まれる場合、その誤差の範囲で密度を変化させながら輸送計算を実行することで、密度の誤差が計算結果に与える影響を調べることが可能です。PHITS では、専用の分析機能スクリプト “autorun.bat” （Windows) や “autorun.sh” （Mac, Linux）を利用することで、計算条件の変化がタリー結果に与える影響を自動的に分析することができます。
（注）ただし、現在のバージョンでは次の制限があります。利用可能なのは [t-track] と [t-deposit]（output=dose）のみです。変えられる計算条件は 1 つだけです。また、anatally サブセクションを指定しますが、3.3 節にある itall=3 の場合に指定できる ix,iy,iz,ipart 等のパラメータは利用できません。
分散分析機能は以下のような流れで利用します。

(1) PHITS の輸送計算の入力情報をまとめたインプットファイル（ここでは “phits.inp” とする）を用意する。

(2) 分析機能スクリプト用のインプットファイル（ここでは “autorun.inp” とする）を用意する。

(3) autorun.inp を入力ファイルとして “/phits/utility/autorun” フォルダにあるスクリプト autorun.bat or autorun.sh を実行する。

各インプットファイルの内容とスクリプトの動作の詳細は次節以降でご説明します。

[85] 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 輸送計算のインプットファイル (phits.inp)

輸送計算の情報をまとめた phits.inp（ファイル名変更可能）では、変化させたい数値をユーザー定義定数 `ci`($i = 1, \cdots, 99$) により指定してください。また、`[parameters]` セクションにおいて `icntl=16` としてください。そして、分析機能を利用したいタリーセクションに、次のような anatally サブセクションを設定してください。

例 57: 分散分析機能を利用する場合のタリーセクションにおける anatally サブセクション

```
1:  anatally start
2:  manatally = 1
3:  sfile = track_a.out
4:  anatally end
```

Anatally サブセクションは、`anatally start` と `anatally end` で挟んだ領域で設定します。`manatally` は分析内容を決めるオプションです。`manatally=1` の場合、計算条件 `ci` の変化がタリー結果に与える影響を分散分析 (ANOVA; analysis of variance) に基づき系統的不確かさとして評価します。`sfile` は分析結果を出力するファイル名です。

6.9.2.2 分析機能スクリプト用インプットファイル (autorun.inp)

専用のスクリプト (autorun.bat, autorun.sh) を実行する際に使用するインプットファイル autorun.inp（ファイル名変更可能）は次の形式で作成してください。

例 58: 分析機能を利用する場合のスクリプト用インプットファイル

```
1:  file=phits.inp
2:  set:c1
3:  c-type=2
4:  cmin=10
5:  cmax=100
6:  nc=10
```

`file=`の後に、輸送計算の情報をまとめた PHITS のインプットファイル名を書いてください。`set:`の後にはそのインプットファイル内で変化させたい計算条件 `ci`($i = 1, \cdots, 99$) の情報を記載してください。`set:c1` の次の行から、`e-type` 等のタリーのメッシュタイプと同じ形式で `ci` のデータを指定します。`c-type=1,2,3` が指定できます。上の例では、`c1`=10 より 100 までを 10 きざみで変化させ、各条件の輸送計算を実行します。

6.9.2.3 スクリプトの動作

スクリプト autorun.bat, autorun.sh は次の流れで各計算を実行します。

(1) autorun.inp から `ci` に関する情報を読み取り、“nc_info.inp” に書き出す。

(2) `icntl=16` の条件で PHITS を実行し、phits.inp と nc_info.inp の内容に基づき、輸送計算実行用の一時ファイル “phits_tmp.inp” と分析用ファイル “anatally.inp” および計算条件ファイル “varfile_j.inp” ($j = 1, \cdots, nc$) を作成する。

(3) 計算条件を変えて求めた輸送計算の結果をまとめる “outfiles” フォルダを作成する。

(4) phits_tmp.inp と varfile_j.inp をインプットファイルとして、合計 `nc` 回 PHITS を実行する。その際、j 番目 ($j = 1, \cdots, nc$) の PHITS のタリー結果は、outfiles フォルダ中に “/j/” フォルダを作成し、各々移動する。

(5) anatally.inp をインプットファイルとして、`icntl=17` の条件で PHITS を実行する。その分析結果を `sfile` で指定したファイルに書き出す。

6.9.2.4 出力形式

`sfile` で指定したファイルには、通常のタリー結果の平均値と統計誤差 $\sigma_{\rm stat}$ に加え、系統的不確かさ $u_{\rm syst}$ と全不確かさ $u_{\rm tot}$ が出力されます。これらは次の関係を満たします。

$$u_{\rm tot}^2 = u_{\rm syst}^2 + \sigma_{\rm stat}^2 \tag{25}$$

ここで、統計誤差 $\sigma_{\rm stat}$ は統計的不確かさ $u_{\rm stat}$ と標本数 N を用いて、$\sigma_{\rm stat}^2 = u_{\rm stat}^2/N$ により計算されます。詳細は文献[86] を参照してください。

出力は、例えば `axis=x, part=all` の場合は次のようになります。

```
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
            .........
```

`all` の列にタリー結果の平均値を出力し、その右側に全不確かさ、系統的不確かさ、統計誤差の順に相対値で出力します。なお、統計量が少ない場合は系統的不確かさの推定値が負となり、その場合は 0 と出力します。

[86] 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).

7 タリー入力書式

7.1 [T-Track] セクション

指定した任意の空間における粒子の fluence を出力します。このタリーでは、指定した空間を粒子が通過した際に、その空間における飛跡長 (track length) を計算しており (図 57 参照)、飛跡長の和を空間の体積で割ることによって、単位面積あたりの粒子の流量が得られるようになっています。

例えば、このタリーは指定した空間に置いた測定機器の応答の状況を調べるのに利用できます。各測定機器がもつ応答性能を ([cm^2] の単位をもつ) 断面積の形で計算しておけば、その値を本タリーで求めた fluence と掛け合わせることにより、シミュレートした状況においてその測定機器が何回応答するかを評価できます。

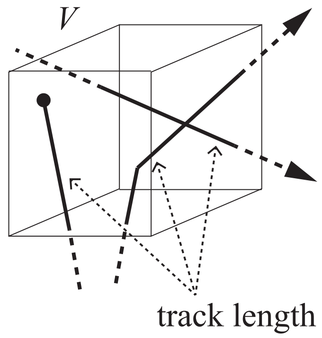


図 57: Track タリー: 空間内の飛跡 (実線) の長さを計算する。

表 101: [t-track] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz, tet	形状メッシュ、形状メッシュサブセクションが必要
part =	all(省略時), 粒子名	ひとつの [t-track] セクション最大 6 個まで
material =	(省略可) all, 数	スコアする material を限定する。複数定義可 all : デフォルト、この場合省略した場合と同じ 数を指定した場合、その数だけの material を次の行に記述する。 負の数にした場合は、それらの material を対象から外すことを意味する。
(次行)	2 5 8	material 番号
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ エネルギーメッシュサブセクションが必要
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ 時間メッシュサブセクションが必要
a-type =	1, 2, -1, -2 (省略可)	mesh=r-z とした場合の θ に関する角度メッシュ 角度メッシュサブセクションが必要

表 102: [t-track] パラメータ (2)

name	値	説明
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/nsec/MeV/source] 13:[1/cm^2/Lethargy/nsec/source] 14:[cm/nsec/source]
axis =	eng, reg, x, y, z, r, tet, rad, deg, xy, yz, xz, rz, t	出力データの x 軸 出力データの x 軸（mesh=tet でのみ有効） 出力データの x 軸（mesh=r-z で、θ に関する自由度） 2 次元表示 時間軸
file =	file name	axis の数だけ定義する
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の場合でも 1 つのみ指定。
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト

unit = 3, 13 の場合の Lethargy はエネルギーに関する自然対数目盛を表しています。各エネルギービンの上限と下限がそれぞれ $E_{\rm high}, E_{\rm low}$ のときに、各 Lethargy の幅を $\ln(E_{\rm high}/E_{\rm low})$ で与えます。

unit = 1, 2, 3, 11, 12, 13 を指定すると、fluence、すなわち入射粒子あたりの飛跡長の和をその空間の体積で割った値を出力します。ただし、reg メッシュを用いている場合は、その体積を [Volume] セクションで与える必要があります。これが与えられていないときは、体積が 1cm^3 であるとして、入射粒子あたりの飛跡長の和をそのまま出力します。r-z, xyz メッシュの場合は体積が自動的に計算できますので、その値が用いられます。unit=4,14 を指定すると、体積で割らずにそのまま飛跡長の和を出力します。

本タリーに関しては、mesh=r-z とした場合に、既存の r, z メッシュに加え、θ メッシュを指定することができます。θ に関するメッシュを指定する際は、a-type を用いてください。また、角度を横軸にして出力する場合は、axis として rad か deg を指定してください。rad は radian で、deg は degree の単位で出力します。

表 103: [t-track] パラメータ (3)

name	値	説明
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は, icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時), 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示 この下に xyz 形状メッシュセクションが必要
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域 を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める 分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する 線の太さを定義します。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された 体積の値が使用されます。 また、reg=において (　) 等により特殊な領域指定を行った場合は、 volume を省略すると、インプットエコーに PHITS 内部で与えられた領 域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、 各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ 1 辺のスキャン数
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイル を作成。ファイル名は出力ファイルの拡張子を eps に 変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、 axis= xy, yz, xz, rz の場合は無効。
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。 ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkout =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルを出力する。 ファイル名は出力ファイルの拡張子を vtk に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkfmt =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルのフォーマット。 テキスト形式（0）またはバイナリ形式（1）を選択する。

表 104: [t-track] パラメータ (4)

name	値	説明
foamout =	0(省略時), 1, 2	1 でタリー出力結果（数値データ）を OpenFOAM フィールドデータ形式で出力する。ファイル名は出力ファイルの拡張子を foam に変えたファイル名。 2 でタリー出力結果（四面体要素番号、重心 *x* 座標、*y* 座標、*z* 座標、体積、数値データ、相対誤差）を CSV 形式（コンマ区切り）で出力する。ファイル名は出力ファイルの拡張子を csv に変えたファイル名。 mesh=tet, axis=tet のときのみ有効。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値 i-th カウンターの最大値
chmin(i) = chmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	ヒストリー内で発生した全粒子の i-th カウンターの最小値。バッチ分散モード (istdev=1) では設定不可。 ヒストリー内で発生した全粒子の i-th カウンターの最大値。バッチ分散モード (istdev=1) では設定不可。
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
gslat =	2（省略時）, = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。（ただしボクセルファントムはきれいに描画されません。） 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。
multiplier =	物質数 (省略可)	multiplier を物質毎に指定します。 詳細は 5.24.1 Multiplier サブセクションをご覧ください。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

Anatally サブセクションを設定することで、タリー結果と統計誤差の推移を表示する機能を有効にした際に、出力させるメッシュを指定することができます。Anatally サブセクションは、anatally start と anatally end で挟んだ領域で設定します。詳細は、3.3 節をご覧ください。

7.2 [T-Cross] セクション

指定した任意の面における粒子の current、または flux を出力します。粒子が面を通過する度にそのままカウントするものが current で、面の法線ベクトルに対して粒子が角度 θ で入射した場合に $1/\cos\theta$ の重みを付けてカウントするものが flux（ただし、単位時間あたりの量ではないので、正確には fluence）です。これらは共に面を通過する単位面積あたりの粒子の数を意味する物理量ですが、current が単に指定した面を通過する粒子の総量を評価するのに対し、flux は粒子の流れの方向に垂直な面を通過する量を評価します。したがって、単位面積あたりの値を求める際に必要となる面積が違うこととなり (図 58 参照)、flux の場合は $1/\cos\theta$ の重みを付けてカウントすることになります。なお、図 58 の S は、`mesh=reg` を選択している場合は形状メッシュサブセクションにおいて `area` として与えます。`mesh=r-z`, `xyz` を選択している場合は自動的に計算された値が使用されるため必要ありません。

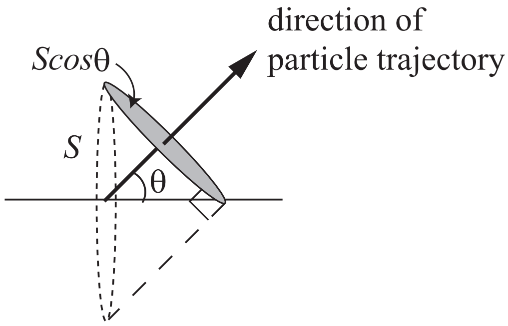


図 58: 指定した面の面積 S と粒子の運動の方向に対して有効な面素 $S\cos\theta$ の関係。

本タリーの flux は $1/\cos\theta$ の重みを考慮した量となるため、`[t-track]` で指定する領域の厚さを無限に薄くした場合と同等の結果が得られます。これにより例えば、指定した面に沿って設置した測定機器の応答の状況を調べるのに利用でき、前もって ([cm^2] の単位をもつ) 断面積の形で計算しておいた各測定機器の応答性能と掛け合わせることにより、その測定機器が応答する回数を見積もることができます。

表 105: [t-cross] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz	形状メッシュ、形状メッシュサブセクションが必要
part =	all(省略時), 粒子名	ひとつの [t-cross] セクション最大 6 個まで
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ エネルギーメッシュサブセクションが必要
a-type =	1, 2, -1, -2 a-curr, oa-curr の時に必要	角度メッシュ (1, 2 :cos, -1, -2 :degree) 角度メッシュサブセクションが必要
iangform =	0（省略時）,1,2,3	a-type mesh の角度基準を選択する。0 で面の法線ベクトル、1 で x 軸、2 で y 軸、3 で z 軸。
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ 時間メッシュサブセクションが必要
unit =	1, 2, 3, 4, 5, 6 11, 12, 13, 14, 15, 16	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: [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, reg, x, y, z, r, cos, the, t xy yz, xz, rz	出力データの x 軸 角度 (cos, the), 時間 (t) 2 次元表示 2 次元表示, enclos=1 の場合のみ有効。

r-z メッシュの時にメッシュの側面が円柱状のあるセルの側面と一致する場合や、xyz メッシュの時に座標変換によって存在するセルの境界面とタリー面が一致する場合は、計算結果がおかしくなる可能性があるので注意してください。

角度メッシュを用いることで、面の法線ベクトルとのなす角 θ 毎の current もカウントできます。図 59 はこの測定をイメージしたものです。unit=4, 5, 6, 14, 15, 16 を指定すると、角度メッシュサブセクションで与えられる角度のビンの幅から対応する立体角を計算し、単位立体角あたりの量を出力します。

unit = 3, 6, 13, 16 の場合の Lethargy はエネルギーに関する自然対数目盛を表しています。各エネルギービンの上限と下限がそれぞれ $E_{\rm high}, E_{\rm low}$ のときに、各 Lethargy の幅を $\ln(E_{\rm high}/E_{\rm low})$ で与えます。

unit = 4, 5, 6, 14, 15, 16 の場合の sr は、立体角の単位ステラジアン (steradian) を表しています。

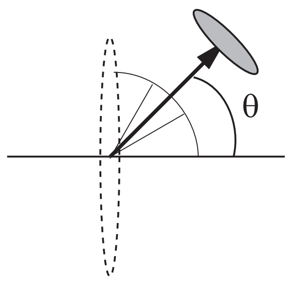


図 59: 角度メッシュを用いた current の角度分布の評価。

表 106: [t-cross] パラメータ (2)

name	値	説明
file =	file name	axis の数だけ定義する
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の 場合でも 1 つのみ指定。
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
2d-type =	1,2,3,4,5,6,7 (省略可、D=2)	2 次元表示のオプション
output =	flux current f-curr b-curr o-curr of-curr ob-curr a-curr oa-curr	面横断の flux を出力 面横断の current を出力 面横断の forward current を出力 面横断の backward current を出力 面横断の omni current を出力 omni は、energy integrated を意味します 面横断の omni forward current を出力 面横断の omni backward current を出力 面横断の角度メッシュ current を出力 面横断の角度メッシュ omni current を出力
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は，icntl=8 の場合にピクセル形式で物質色を表示します。
ginfo =	0(省略時), 1 2	gshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域 を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow オプションの時、領域境界を求める 分解能を各辺 resol 倍します。 gshow オプションの時、領域境界を表示する 線の太さを定義します。
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイル を作成。ファイル名は出力ファイルの拡張子を eps に 変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、axis= xy の 場合は無効。

output=f-curr, b-curr, of-curr, ob-curr は、xyz メッシュと r-z メッシュで利用できます。ただし、xyz メッシュの場合は、z 軸の方向に対してのみタリーが機能します。

表 107: [t-cross] パラメータ (3)

name	値	説明
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値。 i-th カウンターの最大値。
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義。
dump = (次行)	データ数 データ定義	ファイルに粒子の情報をダンプします。mesh=reg のみ。 正で与えた時はバイナリー、負の時はアスキーファイル。 データの順番を定義します。ただし、idmpmode=1 の使用のためには、nocas および nobch の情報が必要です。
gslat =	2（省略時）, = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。(ただしボクセルファントムはきれいに描画されません。) 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。
multiplier =	物質数 (省略可)	multiplier を物質毎に指定します。[87] 詳細は 5.24.1 Multiplier サブセクションをご覧ください。
enclos =	0(省略時), 1	mesh=r-z のときは z 横断面と r 横断面の 2 種類の横断面を定義。mesh=xyz のときは z 横断面を定義。 メッシュで区切られたそれぞれの閉曲面を横断面として定義。

[t-cross] で、dump を用いることができるのは、reg メッシュのみです。 また、axis は reg に固定されます。dump を指定したときは、e-type, a-type, t-type の指定は、上限と下限の意味しかもちません。 unit は 1 に固定、output は current もしくは、a-curr, oa-curr になります。 dump 定義文を用いてダンプさせる時は、axis、file 数ともにひとつに制限されます。dump するデータを書き込むファイルは、file 定義文で指定した名前に "_dmp" を付けたファイルです。file 定義文で定義された file 名には、通常のタリーの結果が書き出されます。(バージョン 2.66 以前では、file=で指定したファイルに dump するデータが書き込まれ、通常のタリー結果を出力するファイルとして.cfg をつけたファイルが作成されていました。) dump ファイルを使用した接続計算を idmpmode=1 で実行するには、nocas および nobch の情報を出力しておく必要があります。また、idmpmode=1 を使用するためには、ここで出力される "_dmp" が付いた dump ファイルと共に file= で指定した名前の通常のタリー出力ファイルが必要になります。メモリ共有型並列計算を実行する場合、本機能による dump ファイルの作成と dumpall によるイベント情報の書き出しは共存できませんのでご注意ください。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

バージョン 3.10 より、mesh=xyz または r-z とした場合に enclos=1 とすることで、メッシュで区切られた閉曲面を横断面として設定する機能を追加しました。このときの forward は閉曲面に流入する方向、backward は閉曲面から流出する方向を意味します。また、単位面積当たりの量の計算では、その閉曲面の総面積を使用します。

[87] バージョン 2.86 より、[t-cross] でも multiplier の機能が使えるようになりました。

形状メッシュとして mesh=reg を選択した場合、それに続く形状メッシュサブセクションにおいて、タリーするための横断面を定義する必要があります。横断面の定義は、出て行く領域 (r-from) と入ってくる領域 (r-to) の領域番号を指定する形で行います[88]。また、単位面積あたりの量を計算するために、その横断面の面積を area （単位は cm^2）として与えてください。

mesh=reg の場合の形状メッシュサブセクションは、次のような書式となります。

```
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
   ...     ...     ....
   ...     ...     ....
```

mesh=reg と書いた次の行で、reg=により、指定したい横断面の数を決めてください。更にその次の行から、行列形式で r-from, r-to, area を指定します。これらは reg=で決めた数の行数分のデータセットが必要です。データ列の順番は、r-from r-to area がデフォルトです。この見出しの行は省略できますが、もし順番を変える場合は、area r-from r-to のように各列の見出しを明示してください。また、読み飛ばしコラム用の non も使えます。領域番号を指定する際は、領域をまとめる ({ 2 - 5 } 8 9) という書式や (6 < 10[1 0 0] < u=3) などの lattice, universe 構造も利用できます。ただし、単一の数字で無い場合は必ず (   ) で括ってください。これらの順番を変えたい場合は、見出しの順番を変えます。

mesh=reg とした際に、粒子 fluence を求める場合は注意が必要です。上の例題において、上から 3 番目に示した指定方法以外は、片方向の fluence になります。したがって、粒子 fluence をタリーする場合は、両方向の寄与を足し合わせるために、r-from と r-to の両方に横断面を挟む 2 つの領域番号を書いてください。

形状メッシュで、mesh=r-z を選択した場合は、$r_i - r_{i+1}$ で定義される領域の nz+1 個の z 横断面と、$z_i - z_{i+1}$ で定義される領域の nr+1 個の r 横断面の 2 種類の横断面が定義されます。r-surface と外部ボイドの面が同じ場合は、この面での cross は、タリーされませんからご注意ください。

形状メッシュで、mesh=xyz を選択した場合は、$x_i - x_{i+1}, y_j - y_{j+1}$ で定義される領域の nz+1 個の z 横断面が定義されます。このとき、x, y 横断面は定義されません。

mesh = r-z，xyz の時は、面を横断する粒子の両方向が自動的に入ります。 また、これらの時の forward は、z 面では、正方向、r 面では中心から外方向を意味します。バージョン 3.05 より、z-type=1 とした場合に nz=0 とすることで、タリー面を 1 面だけ設定できるようにしました。

[88] バージョン 2.96 以前は、r-from と r-to ではなく、それぞれ r-in と r-out を使用していました。r-in と r-out は、バージョン 2.97 以降でも利用できますが、出て行く領域 (r-in) と入ってくる領域 (r-out) の in と out が逆になっていますのでご注意ください。

7.3 [T-Point] セクション

[t-point] タリーは、ある点 (point detector) やリング状の線分 (ring detector) におけるフルエンスを計算することができます。PHITS は、通常、設定した領域に確率的に入ってくる放射線をタリーするため、タリー領域が小さくなると、十分な統計精度を得るためには極めて長い計算時間を要する場合があります。極端な例として、タリー領域を点や線分に設定した場合、どれだけ計算時間を費やしても結果を得ることはできません。[t-point] タリーは、[t-track] や [t-cross] タリーの手法とは異なり、線源位置、もしくは、散乱によって粒子が発生した地点で、タリー位置への方向確率と透過確率を計算して評価することができるため、粒子の存在確率が小さい領域での測定に有効に機能します。 ただし、散乱点での生成粒子の角度分布、エネルギー分布があらかじめ分かっていないと評価できません。また、物質中での荷電粒子のエネルギー損失があると透過確率の評価が難しいので、対象粒子は、核データの存在する中性子と光子に限られます。したがって、[t-point] タリーの利用条件は、

(1) 輸送計算の上限エネルギーは、ライブラリを使う上限エネルギー (dmax) とする

(2) 検出可能な粒子は中性子・光子のみとする

(3) Event Generator mode 及び EGS5 mode を使用しない

(4) 面定義で全反射や白色反射の面を利用しない

となります。本タリーの計算原理や具体的な利用方法については、\phits\utility\tpoint フォルダにある readme やサンプルインプットをご参照ください。

表 108: [t-point] パラメータ (1)

name	値	説明
point =	数	指定する point detector の数。サブセクションが必要
ring =	数	指定する ring detector の数。サブセクションが必要
part =	all(省略時), 粒子名	ひとつの [t-point] セクションに最大 6 個まで
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ エネルギーメッシュサブセクションが必要
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ 時間メッシュサブセクションが必要
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]

unit = 3, 13 の場合の Lethargy はエネルギーに関する自然対数目盛を表しています。各エネルギービンの上限と下限がそれぞれ $E_{\mathrm{high}}, E_{\mathrm{low}}$ のときに、各 Lethargy の幅を $\ln(E_{\mathrm{high}}/E_{\mathrm{low}})$ で与えます。

通常のタリーで指定する `mesh` パラメータの代わりに、`[t-point]` タリーでは、point もしくは ring パラメータを指定します。まず、point detector の場合、入力書式は次のようになります。

```
[ T-point ]
point =    1     # number of point detectors
non     x        y       z        r0
  1     10.0     0.0     50.0     1.0
```

`point=n` で point detector の個数 n を指定します。ひとつの `[t-point]` タリーセクションで定義できるポイントの上限数は 20 です。これ以上のポイントを定義するときは、複数のセクションを用意してください。次の行には、ポイントを指定する座標 (x,y,z)、ポイント周辺の特異領域の半径 (r0) のデータ順を定義します。読み飛ばしのコラムの指標 (non) も使えます。これらのデータの並びを変えるときは、この行で指定してください。この定義文の下に `point=`で指定した数だけデータを定義します。座標は、x,y,z 座標 [cm] で指定し、特異領域の半径 (r0) も [cm] で指定してください。ここで、特異領域の半径については、`\phits\utility\tpoint` フォルダにある readme をご参照ください。

次に、ring detector の場合は、次のような入力書式になります。

```
[ T-point ]
ring =    1     # number of ring detectors
non   axis    ar       rr       r0
  1    z      50.0     10.0     1.0
```

`ring=n` で ring detector の個数 n を指定します。次の行には、リングの軸を x,y,z の文字で指定する axis、原点からリングの位置までの距離 (ar)、リングの半径 (rr)、ポイント周辺の特異領域の半径 (r0) のデータ順を定義します。読み飛ばしのコラムの指標 (non) も使えます。これらのデータの並びを変えるときは、この行で指定してください。この定義文の下に `ring=`で指定した数だけデータを定義します。リング軸は (x,y,z) の文字で指定し、リングまでの距離 (ar) とリングの半径 (rr) は [cm] で指定します。特異領域の半径 (r0) も [cm] で指定してください。ここで、特異領域の半径については、`\phits\utility\tpoint` フォルダにある readme をご参照ください。

表 109: [t-point] パラメータ (2)

name	値	説明
axis =	eng, t	出力データの x 軸
file =	file name	axis の数だけ定義する
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の場合でも 1 つのみ指定。
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。ファイル名は出力ファイルの拡張子を eps に変えたファイル名。 2 とすることで更に誤差棒を表示する。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値 i-th カウンターの最大値
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。
multiplier =	物質数 (省略可)	multiplier を物質毎に指定します 詳細は 5.24.1 Multiplier サブセクションをご覧ください。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

Anatally サブセクションを設定することで、タリー結果と統計誤差の推移を表示する機能を有効にした際に、出力させるメッシュを指定することができます。Anatally サブセクションは、anatally start と anatally end で挟んだ領域で設定します。詳細は、3.3 節をご覧ください。

7.4 [T-Adjoint] セクション

光子輸送の adjoint mode 計算 (iadjoint=1) で結果を出力するタリーです。このタリーの結果は通常の輸送計算における [T-Track] タリーの結果に相当します。Adjoint mode 計算の詳細や [T-Adjoint] の e-type(エネルギーメッシュ)、e-smin、e-smax、factor パラメータ定義については5.2.9節と \phits\utility\adjoint\ フォルダにあるサンプルインプットをご参照ください。

表 110: [t-adjoint] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz	形状メッシュ、形状メッシュサブセクションが必要
material =	(省略可) all, 数	スコアする material を限定する。複数定義可 all : デフォルト、この場合省略した場合と同じ数を指定した場合、その数だけの material を次の行に記述する。負の数にした場合は、それらの material を対象から外すことを意味する。
(next line)	2 5 8	material 番号
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ エネルギーメッシュサブセクションが必要
m-source =	(D=1)	タリーする adjoint マルチソースの数
e-smin =	(D=0.58)	実際の光子線源の最大エネルギー
e-smax =	(D=0.62)	実際の光子線源の最小エネルギー
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ 時間メッシュサブセクションが必要
unit =	1 (のみ)	[1/cm^2/source]
axis =	eng, reg, x, y, z, r xy, yz, xz, rz t	出力データの x 軸 2 次元表示 時間軸
file =	file name	axis の数だけ定義する
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の場合でも 1 つのみ指定。
factor =	(省略可, D=1.0)	normalization factor。実際の光子線源放射量を定義
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
2d-type =	1, 2, 3, 4, 5, 6, 7 (省略可, D=3)	2 次元表示のオプション
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト

表 111: [t-adjoint] パラメータ (2)

name	値	説明
gshow =	0 (省略時), 1，2，3，4，5	mesh=xyz, axis=xy，yz，xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、 LAT 番号 (4) を表示 gshow=5 は、icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0 (省略時), 1，2，3	mesh=xyz, axis=xy，yz，xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示 この下に xyz 形状メッシュセクションが必要
ginfo =	0 (省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1 (省略時), 0.5 (省略時)	gshow, rshow オプションの時、領域境界を求める 分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する 線の太さを定義します。
volume = reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された体積の値が使用されます。 また、reg=において ( ) 等により特殊な領域指定を行った場合は、volume を省略すると、インプットエコーに PHITS 内部で与えられた領域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照
iechrl	72 (省略時)	volume のインプットエコーの時の最大コラム数
volmat	(省略可, D=9)	xyz メッシュで material が指定されている時、 各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ 1 辺のスキャン数
epsout =	0 (省略時)，1，2	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。ファイル名は出力ファイルの拡張子を eps に変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、axis=xy，yz，xz，rz の場合は無効。
ctmin(i) = ctmax(i) =	(省略可, D=-9999) (省略可, D=9999)	*i*-th カウンターの最小値 *i*-th カウンターの最大値
trcl =	(省略可)	r-z，xyz メッシュの座標変換番号もしくは座標変換定義
gslat =	1 (省略時) 0	1 で gshow で lattice の境界を表示 0 で gshow で lattice の境界を非表示

表 112: [t-adjoint] パラメータ (3)

name	値	説明
stdcut =	(省略可, D=-1)	統計誤差による計算打ち切り機能のしきい値。
multiplier =	物質数 (省略可)	multiplier を物質毎に指定します。 詳細は 5.24.1Multiplier サブセクションをご覧ください。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

Anatally サブセクションを設定することで、タリー結果と統計誤差の推移を表示する機能を有効にした際に、出力させるメッシュを指定することができます。Anatally サブセクションは、anatally start と anatally end で挟んだ領域で設定します。詳細は、3.3 節をご覧ください。

7.5 [T-Deposit] セクション

放射線による付与エネルギー（発熱）を出力するタリーです。バージョン 3.05 以降、カーマ近似を用いた中性子や光子による付与エネルギーも計算できるようになりました。カーマ近似の使用の有無は、イベントジェネレータモードや電子輸送計算の設定に従って自動的に調整されます（詳しくは、`[parameters]` セクションの 5.2.8 節にある `ikerman`, `ikermap` を参照）。ただし、中性子のカーマ近似が有効となるのは、核データを利用する `dmax(2)`（通常は 20 MeV）以下のみです。

荷電粒子による付与エネルギーを計算する場合は、ユーザーが usrdfn1.f や usrdfn2.f を変更して自由に定義できるファクターを乗じることができます。ただし、この機能はカーマ近似や飛跡構造解析モードには適応していませんのでご注意ください。サンプルプログラムとして、usrdfn1.f には粒子の LET に ICRP60 で定義された $Q(L)$ 関係を乗じて線量当量を導出するプログラムが、usrdfn2.f には、microdosimetric kinetic model に基づいて生物学的線量を導出するプログラムが入っています。なお、これら既定プログラムを使う場合は、`letmat` で水を指定する必要があります（詳しくは奨励設定の particletherapy を参照）。

本タリーにおいて `output=deposit` とした場合は、ヒストリー毎の付与エネルギー分布を計算するため、検出器の応答を模擬することができます。その際、ファノ因子などを指定することにより検出器の分解能も考慮可能です。なお、`output=deposit` の場合は、カーマ近似による付与エネルギーが考慮されませんので、必ずイベントジェネレータモードや電子輸送計算を行うように設定してください。また、`[weight window]` などの使用により、1 ヒストリー内でエネルギー付与する粒子のウエイトが変わる場合は適切な結果が得られませんのでご注意ください。

表 113: `[t-deposit]` パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz, tet	形状メッシュ、形状メッシュサブセクションが必要。
part =	all(省略時), 粒子名	ひとつの [t-deposit] セクション最大 6 個まで。 ただし、`output=deposit,deposit=0` の場合、all が必須であるため、最大 5 個までしか指定できません。 `neutron` や `photon` を指定した場合は、中性子や光子のカーマ近似による付与エネルギーがそれぞれ出力されます。
material =	(省略可) all, 数	スコアする material を限定する。複数定義可。 all : デフォルト。この場合省略した場合と同じ。 数を指定した場合、その数だけの material を次の行に記述する。負の数にした場合は、それらの material を対象から外すことを意味する。
(次行)	2 5 8	material 番号
letmat =	(省略可)	LET(dE/dx) を評価する物質番号、省略時は、実際の物質。 物質を指定する場合の密度は、[Material] セクションで定義された密度となる。したがって、水に対する LET 分布を計算したい場合、[Material] セクションで、水の密度が 1g/cm^3 となるように定義する必要がある。 また、負の場合は電子・陽電子の dE/dx として 1g/cm^3 の水に対する値が参照される。詳しくは、奨励設定の particletherapy を参照してください。

`[t-deposit]` では、タリーする領域に入る粒子ごとの付与エネルギーを `part=`で指定して計算することはできません。領域に入る粒子毎の付与エネルギーをタリーするには、`[counter]` で `part=`を用いて、特定の入射粒子から引き起こされる付与エネルギーをカウンターで指定する必要があります。

表 114: [t-deposit] パラメータ (2)

name	値	説明
dedxfnc =	(省略可, D=0)	0:ファクター無し、1:usrdfn1 を使用、2:usrdfn2 を使用 サンプルプログラムとして、usrdfn1.f には粒子の LET に ICRP60 で定義された $Q(L)$ 関係を乗じて線量当量を導出するプログラムが、usrdfn2.f には microdosimetric kinetic model に基づいて生物学的線量を導出するプログラムが入っています。これらの関数は、カーマ近似や飛跡構造解析モードによる付与エネルギーは考慮されませんのでご注意ください。
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ。output=deposit に必要。 エネルギーメッシュサブセクションが必要。
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ。時間メッシュサブセクションが必要。
output =	dose deposit	荷電粒子のエネルギー付与をタリーする。 イベント毎のエネルギー付与分布をタリー。e-type セクションが必要。カーマ近似によるエネルギー付与は考慮されないので、必ずイベントジェネレータモードや電子輸送計算を行うように設定してください。
unit =	0, 1, 2, 3, 4, 5	0: Dose [Gy/source]; output=dose のみ 1: Dose [MeV/cm^3/source] 2: Dose [MeV/source] 3: Number [1/source]; output=deposit のみ 4: Number [1/nsec/source]; output=deposit のみ 5: Dose [J/m^2/source]; output=dose のみ
deposit =	0(省略時), 1	output=deposit 選択時のエネルギー表示オプション 0: total deposit および各粒子の寄与の分布（通常はこちらをご使用ください） 1: 各粒子の deposit エネルギー分布（ver. 2.81 以前の計算方法です。）

`output=dose` で `unit=0` とした場合、[Gy/source] を単位とする吸収線量が出力されます。`mesh=reg` の場合は、`volume` パラメーターを用いるか、`[volume]` セクションにおいて各領域の体積を設定する必要があります。ただし、吸収線量は不可算量ですので、各 x 軸に関する積分値 (sum over) は出力されません。また、領域内に複数の物質が混在する場合、その領域の吸収線量は全体の平均値とならないことにご注意ください。例えば、領域の 1 と 2 にある質量 M_1 と M_2 の物質にそれぞれ E_1 と E_2 の熱量が吸収された場合、その平均吸収線量は $\frac{E_1+E_2}{M_1+M_2}$ ですが、PHITS で計算する値は $\frac{E_1}{M_1}\frac{V_1}{V_1+V_2}+\frac{E_2}{M_2}\frac{V_1}{V_1+V_2}$ となります。ここで、V_1 と V_2 はそれぞれ領域 1 と 2 が占める体積を表します。

`output=deposit` で `deposit=0` とした場合、`part` パラメータで指定した粒子種の total deposit (`part=all` の結果) に対する寄与を出力します。その際に、`part` パラメータで `all` が指定されていない時は、自動的に `all` が追加されます。一方、`output=deposit` で `deposit=1` とした場合、粒子種毎の deposit エネルギー分布が個別に出力されるため、各粒子種について得られた結果を足し上げた値と `all` を指定した結果は一致しませんのでご注意下さい。

`output=deposit` で `deposit=0` とした場合、`part=all` 以外のタリー結果に対する統計誤差は、`istdev` の値に関わらず、ヒストリー分散モードにより計算した標準偏差となります。

表 115: `[t-deposit]` パラメータ (3)

name	値	説明
axis =	eng, reg, x, y, z,r, t tet, xy, yz, xz, rz, t-eng, eng-t	出力データの *x* 軸 出力データの *x* 軸（mesh=tet でのみ有効） 2 次元表示
file =	file name	axis の数だけ定義する
resfile =	(省略可, `D=file`)	再開始計算時の過去タリーファイル名。複数 `axis` の 場合でも 1 つのみ指定。
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	*x*-軸テキスト *y*-軸テキスト *z*-軸テキスト
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 `gshow=5` は，`icntl=8` の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時), 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示。 この下に xyz 形状メッシュセクションが必要
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェック を行わない。領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める 分解能を各辺 `resol` 倍します。 gshow, rshow オプションの時、領域境界を表示する 線の太さを定義します。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、`[volume]` セクションで定義された 体積の値が使用されます。 また、`reg=`において `( )` 等により特殊な領域指定を行った場合は、 volume を省略すると、インプットエコーに PHITS 内部で与えられた領 域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数

表 116: [t-deposit] パラメータ (4)

name	値	説明
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、 各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ1 辺のスキャン数
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイル を作成。ファイル名は出力ファイルの拡張子を eps に 変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、 axis= xy, yz, xz, rz, t-eng, eng-t の場合は無効。
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。 ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkout =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルを出力する。 ファイル名は出力ファイルの拡張子を vtk に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkfmt =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルのフォーマット。 テキスト形式（0）またはバイナリ形式（1）を選択する。
foamout =	0(省略時), 1, 2	1 でタリー出力結果（数値データ）を OpenFOAM フィールドデータ形式で出力する。ファイル名は出力ファイルの拡張子を foam に変えたファイル名。 2 でタリー出力結果（四面体要素番号、重心 *x* 座標、*y* 座標、*z* 座標、体積、数値データ、相対誤差）を CSV 形式（コンマ区切り）で出力する。ファイル名は出力ファイルの拡張子を csv に変えたファイル名。 mesh=tet, axis=tet のときのみ有効。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値 i-th カウンターの最大値
chmin(i) = chmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	ヒストリー内で発生した全粒子の i-th カウンターの最小値。バッチ分散モード (istdev=1) では設定不可。 ヒストリー内で発生した全粒子の i-th カウンターの最大値。バッチ分散モード (istdev=1) では設定不可。
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
gslat =	2（省略時）, = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。（ただしボクセルファントムはきれいに描画されません。） 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。

表 117: [t-deposit] パラメータ (5)

name	値	説明
dresol	(省略可, D=0.0)	検出器の分解能を表すパラメータ 1。output=deposit の場合のみ有効。dresol=σ_r, dfano=F の場合、各イベントの付与エネルギー E は標準偏差が $\sqrt{\sigma_r^2 + FE}$ のガウス分布に従って分散される。ただし、dresol<0 の値を入力すると、エネルギー分解能（エネルギー依存性やピーク形状などは）“usresol.f” で定義されたもので置き換わる。
dfano=	(省略可, D=0.0)	検出器の分解能を表すパラメータ 2。output=deposit の場合のみ有効。dresol=σ_r, dfano=F の場合、各イベントの付与エネルギー E は標準偏差が $\sqrt{\sigma_r^2 + FE}$ のガウス分布に従って分散される。
stdcut=	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。
nlatcel	（省略可）	Lattice 内にある同一セル番号へのエネルギー付与量を各格子別に扱って統計処理したい場合に，その Lattice を定義するセル番号（lat コマンドのあるセル番号）を指定する。1 つのみ指定可能で，省略もしくは 0 が指定された場合は各格子別に扱わない。output=deposit のときのみ有効。
nlatmem	（省略可、D=1000）	nlatcel を利用した際の 1 ヒストリー内で別個に扱うセルの最大個数。

一般に fano factor は無次元量として定義されていますが、dfano パラメータはエネルギーの次元をもつ量として定義されていますのでご注意ください。例えば、Si 半導体検出器で fano factor を 0.1、一対の電子正孔対を生成するのに必要な平均エネルギーを 3.62eV とすると、dfano=3.62E-06 * 0.1 = 3.62E-07 となります。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

reg = (100 200 300 ···) のような複数領域における付与エネルギーの総和を出力する機能の拡張として、条件に応じて重み付けを行う機能を追加しました。本機能を用いる事で、下式に示すように、指定した領域 i へのヒストリー毎の付与エネルギー $E_{\text{history},i}$ に対して、条件に応じて異なる係数 $\alpha_i(N_{\text{list}})$ を乗じ、複数領域の deposit energy の総和 E_{history} を算出します。

$$E_{\text{history}} = \sum_i \alpha_i(N_{\text{list}}) \times E_{\text{history},i} \tag{26}$$

本機能はヒストリー毎の付与エネルギーに応じて重み付けを行う機能で、特定領域への付与エネルギーに応じて電荷収集効率が変化するような、半導体ソフトエラー解析などに利用できます。エネルギーや LET など粒子の条件に応じて付与エネルギーに重み付けを行いたい場合は、ユーザー定義重み付け関数（usrdfn1.f や usrdfn2.f）を作成する必要があります。本機能は mesh = reg のみ対応しており、reg = weightsum とする事で動作します。付与エネルギーに乗じる係数のリストを複数定義し、ヒストリー毎に指定した条件に応じて異なる係数リストを使用する事が可能です。本機能を使用する際の書式は次の通りです。

```
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
```

まず、`ncond = m` で条件の数 *m* を指定します。この次の行には、条件番号 (no)、領域番号 (`cell`)、演算子 (`operator`)、閾エネルギー (`ethres`)、リスト番号 (`list`) のデータ順を定義します。これらのデータの並びを変えるときは、この行で指定してください。この定義文の下に `ncond =`で指定した数だけ条件を定義します。条件は、「1 ヒストリーにおいて、領域番号 (`cell`) への deposit energy が、閾エネルギー (`ethres`) に対して演算子 (`operator`) の関係を満たす場合、後述するリスト番号 (`list`) の係数を使用する」ことを意味します。条件に合致するか否かのチェックは上の行から順に行うため、いくつかの条件に合致するヒストリーに対しては、先に指定した条件が優先されます。全ての条件に合致しないヒストリーに対しては、`list00` に記載した係数が使用されます。条件番号 (no) に and と記載することで複数の条件を設定できます。領域番号 (`cell`) では、領域をまとめる `( 2 - 5 8 9 )` という書式が使えます。また、`( 6 < 10[1 0 0] < u=3 )` などの lattice, universe 構造も指定できます。ただし、単一の数字でない場合は必ず `( )` で括ってください。演算子 (`operator`) では、greater than(`gt`), greater equal(`ge`), equal(`eq`), less eqaul(`le`), less than (`lt`) の 5 種類を使用できます。閾エネルギー (`ethres`) の値の単位は MeV です。

続いて、`ncell = n` で総和を取る対象とするセルの数 *n* を指定します。この次の行には、領域番号 (`cell`)、リスト番号 (`list`*xx* のデータ順を定義します。これらのデータの並びを変えるときは、この行で指定してください。この定義文の下に `ncell =`で指定した数だけ領域番号と係数を定義します。こちらの領域番号 (`cell`) でも、領域をまとめる書式、lattice, universe 構造を指定できます。単一の数字でない場合は必ず `( )` で括ってください。なお、同じ領域番号を複数定義すると deposit energy が積算されますので注意してください。`list00` を定義しない場合、全ての係数の値がゼロに設定されます。

7.6 [T-Deposit2] セクション

このタリーは、[t-deposit] を 2 つの領域で同時に実行し、付与エネルギーをタリーしてその相関などを見るものです。dE, E カウンターなどの模擬に用いることができ、2 つの領域の付与エネルギーの相関を 2 次元プロットすることなどができます。[t-deposit] において output=deposit とした場合と同様に、検出器の応答を模擬することができます。このタリーでは、[t-deposit] タリー同様、任意の物質の LET(dE/dx) の関数のユーザー定義のファクターを乗じることができます。これは、usrdfn1.f, usrdfn2.f で定義されます。

表 118: [t-deposit2] パラメータ (1)

name	値	説明
mesh = reg =	reg r1 r2	形状メッシュ、reg のみ 必ず 2 つの領域番号を指定
part =	all(省略時), 粒子名	ひとつの [t-deposit2] セクション最大 6 個まで
letmat1 = letmat2 =	(省略可) (省略可)	領域 r1 に対する LET(dE/dx) を評価する物質番号。 省略時は、実際の物質。 領域 r2 に対する LET(dE/dx) を評価する物質番号。 省略時は、実際の物質。 物質を指定する場合の密度は、[Material] セクションで定義された密度となる。したがって、水に対する LET 分布を計算したい場合、[Material] セクションで、水の密度が 1g/cm^3 となるように定義する必要がある。 また、負の場合は電子・陽電子の dE/dx として 1g/cm^3 の水に対する値が参照される。詳しくは、奨励設定の particletherapy を参照してください。
dedxfnc1 = dedxfnc2 =	(省略可, D=0) (省略可, D=0)	領域 r1 に対して、0:ファクター無し、 1: usrdfn1 を使用、2: usrdfn2 を使用。 領域 r2 に対して、0:ファクター無し、 1: usrdfn1 を使用、2: usrdfn2 を使用。 サンプルプログラムとして、usrdfn1.f には粒子の LET に ICRP60 で定義された $Q(L)$ 関係を乗じて線量当量を導出するプログラムが、usrdfn2.f には microdosimetric kinetic model に基づいて生物学的線量を導出するプログラムが入っています。
e1-type = e2-type =	1, 2, 3, 4, 5 1, 2, 3, 4, 5	領域 r1 に対するエネルギーメッシュ、 エネルギーメッシュサブセクションが必要。 領域 r2 に対するエネルギーメッシュ、 エネルギーメッシュサブセクションが必要。
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ、時間メッシュサブセクションが必要
unit =	1, 2	1: Number [1/source] 2: Number [1/nsec/source]

表 119: [t-deposit2] パラメータ (2)

name	値	説明
axis =	eng1, eng2, t, e12, e21, t-e1, t-e2, e1-t, e2-t	出力データの x 軸 2 次元表示
file =	file name	axis の数だけ定義する
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の 場合でも 1 つのみ指定。
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された体積の値が使用されます。 また、reg=において ( ) 等により特殊な領域指定を行った場合は、volume を省略すると、インプットエコーに PHITS 内部で与えられた領域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイル を作成。ファイル名は出力ファイルの拡張子を eps に 変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、axis= e12, e21, t-e1, t-e2, e1-t, e2-t の場合は無効。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値 i-th カウンターの最大値
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。

[t-deposit2] では、荷電粒子のエネルギー付与だけをタリーするので、タリーする領域に入る粒子ごとのエネルギー付与を part=で指定することはできません。領域に入る粒子毎の付与エネルギーをタリーするには、[counter] で part=を用いて、特定の入射粒子から引き起こされる付与エネルギーをカウンターで指定する必要があります。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

7.7 [T-Heat] セクション

カーマ近似を用いて付与エネルギー（発熱）を出力するタリーです。ただしバージョン 3.05 以降 [t-deposit] でも同様の計算が可能となったため、それ以降のバージョンでは利用を奨励していません。なお、イベントジェネレータモードを用いた場合 (e-mode≥1) は中性子のカーマ近似は用いません。また、electron=1 かつ電子の輸送を含んでいる時は、光子のカーマ近似は用いずに電子の付与エネルギーから計算します。

表 120: [t-heat] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz	形状メッシュ、形状メッシュサブセクションが必要
axis =	reg, x, y, z, r, xy, yz, xz, rz	出力データの x 軸 2 次元表示
file =	file name	axis の数だけ定義する
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の場合でも 1 つのみ指定。
material =	(省略可) all, 数	スコアする material を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。数を指定した場合、その数だけの material を次の行に記述する。負の数にした場合は、それらの material を対象から外すことを意味する。
(次行)	2 5 8	material 番号
output =	heat simple	total: 全付与エネルギー total: 全付与エネルギー ncut, gcut, pcut: 計算打切エネルギー以下の中性子、光子、陽子によるエネルギー付与。(ただしページ下の注意書きをご覧ください。) leakage: 外部ボイドに出ていった粒子の運動エネルギー。 更に、total: 全付与エネルギー recoil: 計算打切エネルギー (emin(15-19)) 設定時の残留核の運動エネルギー。 ionization: 荷電粒子のエネルギーによる付与エネルギー。 low neutron: 中性子カーマ係数を用いた際の付与エネルギー。 photon: 光子カーマ係数を用いた際の付与エネルギー。ただし、electron=1 のときは、カットオフ電子による付与エネルギー。 others: 残留核の残留エネルギー。igamma=1 の場合、励起状態にある残留核から光子が放出されるのでゼロに近づく。

計算打切エネルギー emin 以下の中性子、光子、陽子の扱いに注意して下さい。計算打切エネルギー以下の中性子、光子、陽子であっても、[parameters] セクションにおいてこれらの情報を出力しない設定 (incut=0, igcut=0, ipcut=0:デフォルト値) の場合、上の ncut, gcut, pcut の成分には含まれず、stopped particle のところにタリーされます。incut>0, igcut>0, ipcut>0 の時のみ、ncut, gcut, pcut にタリーされます。

[t-heat] について全般的に言えることですが、最終的に熱に変換するエネルギーは、荷電粒子の ionization の過程を通してです。しかしながら、輸送コードの中では、cutoff エネルギーがあり、最後の過程まで追わずに輸送を終了します。従って、熱の成分として recoil、others などの成分が出力されます。これらの成分は、輸送のパラメータにより変化します。

表 121: [t-heat] パラメータ (2)

name	値	説明
output = (続き)	all	simple の結果に加えて、以下の各粒子の寄与を書き出す。 recoil に対する $d, t,$ ^{3}He, α, residual nuclei の寄与。 ionization に対する p, π^+, π^-, others の寄与。(ionization に関しては part で指定した粒子の寄与も出力するが、eps ファイルには出力されない。) stopped particle: 物質中で停止した粒子の運動エネルギーの proton, neutron, π^+, π^-, others の寄与。(stopped particle に関しては part で指定した粒子の寄与も出力するが、eps ファイルには出力されない。) others: 残留核の励起エネルギー (remaining excitation energy) と fission の成分が出力される。 axis が 2 次元表示のときは、all は simple と同じで、total, recoil, ionization, low neutron, electron, others が出力される。
part =	粒子名 (省略可)	output=all の場合に、ここで指定した荷電粒子による付与エネルギー（ionization 成分）と物質中で停止した粒子のエネルギー量（stopped particle 成分）を出力する。ただし、eps ファイルには出力されない。 また、カーマ近似を使用した計算において光子と中性子による寄与を調べる場合、part=photon neutron としてもこれらの寄与を区別できないので注意してください。各々の寄与を求める場合は output=simple として出力される内容をご確認ください。
unit =	0, 1, 2	0: [Gy/source] 1: [MeV/cm^3/source] 2: [MeV/source]
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション

unit = 0 とした場合、[Gy/source] を単位とする吸収線量が出力されます。mesh = reg の場合は、volume パラメーターを用いるか、[volume] セクションにおいて各領域の体積を設定する必要があります。ただし、吸収線量は不可算量ですので、各 x 軸に関する積分値 (sum over) は出力されません。また、領域内に複数の物質が混在する場合、その領域の吸収線量は全体の平均値とならないことにご注意ください。例えば、領域の 1 と 2 にある質量 M_1 と M_2 の物質にそれぞれ E_1 と E_2 の熱量が吸収された場合、その平均吸収線量は $\frac{E_1+E_2}{M_1+M_2}$ ですが、PHITS で計算する値は $\frac{E_1}{M_1}\frac{V_1}{V_1+V_2}+\frac{E_2}{M_2}\frac{V_1}{V_1+V_2}$ となります。ここで、V_1 と V_2 はそれぞれ領域 1 と 2 が占める体積を表します。

表 122: [t-heat] パラメータ (3)

name	値	説明
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は，icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時), 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示。 この下に xyz 形状メッシュセクションが必要
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域 を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める 分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する 線の太さを定義します。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された 体積の値が使用されます。 また、reg=において ( ) 等により特殊な領域指定を行った場合は、 volume を省略すると、インプットエコーに PHITS 内部で与えられた領 域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、 各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ 1 辺のスキャン数
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイル を作成。ファイル名は出力ファイルの拡張子を eps に 変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、 axis= xy, yz, xz, rz の場合は無効。
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。 ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkout =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルを出力する。 ファイル名は出力ファイルの拡張子を vtk に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkfmt =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルのフォーマット。 テキスト形式（0）またはバイナリ形式（1）を選択する。

表 123: [t-heat] パラメータ (4)

name	値	説明
electron =	0(省略時), 1	電子の寄与のオプション 0: 光子のカーマ係数を使う 1: 電子のエネルギー付与から計算する。 この場合、電子の輸送が必要
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値 i-th カウンターの最大値
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
gslat =	2（省略時）, = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。(ただしボクセルファントムはきれいに描画されません。) 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。
	0	0 で gshow で lattice の境界を非表示
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

7.8 [T-Yield] セクション

生成核種を出力します。 dmax(2) 以下のエネルギーの中性子による生成核種はタリーされません。ただし、e-mode≥1 の時は、タリーされます。

表 124: [t-yield] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz, tet	形状メッシュ、形状メッシュサブセクションが必要。
special =	D=0 (省略可)	> 0 の時タリー領域での核反応を special 回繰り返し、また統計崩壊を反応あたり 10 回繰り返し、核種生成断面積の統計を上げる。
part =	all(省略時) 反応入射粒子名	ひとつの [t-yield] セクションで最大 6 個まで指定可能。 入射粒子を限定する。
material =	(省略可) all, 数	スコアする material を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの material を次の行に記述する。 負の数にした場合は、それらの material を対象から外すことを意味する。
(次行)	2 5 8	material 番号
mother =	(省略可) all, 数	核種生成をする母核を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの核種を次の行に記述する。 負の場合は、それらの母核を対象から外すことを意味する。
(次行)	208Pb Pb	質量数を指定すれば、その核。 質量数を指定しなければ、Pb の同位体全てを指定。 複数の母核群を指定したいときは、複数の [t-yield] セクションを定義する。
nucleus =	(省略可) all, 数	出力する核種生成を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの核種を次の行に記述する。
(次行)	208Pb Pb	質量数を指定すれば、その核。 質量数を指定しなければ、Pb の同位体全てを指定。
elastic =	1（省略時）,0,-1	弾性散乱による反跳原子核の寄与を含めるかどうかのオプション。 1: 含める。 0: elastic collision の recoil された標的核を除く。 -1: 標的核と同じ原子核が生成された場合は除く。
unit =	1, 2	1: [1/source] 2: [1/cm^3/source]

[t-yield] セクションでは、part によりタリーする入射粒子を指定できますが、出力されるのはそれらの寄与を足し合わせた結果となります。入射粒子毎の寄与を分けてタリーしたい場合は、複数の [t-yield] セクションを設定してください。

表 125: [t-yield] パラメータ (2)

name	値	説明
axis =	reg, x, y, z, r, tet, xy, yz, xz, rz mass charge chart dchain	出力データの x 軸 出力データの x 軸（mesh=tet でのみ有効） 2 次元表示 質量数分布を表示。nucleus を指定した場合 isotope 分布 原子番号に関する分布を表示。nucleus の指定不可 nucleus chart 形式 (x 軸:N, y 軸:Z)。 nucleus の指定不可 DCHAIN 用の出力。全ての isotope を出力。 mesh = reg のみ [t-yield] のみ、1 つのタリーセクションにつき 1 つの axis しか指定できません。
file =	file name	出力ファイル名を定義する。
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。
output =	(省略可) product cutoff	タリーするタイミングを変更します。省略時は product 核反応で生成された核種をタリーする。デフォルト。 エネルギーカットオフで止まった核種をタリーする。 原子核を輸送させない場合は、product と同じになる。
info =	0, 1	chart のとき stable nuclei と magic number を出力
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト

output=cutoff を指定したときは、part, mother の指定は無視されます。

[parameters] セクションにおいて igamma=3 とすると、γ 崩壊とアイソマー生成が EBITEM モデルを用いて考慮されます。その際、axis=chart, dchain とした場合に、生成されたアイソマーの情報も出力されます。

表 126: [t-yield] パラメータ (3)

name	値	説明
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は，icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時) 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示。 この下に xyz 形状メッシュセクションが必要
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域 を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める 分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する 線の太さを定義します。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された体積の値が使用されます。 また、reg=において ( ) 等により特殊な領域指定を行った場合は、volume を省略すると、インプットエコーに PHITS 内部で与えられた領域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、 各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ1 辺のスキャン数
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。 ファイル名は出力ファイルの拡張子を eps に変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、 axis= xy，yz，xz，rz，chart，dchain の場合は無効。
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。 ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。 mesh=xyz，axis=(xy，yz，xz のいずれか) のときのみ有効。
vtkout =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルを出力する。 ファイル名は出力ファイルの拡張子を vtk に変えたファイル名。 mesh=xyz，axis=(xy，yz，xz のいずれか) のときのみ有効。
vtkfmt =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルのフォーマット。 テキスト形式（0）またはバイナリ形式（1）を選択する。

カウンターの動作契機を反応を起こした時 (coll) とした場合、ctmin, ctmax で参照するのは反応後の値、すなわち変化後のカウンター値です。

nucleus で出力させる核種を指定し、stdcut を与えることで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

ndata=1 の時、使用する核データの反応チャンネルは、以下のものです。

^{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	

ndata=2,3 の時、file(27) のフォルダに格納する放射化断面積のファイル名は、“Pd107-y-n.dat” のように、“[標的核の元素記号]+[質量数（3 桁）]+[-y-]+[入射粒子 (p or n)]+[.dat]” と指定してください。ただし、元素記号が 1 文字の場合は “O_” のように、アンダーバーを付記してください。また、入射粒子として指定できるのは陽子 (p) と中性子 (n) のみです。ファイルの中のフォーマットは、

```
# 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
 .......
```

のように、ZAP, LIP, INT によりそれぞれ生成核種、アイソマー状態、内挿方法を指定します。生成核種は $1000Z + A$（3 桁）のフォーマットで指定し、そのアイソマー状態を 0: 基底状態、1:第 1 準安定状態、2:第 2 準安定状態の番号により指定します。次の INT = (1,2,3,4,5) は、内挿方法の指定で、

```
INT = 1  : Histogram
      2  : Linear - Linear
      3  : Log - Linear
      4  : Linear - Log
      5  : Log - Log
```

となっています。2 行目は、

```
# Elab (MeV)    sigma (b)
```

で、1 コラム目に入射エネルギー [MeV]、2 コラム目に断面積 [Barn] を書き並べてください。ここで、エネルギービン数は自由です。ただし、入射エネルギーが用意したエネルギー領域より低い場合と高い場合は、それぞれ指定した最小と最大のエネルギーにおける断面積を使用します。タリーでは、このデータから He より大きい核種をタリーします。

表 127: [t-yield] パラメータ (4)

name	値	説明
foamout =	0(省略時), 1, 2	1 でタリー出力結果（数値データ）を OpenFOAM フィールドデータ形式で出力する。ファイル名は出力ファイルの拡張子を foam に変えたファイル名。 2 でタリー出力結果（四面体要素番号、重心 x 座標、y 座標、z 座標、体積、数値データ、相対誤差）を CSV 形式（コンマ区切り）で出力する。ファイル名は出力ファイルの拡張子を csv に変えたファイル名。 mesh=tet, axis=tet のときのみ有効。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値 i-th カウンターの最大値
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
gslat =	2（省略時）, = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。（ただしボクセルファントムはきれいに描画されません。） 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。
	0	0 で gshow で lattice の境界を非表示
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。
ndata =	0(省略時), 1, 2, 3	1 を指定すると、核子入射で標的核が α, ^{14}N, ^{16}O の場合の核種生成断面積に核データを用います。 2 を指定すると、核データを用いている核種を標的にする核反応で、核データを使うエネルギー領域 (e-mode 領域を除く) の生成核種断面積を file(27) で指定しているフォルダ内のファイルから読み込んで用います。 3 を指定すると、核データを用いていない核種についても、核反応の生成核種断面積を file(27) で指定しているフォルダ内に存在するファイルから読み込んで用います。この時、このフォルダにある核種は必ず（e-mode 以外、モデル計算も）このデータを用います。

7.9 [T-Product] セクション

核反応による生成粒子、ソースの発生粒子をタリーします。[t-yield] との違いは、エネルギー分布や、時間分布が取れることです。データライブラリを用いて反応を模擬する光子、電子、dmax(2) 以下の低エネルギー中性子による反応の生成粒子はタリーしていません。ただし、e-mode≥1 を用いると核データを用いた中性子反応からの生成粒子、残留核をタリーすることができます。

表 128: [t-product] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz, tet	形状メッシュ、形状メッシュサブセクションが必要
part =	all(省略時), 粒子名	ひとつの [t-product] セクション最大 6 個まで
material =	(省略可) all, 数	スコアする material を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの material を次の行に記述する。 負の数にした場合は、それらの material を対象から外すことを意味する。
(次行)	2 5 8	material 番号
mother =	(省略可) all, 数	核反応をする母核を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの核種を次の行に記述する。負の数にした場合は、それらの母核を対象から外すことを意味する。 output=atomic のときは使用不可。
(次行)	208Pb Pb	質量数を指定すれば、その核。 質量数を指定しなければ、Pb の同位体全てを指定。 複数の母核群を指定したいときは、複数の [t-product] セクションを定義する。
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ エネルギーメッシュサブセクションが必要
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ、時間メッシュサブセクションが必要
a-type =	1, 2, -1, -2 (省略可)	角度メッシュ (1, 2 :cos, −1, −2 :degree) 角度メッシュサブセクションが必要

表 129: [t-product] パラメータ (2)

name	値	説明
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, reg, x, y, z, r, tet, xy, yz, xz, rz t, cos, the	出力データの x 軸 出力データの x 軸（mesh=tet でのみ有効） 2 次元表示 時間 (t)、角度 (cos, the)
file =	file name	axis の数だけ定義する
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の場合でも 1 つのみ指定。

unit = 5, 6, 15, 16, 25, 26, 35, 36 の場合の Lethargy はエネルギーに関する自然対数目盛を表しています。各エネルギービンの上限と下限がそれぞれ $E_{\mathrm{high}}, E_{\mathrm{low}}$ のときに、各 Lethargy の幅を $\ln(E_{\mathrm{high}}/E_{\mathrm{low}})$ で与えます。

unit = 21 - 26, 31 - 36 の場合の sr は、立体角の単位ステラジアン (steradian) を表しています。

表 130: [t-product] パラメータ (3)

name	値	説明
output =	source nuclear (default) nonela elastic decay fission atomic	ソースの発生粒子 核反応からの生成粒子、elastic を含む 非弾性散乱からの生成粒子 弾性散乱からの生成粒子 崩壊反応からの生成粒子 核分裂からの生成粒子 原子相互作用からの生成粒子
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は, icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時), 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示。 この下に xyz 形状メッシュセクションが必要
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域 を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める 分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する 線の太さを定義します。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された体積の値が使用されます。 また、reg=において ( ) 等により特殊な領域指定を行った場合は、volume を省略すると、インプットエコーに PHITS 内部で与えられた領域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、 各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ 1 辺のスキャン数

カウンターの動作契機を反応を起こした時 (`coll`) とした場合、`ctmin`, `ctmax` で参照するのは反応後の値、すなわち変化後のカウンター値です。

`[t-product]` で、dump を指定したときは、e-type, t-type の指定は、上限と下限の意味しかもちません。unit は 1 に固定です。dump 定義文を用いてダンプさせる時は、axis、file 数ともにひとつに制限されます。dump するデータを書き込むファイルは、file 定義文で指定した名前に "_dmp" を付けたファイルです。file 定義文で定義された file 名には、通常のタリーの結果が書き出されます。(バージョン 2.66 以前では、file 定義文で指定したファイルに dump するデータが書き込まれ、通常のタリー結果を出力するファイルとして `.cfg` をつけたファイルが作成されていました。) dump ファイルを使用した接続計算を `idmpmode=1` で実行するには、nocas および nobch の情報を出力しておく必要があります。また、`idmpmode=1` を使用するためには、ここで出力される "_dmp" が付いた dump ファイルと共に `file=`で指定した名前の通常のタリー出力ファイルが必要になります。メモリ共有型並列計算を実行する場合、本機能による dump ファイルの作成と `dumpall` によるイベント情報の書き出しは共存できませんのでご注意ください。

ソースに dump ファイルを用い、`[t-product]` で、dump を指定し、output = source、icntl = 6 を指定すると、dump ファイルの加工ができます。既存の dump ファイルから、特定の粒子や、特定のエネルギーの情報、また、情報の数を減らしたファイルなどを、新しい dump ファイルとして作成できます。

`stdcut` を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。`stdcut` を正とし、`[parameters]` セクションにおいて `itall=0,1` とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として `stdcut` よりも小さくなった場合に、そのバッチで計算を終了します。また、`stdcut` を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

表 131: [t-product] パラメータ (4)

name	値	説明
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。ファイル名は出力ファイルの拡張子を eps に変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、axis= xy，yz，xz，rz の場合は無効。
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。 ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。 mesh=xyz，axis=(xy，yz，xz のいずれか) のときのみ有効。
vtkout =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルを出力する。 ファイル名は出力ファイルの拡張子を vtk に変えたファイル名。 mesh=xyz，axis=(xy，yz，xz のいずれか) のときのみ有効。
vtkfmt =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルのフォーマット。 テキスト形式（0）またはバイナリ形式（1）を選択する。
foamout =	0(省略時), 1, 2	1 でタリー出力結果（数値データ）を OpenFOAM フィールドデータ形式で出力する。ファイル名は出力ファイルの拡張子を foam に変えたファイル名。 2 でタリー出力結果（四面体要素番号、重心 x 座標、y 座標、z 座標、体積、数値データ、相対誤差）を CSV 形式（コンマ区切り）で出力する。ファイル名は出力ファイルの拡張子を csv に変えたファイル名。 mesh=tet，axis=tet のときのみ有効。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値 i-th カウンターの最大値
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
dump = (次行)	データ数 データ定義	ファイルに粒子の情報をダンプします。mesh=reg のみ。 正で与えた時はバイナリー、負の時はアスキーファイル。 データの順番を定義します。ただし、idmpmode=1 の使用のためには、nocas および nobch の情報が必要です。
gslat =	2（省略時）， = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。（ただしボクセルファントムはきれいに描画されません。） 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。

7.10 [T-DPA] セクション

“原子あたりのはじき出し数 (DPA; Displacement Per Atom)” の関数で評価される放射線損傷は、原子炉や加速器施設等で使用される材料の劣化を表す指標で、大強度ビーム、特に重イオンに関して重要な課題の一つです。原子のはじき出しは入射粒子と標的原子の衝突を起源として起こり、飛ばされた原子 (一次はじき出し原子, PKA; Primary Knock on Atom) が再度別の原子に衝突することによって連鎖的に発生します。このタリーではその頻度を DPA として出力します。輸送荷電粒子のクーロン散乱断面積からの寄与を含んでおり、重イオン入射の場合にも適切に評価されます。また、e-mode≥1 を用いて、低エネルギーの中性子入射反応から生成する荷電粒子による DPA の寄与も計算できます。

表 132: [t-dpa] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz, tet	形状メッシュ、形状メッシュサブセクションが必要
part =	all(省略時), 反応入射粒子名	ひとつの [t-dpa] セクション最大 6 個まで
material =	(省略可) all, 数	スコアする material を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの material を次の行に記述する。 負の数にした場合は、それらの material を対象から外すことを意味する。
(次行)	2 5 8	material 番号
unit =	1, 2	1: DPA [x1.E-24/source] 2: DPA [1/source]
idpa =	0(省略時), 1	0: 欠陥の非熱的再結合補正を考慮しない dpa (NRT-dpa) 1: 欠陥の非熱的再結合補正を考慮する dpa (arc-dpa)
axis =	reg, x, y, z, r, tet, xy, yz, xz, rz	出力データの x 軸 出力データの x 軸（mesh=tet でのみ有効） 2 次元表示
file =	file name	axis の数だけ定義する
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の場合でも 1 つのみ指定。
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション

表 133: [t-dpa] パラメータ (2)

name	値	説明
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	*x*-軸テキスト *y*-軸テキスト *z*-軸テキスト
output =	dpa	total DPA の出力
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は, icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時), 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示。 この下に xyz 形状メッシュセクションが必要
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域 を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める 分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する 線の太さを定義します。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された体積の値が使用されます。 また、reg=において ( ) 等により特殊な領域指定を行った場合は、volume を省略すると、インプットエコーに PHITS 内部で与えられた領域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、 各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ1 辺のスキャン数

表 134: [t-dpa] パラメータ (3)

name	値	説明
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。ファイル名は出力ファイルの拡張子を eps に変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、axis= xy, yz, xz, rz の場合は無効。
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。 ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkout =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルを出力する。 ファイル名は出力ファイルの拡張子を vtk に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkfmt =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルのフォーマット。 テキスト形式（0）またはバイナリ形式（1）を選択する。
foamout =	0(省略時), 1, 2	1 でタリー出力結果（数値データ）を OpenFOAM フィールドデータ形式で出力する。ファイル名は出力ファイルの拡張子を foam に変えたファイル名。 2 でタリー出力結果（四面体要素番号、重心 x 座標、y 座標、z 座標、体積、数値データ、相対誤差）を CSV 形式（コンマ区切り）で出力する。ファイル名は出力ファイルの拡張子を csv に変えたファイル名。 mesh=tet, axis=(tet) のときのみ有効。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値 i-th カウンターの最大値
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
gslat =	2（省略時）, = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。（ただしボクセルファントムはきれいに描画されません。） 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

7.11 [T-LET] セクション

任意の物質の LET(dE/dx) の関数として、track length や、dose をタリーします。[t-heat] と異なり、ここでの dose は、荷電粒子のエネルギー付与だけをタリーします。LET の単位は、keV/μm です。[T-LET] を使い、中性子の輸送計算を行う場合は、Event Generator mode (e-mode≥1) にする必要があります。また、電子に関しては、計算打切エネルギー (emin) 以下の粒子の寄与はタリーさせませんので、ご注意ください。

表 135: [t-let] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz	形状メッシュ、形状メッシュサブセクションが必要
part =	all(省略時), 粒子名	ひとつの [t-let] セクション最大 6 個まで
material =	(省略可) all, 数	スコアする material を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの material を次の行に記述する。 負の数にした場合は、それらの material を対象から外すことを意味する。
(次行)	2 5 8	material 番号
letmat =	(省略可)	LET(dE/dx) を評価する物質番号、省略時は、実際の物質。物質を指定する場合の密度は、[Material] セクションで定義された密度となる。したがって、水に対する LET 分布を計算したい場合、[Material] セクションで、水の密度が 1g/cm^3 となるように定義する必要がある。 また、負の場合は電子・陽電子の dE/dx として 1g/cm^3 の水に対する値が参照される。詳しくは、奨励設定の particletherapy を参照してください。
l-type =	1, 2, 3, 4, 5	LET メッシュ [keV/μm] LET メッシュサブセクションが必要。 ただし、メッシュを細かく設定しすぎる (1 桁あたり 20 分割以上) と、得られた LET スペクトルに物理的に意味のないピークが生じますのでご注意ください。

表 136: [t-let] パラメータ (2)

name	値	説明
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] ただし $\int f(L)dL = 1$ に規格化。 14: $L * d(L)$ [keV/μm] ただし $\int d(L)dL = 1$ に規格化。
axis =	let,reg,x,y,z,r, xy, yz, xz, rz	出力データの x 軸。 2 次元表示。
file =	file name	axis の数だけ定義する。
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。axis が複数の場合でも 1 つのみ指定。
factor =	(省略可、D=1.0)	Normalization factor.
title =	(省略可)	出力ファイルヘッダーのタイトル。
angel =	(省略可)	angel パラメータ。
sangel =	(省略可)	sangel パラメータ。
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション。
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト。 y-軸テキスト。 z-軸テキスト。

Version 3.02 より新たな unit(=13,14) が加わりました。どちらもマイクロドジメトリ分野でよく使われる形式で、$f(L)$ と $d(L)$ は、それぞれ LET(L) に対する飛跡及び線量の確率密度 (probability density) を表します。unit=13, 14 の結果は、それぞれ unit=2,4 の結果と比例関係にありますが、確率密度関数の積分値が 1 に規格化されているため、その絶対値が異なります。なお、unit=13,14 は、axis=let のみ有効となります。

表 137: [t-let] パラメータ (3)

name	値	説明
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時), 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示。この下に xyz 形状メッシュセクションが必要。
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する線の太さを定義します。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された体積の値が使用されます。 また、reg=において（ ）等により特殊な領域指定を行った場合は、volume を省略すると、インプットエコーに PHITS 内部で与えられた領域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数。
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ1 辺のスキャン数。
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。ファイル名は出力ファイルの拡張子を eps に変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、axis= xy, yz, xz, rz の場合は無効。
ctmin(*i*) = ctmax(*i*) =	(省略可、D=-9999) (省略可、D= 9999)	*i* 番のカウンターの最小値。 *i* 番のカウンターの最大値。
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義。
gslat =	2（省略時）, = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。(ただしボクセルファントムはきれいに描画されません。) 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。

表 138: [t-let] パラメータ (4)

name	値	説明
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

7.12 [T-SED] セクション

PHITS は、巨視的な体系内での放射線挙動を解析する計算コードですので、マイクロドジメトリなどで使われる DNA や細胞サイズの微視的な領域内での付与エネルギー（lineal energy (y) もしくは specific energy (z)）分布を、`[t-deposit]` や `[t-heat]` などを用いて直接計算することはできません。そこで、飛跡構造解析の結果に基づいて構築した計算式を用いて、微小領域におけるエネルギー付与分布を計算するタリー `[t-sed]` を導入しました。このタリーを用いれば、δ 線やオージェ電子による寄与も考慮して、微小領域における y 分布や z 分布を計算することができます。なお、sed の名前の由来は、Specific Energy Distribution です。計算方法の詳細は、下記の文献[89] [90] を参照してください。また、`[delta ray]` セクションによってコントロールする δ 線の発生機能は、本タリーとは併用できませんのでご注意ください。

`[t-sed]` は任意の物質中の微小領域内付与エネルギー分布をタリーします。ただし、水以外の物質に対する精度検証は行っていませんので、ご注意ください。その際、`[t-heat]` と異なり、dose は、荷電粒子のエネルギー付与だけをタリーします。したがって、`[t-sed]` を使い中性子の輸送計算を行う場合は、event generator mode (e-mode≥1) にする必要があります。微小領域の付与エネルギーを表す単位として、deposit energy ε (MeV), lineal energy y (keV/μm), specific energy z (Gy) があります。各単位の詳細は、ICRU Report36 をご参照ください。

表 139: `[t-sed]` パラメータ (1)

name	値	説明
`mesh =`	`reg, r-z, xyz`	形状メッシュ。形状メッシュサブセクションが必要。
`part =`	`all`(省略時), 粒子名	ひとつの `[t-sed]` セクションで最大 6 個まで指定可能。
`material =`	(省略可) `all`, 数	スコアする `material` を限定する。複数定義可。 `all` : デフォルト。この場合省略した場合と同じ。 数を指定した場合、その数だけの `material` を次の行に記述する。 負の数にした場合は、それらの `material` を対象から外すことを意味する。
(次行)	2 5 8	物質番号。
`letmat =`	(省略可)	LET(dE/dx) を評価する物質番号、省略時は、実際の物質。 物質を指定する場合の密度は、`[material]` で定義された密度となります。したがって、水に対する LET 分布を計算したい場合、`[material]` で、水の密度が 1g/cm^3 となるように定義する必要があります。水以外の物質に対する精度検証は行っていませんので、ご注意ください。また、負の場合は電子・陽電子の dE/dx として 1g/cm^3 の水に対する値が参照されます。詳しくは、奨励設定の `particletherapy` を参照してください。
`se-unit =`	1,2,3	微小領域の付与エネルギー単位。 1: Deposit energy ε [MeV] 2: Lineal energy y [keV/μm] 3: Specific energy z [Gy]
`cdiam =`	(省略可、D=1,0)	微小領域（球）の直径。単位は μm。

[89] 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).

[90] 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).

表 140: [t-sed] パラメータ (2)

name	値	説明
se-type =	1, 2, 3, 4, 5	付与エネルギーメッシュ（単位は se-unit で指定）。 エネルギーメッシュサブセクションが必要（ne，emin，emax などで指定）。 実行中に'Warning: Z bin is not enough!' と出た場合は、ここで指定する specific energy（もしくは lineal energy）の範囲が狭すぎて、マイクロドジメトリ関数で分布が正しく計算できないことを示しています。この場合、emin を下げる、emax を上げる、ne を大きくするなどの対応が必要です。微少領域の直径が 1μm の Lineal energy を計算する場合 (cdiam=1.0，se-unit=2)、se-type=3，emin=0.01，emax=10000.0, ne=60 以上が適当です。
unit =	1, 2, 3, 4, 5, 6 7,8	1: Track [cm/(keV/μm)/source] 2: Dose [MeV/(keV/μm)/source] $y * f(y)$ に比例 3: Track [cm/ln(keV/μm)/source] 4: Dose [MeV/ln(keV/μm)/source] $y * d(y)$ に比例 5: Track [cm/source] 6: Dose [MeV/source] 7: $y * f(y)$ [dimensionless] ただし $\int f(y)dy = 1$ に規格化。 8: $y * d(y)$ [keV/μm] ただし $\int d(y)dy = 1$ に規格化。 (se-unit=2 の場合。se-unit=1,3 の場合は、(keV/μm) が MeV もしくは Gy となる。)
axis =	sed,reg,x,y,z,r, xy, yz, xz, rz	出力データの x 軸。 2 次元表示。
file =	file name	axis の数だけ定義する。
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数の axis がある場合でも 1 つのみ指定。
factor =	(省略可、D=1.0)	Normalization factor.
title =	(省略可)	出力ファイルヘッダーのタイトル。
angel =	(省略可)	angel パラメータ。
sangel =	(省略可)	sangel パラメータ。
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション。

Version 3.02 より新たな unit(=7,8) が加わりました。どちらもマイクロドジメトリ分野でよく使われる形式で、$f(y)$ と $d(y)$ は、それぞれ Lineal energy, y に対する飛跡及び線量の確率密度 (probability density function) を表します。unit=7,8 の結果は、それぞれ unit=2,4 の結果と比例関係にありますが、前者は確率密度の積分値が 1 に規格化されているため、その絶対値が異なります。なお、unit=7,8 は、axis=sed のみ有効となります。

表 141: [t-sed] パラメータ (3)

name	値	説明
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト。 y-軸テキスト。 z-軸テキスト。
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は, icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時), 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示。 この下に xyz 形状メッシュセクションが必要
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める 分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する 線の太さを定義します。
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、 各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ 1 辺のスキャン数
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイル を作成。ファイル名は出力ファイルの拡張子を eps に 変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、 axis= xy, yz, xz, rz の場合は無効。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値 i-th カウンターの最大値
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
gslat =	2(省略時), = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。(ただしボクセルファントムはきれいに描画されません。) 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。

表 142: [t-sed] パラメータ (3)

name	値	説明
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

7.13 [T-Time] セクション

Energy cut off により輸送を止められた粒子と escape 粒子、decay 粒子の個数を時間タリーで出力します。時間の単位は nsec です。

このタリーを用いると、energy cut off 粒子や escape 粒子、decay 粒子のエネルギースペクトルを取ることができます。特に、energy cut off 粒子だけに注目して発生時のスペクトルを取るタリーは、このタリー以外にはありません。

表 143: [t-time] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz	形状メッシュ、形状メッシュサブセクションが必要
part =	all(省略時), 粒子名	ひとつの [t-time] セクション最大 6 個まで
material = (次行)	(省略可) all, 数 2 5 8	スコアする material を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの material を次の行に記述する。 負の数にした場合は、それらの material を対象から外すことを意味する。 material 番号
t-type =	1, 2, 3, 4, 5	時間メッシュ 時間メッシュサブセクションが必要
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ エネルギーメッシュサブセクションが必要
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	出力データの x 軸 2 次元表示
file =	file name	axis の数だけ定義する
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の場合でも 1 つのみ指定。
output =	all cutoff escape decay	energy cut off 粒子、escape 粒子および decay 粒子 energy cut off 粒子 escape 粒子 decay 粒子
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ

表 144: [t-time] パラメータ (2)

name	値	説明
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は, icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時), 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示 この下に xyz 形状メッシュセクションが必要。
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域 を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める 分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する 線の太さを定義します。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義され た体積の値が使用されます。 また、reg=において ( ) 等により特殊な領域指定を行った場合は、 volume を省略すると、インプットエコーに PHITS 内部で与えられた 領域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、 各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ 1 辺のスキャン数
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイル を作成。ファイル名は出力ファイルの拡張子を eps に 変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、 axis= xy, yz, xz, rz の場合は無効。
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。 ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkout =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルを出力する。 ファイル名は出力ファイルの拡張子を vtk に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkfmt =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルのフォーマット。 テキスト形式（0）またはバイナリ形式（1）を選択する。

表 145: [t-time] パラメータ (2)

name	値	説明
ctmin(i) =	(省略可、D=-9999)	i-th カウンターの最小値
ctmax(i) =	(省略可、D= 9999)	i-th カウンターの最大値
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
dump =	データ数	ファイルに粒子の情報をダンプします。mesh=reg のみ。 正で与えた時はバイナリー、負の時はアスキーファイル。
(次行)	データ定義	データの順番を定義します。ただし、idmpmode=1 の使用のためには、nocas および nobch の情報が必要です。
gslat =	2(省略時),	gshow を指定した際の lattice の境界線に関するオプション。
	= 0	境界線を描画しない。
	= 1	境界線を描画する。(ただしボクセルファントムはきれいに描画されません。)
	= 2	同じセルの場合は境界線を描画しない。
	= 3	同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。

[t-time] で、dump を用いることができるのは、output = cutoff だけです。dump を指定したときは、e-type, t-type の指定は、上限と下限の意味しかもちません。unit は 1 に固定です。dump 定義文を用いてダンプさせる時は、axis、file 数ともにひとつに制限されます。dump するデータを書き込むファイルは、file 定義文で指定した名前に "_dmp" を付けたファイルです。file 定義文で定義された file 名には、通常のタリーの結果が書き出されます。(バージョン 2.66 以前では、file 定義文で指定したファイルに dump するデータが書き込まれ、通常のタリー結果を出力するファイルとして.cfg をつけたファイルが作成されていました。) dump ファイルを使用した接続計算を idmpmode=1 で実行するには、nocas および nobch の情報を出力しておく必要があります。また、idmpmode=1 を使用するためには、ここで出力される "_dmp" が付いた dump ファイルと共に file=で指定した名前の通常のタリー出力ファイルが必要になります。メモリ共有型並列計算を実行する場合、本機能による dump ファイルの作成と dumpall によるイベント情報の書き出しは共存できませんのでご注意ください。

[t-time] で dump を用いると、ncut、gcut、pcut で条件付けられた *PHITS* の繋ぎ計算用ファイルを作成できます。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

7.14 [T-Interact]（従来の [T-Star]） セクション

ある領域内で起きた反応回数を出力します。従来は、star density を計算するため [t-star] と名付けていましたが、原子相互作用などより一般的な反応の回数を計算できるよう拡張しましたので、version 3.04 より [t-interact] に名称変更しました。ただし、プログラム上は [t-star] という名称も引き続き利用することが可能です。なお、原子相互作用による反応回数をタリーするためには、EGS5 モードを使用してください (negs=1)。

表 146: [t-interact] パラメータ (1)

name	値	説明
mesh =	reg, r-z, xyz	形状メッシュ、形状メッシュサブセクションが必要。
MorP =	mean, prob(省略可)	反応数の平均値（mean）もしくはヒストリ毎の頻度分布（prob）を選択します。ただし、prob は mesh=reg のときのみ選択可能です。
part =	all(省略時), 反応入射粒子名	ひとつの [t-interact] セクションに最大 6 個まで。
material =	(省略可) all, 数	スコアする material を限定する。複数定義可。 all: デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの material を次の行に記述する。負の数にした場合は、それらの material を対象から外すことを意味する。
(次行)	2 5 8	material 番号
mother =	(省略可) all, 数	核反応をする母核を限定する。複数定義可。 all: デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの核種を次の行に記述する。負の数にした場合は、それらの母核を対象から外すことを意味する。output=atomic のときは使用不可。
(次行)	208Pb Pb	質量数を指定すれば、その核。質量数を指定しなければ、Pb の同位体全てを指定。複数の母核群を指定したいときは、複数の [t-interact] セクションを定義する。
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ。エネルギーメッシュサブセクションが必要。
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ。時間メッシュサブセクションが必要。
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	出力データの x 軸。 2 次元表示。 時間軸。 反応数（MorP=prob のときに選択）。
maxact =	(省略可、D=100)	axis=act のときの最大反応数。この数を超える反応数が起きた場合は、その確率が overflow の欄（sum over の手前）に出力されます。

表 147: [t-interact] パラメータ (2)

name	値	説明
file =	file name	axis の数だけ定義する。
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の場合でも 1 つのみ指定。
output =		出力させる反応の種類。1 種類のみ指定可能。
	all	全ての反応。
	decay	Decay 反応。
	elastic	弾性散乱。
	nuclear	核反応。ただし elastic は含まない。[t-product] の nonela に相当。
	fission	核分裂。
	absorption	吸収反応。
	heavyion	重イオン反応。
	transmut	標的核が核変換を起こした場合の反応。
	atomic	多重クーロン散乱を除いた原子相互作用。
	deltaray	デルタ線生成。
	knockelec	電子と陽電子による knock on electron 生成。
	atmflu	原子蛍光 X 線放出。
	auger	Auger 電子放出。
	brems	制動放射。
	photoelec	光電効果。
	compton	コンプトン散乱。
	pairprod	電子陽電子対生成。
	annih	陽電子消失。
	ets_elast	飛跡構造解析モードにおける弾性散乱 (elastic scattering)。
	ets_ioniz	飛跡構造解析モードにおける電離 (ionization)。
	ets_e-exc	飛跡構造解析モードにおける電子的励起 (electronic excitation)。
	ets_v-exc	飛跡構造解析モードにおける振動励起 (vibration excitation)。
	ets_p-exc	飛跡構造解析モードにおけるフォノン励起 (photon excitation)。
	ets_r-exc	飛跡構造解析モードにおける回転励起 (rotation excitation)。
	ets_dea	飛跡構造解析モードにおける解離性電子付着（dissociative electron attachment）。
	ets_ioniz_e-exc	飛跡構造解析モードにおける電離と電子的励起の和。
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル。
angel =	(省略可)	angel パラメータ。
sangel =	(省略可)	sangel パラメータ。
2d-type =	1,2,3,4,5,6,7 (省略可、D=3)	2 次元表示のオプション。
x-txt =	(省略可)	x-軸テキスト。
y-txt =	(省略可)	y-軸テキスト。
z-txt =	(省略可)	z-軸テキスト。

表 148: [t-interact] パラメータ (3)

name	値	説明
gshow =	0(省略時), 1, 2, 3, 4, 5	mesh=xyz, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示 gshow=5 は, icntl=8 の場合にピクセル形式で物質色を表示します。
rshow =	0(省略時), 1, 2, 3	mesh=reg, axis=xy,yz,xz の時、領域境界 (1)、 物質番号 (2)、領域番号 (3) を表示。 この下に xyz 形状メッシュセクションが必要。
ginfo =	0(省略時), 1 2	gshow, rshow オプションの時領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。
resol = width =	1(省略時) 0.5(省略時)	gshow, rshow オプションの時、領域境界を求める。 分解能を各辺 resol 倍します。 gshow, rshow オプションの時、領域境界を表示する線の太さを定義します。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された体積の値が使用されます。 また、reg=において (　) 等により特殊な領域指定を行った場合は、volume を省略すると、インプットエコーに PHITS 内部で与えられた領域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
iechrl =	72(省略時)	volume のインプットエコーの時の最大コラム数。
volmat =	(省略可、D=9)	xyz メッシュで material が指定されている時、各メッシュの volume correction を行う。(0 で無し) volmat の値は、xyz メッシュ1 辺のスキャン数。
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。ファイル名は出力ファイルの拡張子を eps に変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、axis= xy, yz, xz, rz の場合は無効。
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkout =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルを出力する。ファイル名は出力ファイルの拡張子を vtk に変えたファイル名。mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkfmt =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルのフォーマット。テキスト形式（0）またはバイナリ形式（1）を選択する。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値。 i-th カウンターの最大値。
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義。

表 149: [t-interact] パラメータ (4)

name	値	説明
gslat =	2（省略時）,	gshow を指定した際の lattice の境界線に関するオプション。
	= 0	境界線を描画しない。
	= 1	境界線を描画する。（ただしボクセルファントムはきれいに描画されません。）
	= 2	同じセルの場合は境界線を描画しない。
	= 3	同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥3 の場合でも、セル番号や Lat 番号は表示されなくなります。
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

7.15 [T-Dchain] セクション

このセクションでは、DCHAIN を実行するための入力ファイルを作成します。PHITS と接続計算用に整備した DCHAIN の接続計算の概要を図 60 に示します。DCHAIN の実行方法や例題については、`\phits\recommendation\DCHAIN` フォルダにある資料やサンプルインプットをご参照ください。また、DCHAIN の詳細は、`\phits\dchain-sp\manual` にあるマニュアルや DCHAIN-SP についてまとめられた資料[91] をご参照ください。

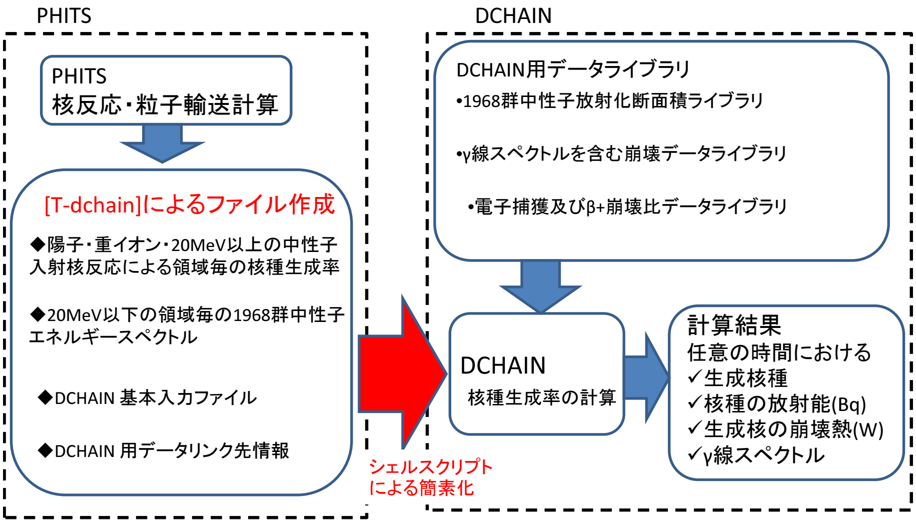


図 60: PHITS と DCHAIN の接続計算の概要。

`[t-dchain]` は、1968 群の中性子スペクトル、核種生成率、DCHAIN 用の基本入力ファイルを自動的に作成するためのタリーです。このタリーの出力を DCHAIN に入力することにより、任意の時間における生成核種の放射能、崩壊熱、γ 線スペクトルを計算できます。

DCHAIN は、20MeV 以下の中性子スペクトルに 1968 群の放射化断面積を乗じて 20MeV 以下の中性子による核種生成率を計算し、陽子、重イオン、中間子、20 MeV 以上の高エネルギー中性子による核種生成率と足し合わせます。その後、崩壊データを利用して照射中及び照射終了後の任意の時刻における核種の蓄積量、誘導放射能、γ 線エネルギースペクトル及び α、β 及び γ 線の放出による崩壊熱等を評価します。

PHITS 計算における粒子輸送や `[source]` セクションで指定する線源の時間分布による経過時間は、DCHAIN で考慮する経過時間とは無関係ですのでご注意ください。

なお、イベントジェネレータモードを使うと、低エネルギー中性子核反応による放射化も、DCHAIN のデータライブラリを使わずに PHITS で直接計算することができます。ただし、その計算精度は、データライブラリを使った場合と比べてあまり検証されていませんので、`[t-dchain]` タリーを使う場合は、イベントジェネレータモードをオフ (`e-mode=0`) として計算することを奨励いたします。また、最初から放射化している核種を `[material]` セクションで設定しても、その放射能の寄与は DCHAIN 計算に含まれないのでご注意下さい。

バージョン 3.00 以降では、`[material]` において質量数を指定せずに定義した元素について、天然存在比による展開を考慮した上で DCHAIN のインプットを作成するようにしました。ただし、同一核種を複数定義した場合は、後に定義した方しか考慮されないのでご注意ください。例えば、ある物質番号の組成として、

```
MAT[1] Fe 1  56Fe 1
```

と定義した場合、`Fe` に含まれる `56Fe` の寄与は DCHAIN 実行時には無視されます。

[91] 甲斐哲也 他, “DCHAIN-SP 2001: 高エネルギー粒子誘導放射能計算コード”, JAERI-Data/Code-2001-016 (2001).

表 150: [t-dchain] パラメータ (1)

name	値	説明
mesh =	reg, xyz, tet	形状メッシュ（reg, xyz, tet のみ対応）。 形状メッシュサブセクションが必要。（reg=セル番号）
file =	file name	DCHAIN のインプットファイル名。1 つだけ定義する。 拡張子は任意のものが使用できるが、.dtrk, .dyld, .dout は使用不可。
title =	（省略可）	出力ファイルヘッダーのタイトル。
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。
ndata =	0(省略時), 1, 2, 3	1 を指定すると、核子入射で標的核が α, ^{14}N, ^{16}O の場合の核種生成断面積に核データを用います。 2 を指定すると、核データを用いている核種を標的にする核反応で、核データを使うエネルギー領域 (e-mode 領域を除く) の生成核種断面積を file(27) で指定しているフォルダ内のファイルから読み込んで用います。 3 を指定すると、核データを用いていない核種についても、核反応の生成核種断面積を file(27) で指定しているフォルダ内に存在するファイルから読み込んで用います。この時、このフォルダにある核種は必ず（e-mode 以外、モデル計算も）このデータを用います。
iredufmt =	0,1(省略時)	[T-Dchain] タリーの出力形式の切り替えオプション。 0 の場合、古い形式を使用する。（普通の [T-Track] と [T-Yield] の形式） 1 の場合、[T-Dchain] のために用意された新しい形式を使用する。xyz や tet で大きなメッシュ数の場合にはこちらをご使用ください。
timeevo = (次行)	データ数 時間 相対強度	照射・冷却時間ステップの数。 照射時間または冷却時間、相対照射強度 (timeevo 個)。 ●照射・冷却時間ステップは、計算開始時からの通算時間ではなく、各照射・冷却ステップ毎の時間。時間の単位は、秒 (s)、分 (m)、時 (h)、日 (d)、年 (y) であり、数値と単位の間には 1 つ以上の空白が必要。例 : 5 秒 → 5 s, 22 日 → 22.0 d, 8.3 年 → 8.3e+0 y ●相対照射強度は、線源の基準強度 (amp) に対する相対強度で、照射時間ステップ毎に変化させることができる。例 : 基準強度と同じ場合は 1.0、冷却時間ステップの場合は 0.0 とする。※例 59 [t-dchain] タリーの入力例を参照。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

ndata の詳細については、7.8 を参照してください。

表 151: [t-dchain] パラメータ (2)

name	値	説明
outtime = (次行)	データ数 時間	計算結果出力時間の数 計算結果出力時間（outtime 個）。 ●正の値を指定した場合は計算開始時からの通算時間で、負の値を指定した場合は最後の照射時間ステップ終了後からの時間になる。時間の単位は、秒 (s)、分 (m)、時 (h)、日 (d)、年 (y) であり、数値と単位の間には 1 つ以上の空白が必要。照射中と冷却中の任意の時間を指定でき、照射・冷却時間ステップの区切り時間と一致する必要はない。ただし、timeevo で指定した値を超えた時間を指定することはできない。※例 59 [t-dchain] タリーの入力例を参照。 また、放射能が 0 のとき（例えば”0.0 m”など）の結果を出力するためには DCHAIN パラメータ iprtb2 を 0 としてください（“iprtb2 = 0” と [t-dchain] セクション内に書けば設定できます）。
amp =	（省略可, D=1.0）	線源の基準強度 (source/秒)。
target =	（省略可, D=0）	対象領域の構成物質（核種及び密度）の設定 なし：0, 次行以降の記載不要 あり：1, 次行以降の記載必要 ※構成物質に関するパラメータは、入力ファイルの [Material], [Cell], [Volume] セクションから自動的に設定されるので、通常は記述の省略が可能。 ※ [Material] で定義した核種以外を DCHAIN で計算したい場合や [Volume] で体積を定義しない場合は、次行以降に target=1 として対象領域に対する通し番号、cell 番号、体積の順番で入力し、tg-list サブセクションが必要。（例 60 をご参照ください。）
iertdcho =	(D=1) 1（省略時） 0	PHITS の統計誤差の DCHAIN 計算における伝播に関するオプション。（iredufmt=0 の場合、[t-yield] の結果の不確かさが部分的に “***_err.dyld” に出力されます。） DCHAIN 計算における伝播を評価し、その結果を放射化出力ファイル（“***.act”）に書き出します。 DCHAIN 計算における伝播を評価せず、結果の出力も行いません。
itdecs =	(D=1) 0 1（省略時）	核破砕反応によるターゲット減少に関するオプション。 核破砕反応によるターゲット減少を考慮しない。 考慮する。
itdecn =	(D=1) 0 1（省略時）	核データ領域の中性子反応によるターゲット減少に関するオプション。 核データ領域の中性子反応によるターゲット減少を考慮しない。 考慮する。

表 152: [t-dchain] パラメータ (3)

name	値	説明
`inxslib =`	(D=100)	DCHAIN で使用される中性子反応断面積ライブラリの選択。
	0,40	JEFF-3.1A （2009–2019 年の DCHAIN-SP 標準）
	1	FENDL/A-2.0 （2001–2009 年の DCHAIN-SP 標準）（古い 175 群の t-track 中性子フラックスを使用した場合のみ動作します。）
	2,20	JENDL/AD-2017
	21	JENDL-4.0[92]
	30	ENDF/B-VII.1[93]
	31	ENDF/B-VIII.0[94]
	41	JEFF-3.3[95]
	50	FENDL/A-3.0[96]
	51	EAF-2010[97]
	60	BROND-3.1[98]
	70	CENDL-3.1[99]Note that missing isotopic data for S, Cl, K, Ca, V, Zn, W, Hg, and Tl.
	90	TENDL-2017[100]ただし、ファイル容量が大きいためパッケージには含まれていません。必要な方は PHITS 事務局までご連絡ください。
	100	hybrid library composed of, in order: JENDL/AD-2017 + JENDL-4.0 + ENDF/B-VIII.0 + JEFF-3.3 + FENDL/A-3.0
	101	hybrid library of: ENDF/B-VIII.0 + JENDL/AD-2017 + 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/AD-2017 + JENDL-4.0
	120	hybrid library of: JENDL/AD-2017 + JENDL-4.0
	-1	`hnxslib` パラメータでライブラリ名を指定する。（詳細は DCHAIN のマニュアルをご覧ください。）

Hybrid library は、左側が優先ライブラリです。計算時にある反応チャンネルが含まれる場合はそのデータを利用し、含まれなければその右にあるライブラリを利用し、という順番でデータを探します。

[92] 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).
[93] 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.
[94] 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.
[95] A. Plompen et al., “The Joint Evaluated Fission and Fusion Nuclear Data Library, JEFF-3.3,” Submitted to The European Physical Journal (2019).
[96] R. A. Forrest, et al., “FENDL-3 Library-Summary Documentation,” Tech. Rep. INDC(NDS)-0628, International Atomic Energy Agency (Dec. 2012).
[97] J.-C. Sublet, et al., “The European Activation File: EAF-2010 neutron-induced cross section library,” Tech. Rep. CCFE-R (10) 05 (2010).
[98] A. I. Blokhin, et al., “New version of neutron evaluated data library BROND-3.1,” Yad. Reak. Konst, 2, 2, 62 (2016).
[99] 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).
[100] D. Rochman et al., “The TENDL library: hope, reality and future,” EPJ Web Conf., 146, 02006 (Sep. 2017).

表 153: [t-dchain] パラメータ (4)

name	値	説明
idcylib =	(D=5)	DCHAIN で使用される崩壊データライブラリの選択。
	0	DCHAIN の旧バージョンで使用されていたハイブリッドライブラリ（EAF-3.1, FENDL/D-1, ENSDF (1997), Table of Isotopes 8th ed. (1996), 核図表 1996）
	2	JENDL/DDF-2015
	3	ENDF/B-VIII.0
	4	ENDF/B-VIII.0（優先）と JENDL/DDF-2015 のハイブリッドライブラリ。
	5（省略時）	JENDL/DDF-2015（優先）と ENDF/B-VIII.0 のハイブリッドライブラリ。
acmin	(D=10^{-20})	放射能出力ファイルに印書する核種の下限条件。
	= 0.0	放射能の下限を自動設定（全放射能 $\times 10^{-10}$）
	> 0.0	放射能の下限値 [Bq] を指定。
	< 0.0	生成核種数密度の下限値 [atoms/cm3] を指定（-1 を乗じた負の値）。
istabl	(D=0)	放射能出力ファイルの安定核種印書オプション。
	= 0	印書しない。
	= 1	印書する。

これら以外にも、DCHAIN のインプットに書き込むパラメータ

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

を [t-dchain] セクションで直接指定することが可能です。各パラメータの意味や DCHAIN の入力パラメータおよび出力結果については、\phits\dchain-sp\manual にある DCHAIN のマニュアルをご参照下さい。

例 59: [t-dchain] タリーの入力例

```
        mesh = reg                      <-region メッシュの指定
         reg = 100                      <-計算領域の指定
        file = testDC.spd               <-DCHAIN 入力ファイル名の指定
       title = [t-dchain] test calc.
         amp = 1.0E12                   <-線源の基準強度 (source/秒)

     timeevo = 4                        <-照射・冷却時間ステップの数
                3.0 h  1.0              <-3 時間運転
                2.0 h  0.0              <-2 時間運転停止
                3.5 h  1.0              <-3.5 時間運転
               15.5 h  0.0              <-15.5 時間運転停止

     outtime = 3                        <-計算結果出力時間の数
                3.0 h                   <-運転開始から 3.0 時間後
               -1.0 h                   <-最後の照射終了時間から 1 時間後
               -3.0 h                   <-最後の照射終了時間から 3 時間後
```

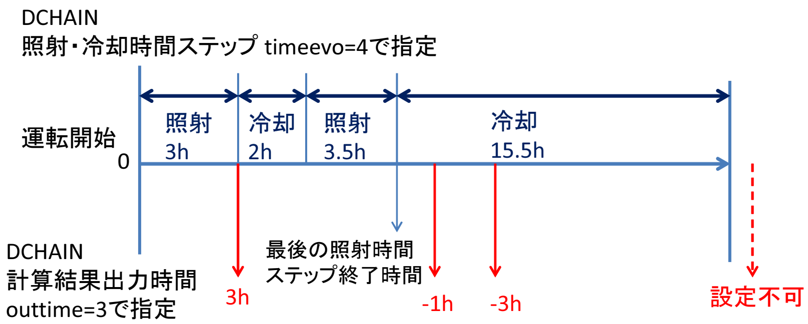


図 61: 例 59 における照射・冷却時間ステップと計算結果出力時間の関係。

例 60: [t-dchain] のパラメータ target=1 の場合の入力例

```
            .......
  :      target = 1                     <-対象領域の構成物質（核種及び密度）の設定有
  :      non     reg     vol            <-省略可
  :        1       1    8000.0          <-通し番号、cell 番号、体積 (cm^3)
  :     tg-list = 2                     <-対象領域構成物質に含まれる核種の数
  :              H-1    6.689E-02       <-核種の元素記号と原子数密度 (10^24 個/cm^3)
  :              O-16   3.345E-02
  :        2       2    2000.0          <-通し番号、cell 番号、体積 (cm^3)
  :     tg-list = 1                     <-対象領域構成物質に含まれる核種の数
  :             Fe-56   8.385E-02       <-核種の元素記号と原子数密度 (10^24 個/cm^3)
```

他に必要な PHITS 入力ファイルの設定及び注意事項は次のとおりです。

- [t-dchain] は、PHITS の入力ファイル 1 個につき、1 つしか設定できません。
- [Parameters] セクションにおいて次のパラメータの設定を行う必要があります。
  - jmout=1: 物質の原子数密度を表示させます。
  - file(21): DCHAIN 用データフォルダの場所を指定します。
- [Volume] セクションにおいて各領域の体積を指定します。PHITS には reg メッシュの体積を自動で計算する機能がないため、この入力が必須となります。

このセクションが作成するファイルで、DCHAIN 実行に必要なファイルを以下に示します。ここで、*** は `file=`で指定した、拡張子を除く名前を示します。

- DCHAIN の基本入力ファイル: `file` で指定したファイル。
- 20MeV 以下の中性子スペクトルファイル: “***.dtrk”
- 核種生成率ファイル: “***.dyld”
- DCHAIN 用データ格納フォルダのリンク名が書かれたファイル: “dch_link.dat”

DCHAIN を実行する際に、下記の名前をもつファイルは消えてしまうのでご注意ください。
“yield.out, out-gsdef, out-gamsporg, out-allreg, spd-act.out, angel-data.ang, out-phits, out-dcychains”

7.16 [T-WWG] セクション

[t-wwg] は、[weight window] セクションのパラメータを出力するタリーです。想定している 3 次元体系に対して、weight window 機能が効果的に動作するパラメータを自動で得ることができる Weight Window Generator (WWG) です。基本的には、まず [t-wwg] を設定したインプットで仮計算を行い、次に、得られた [weight window] のパラメータを加えたインプットで本計算を行う、という流れで本タリーを利用します。出力する [weight window] のパラメータは、仮計算で得られた各セルに飛んできた粒子の数[101] に基づいて決定しています。仮計算において、[t-wwg] で指定したセルに飛んできた粒子の数をタリーしており、その中で最も大きな値が 1 となるよう全ての値を規格化し、[weight window] のパラメータへと変換します。その際、一つも粒子が飛んで来なかったセルについては、最も数の少なかったセルと同じ weight window を設定します。パラメータは、part で指定した粒子毎に 1 セット出力します。もし、エネルギーメッシュや時間メッシュが定義されている場合は、それらのビン毎にパラメータのセットを書き出します。また、[volume] セクション等により各セルの体積を与えておく必要があります。

[t-wwg] を含む入力ファイルで再開始計算機能や sumtally 機能を利用する場合は、[parameters] セクションで ireschk=1 としてください。

[t-wwg] の入力書式は、以下の通りです。

表 154: [t-wwg] パラメータ (1)

name	値	説明
mesh =	reg, xyz	形状メッシュ。形状メッシュサブセクションが必要。reg と xyz のみ指定可。
part =	all(省略時), 粒子名	ひとつの [t-wwg] セクション最大 6 個まで
material =	(省略可) all, 数	スコアする material を限定する。複数定義可 all: デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの material を 次の行に記述する。
(次行)	2 5 8	material 番号。
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ。 エネルギーメッシュサブセクションが必要。
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ。エネルギーメッシュと併用不可。 時間メッシュサブセクションが必要
unit =	1(省略可)	1: [1/cm^2/source] 1 のみ指定可。
axis =	eng, reg, xy, yz, xz, t, wwg	出力データの x 軸。wwg 以外は [t-track] と同じ。 再開始計算が必要な場合は、2 つ以上の axis を定義し、最初の 1 つを reg, eng, t のどれかで指定する。
file =	file name	axis の数だけ定義する。
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。複数 axis の 場合でも 1 つのみ指定。

このタリーでは、mesh は reg と xyz のみ指定可能で、r-z は指定できません。これは、[weight window] セクションのパラメータが r-z の形式に対応していないためです。出力の axis には、基本的に、[weight window] セクションのパラメータを出力する wwg を指定してください。この他に、eng, reg, xy, yz, xz, t も指定できますが、これらは [t-track] で指定した場合と同じ結果しか得られず、[weight window] に関係する値は得られません。ここで、xy, yz, xz の 2 次元表示は rshow=1 を指定したときのみ有効です。

[101] 正確には、モンテカルロ粒子フルエンス、すなわち輸送粒子のウェイトを常に 1 にした粒子フルエンスに基づいて決定します

表 155: [t-wwg] パラメータ (2)

name	値	説明
factor =	(省略可、D=1.0)	normalization factor
title =	(省略可)	出力ファイルヘッダーのタイトル。
angel =	(省略可)	angel パラメータ
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト
rshow =	0(省略時), 1, 2, 3	axis=xy,yz,xz の時、領域境界 (1)、物質番号 (2)、 領域番号 (3) を表示。 この下に xyz 形状メッシュセクションが必要。
ginfo =	0(省略時), 1, 2	rshow 時の領域チェックのオプション。
resol = width =	1(省略時) 0.5(省略時)	rshow 時の表示オプション。 rshow 時の表示オプション。
volume reg vol	(省略可)	reg メッシュの時に、各領域の体積を定義します。この行の下に volume 定義文が必要です。省略した場合、[volume] セクションで定義された体積の値が使用されます。 また、reg=において ( ) 等により特殊な領域指定を行った場合は、volume を省略すると、インプットエコーに PHITS 内部で与えられた領域番号が出力されます。 volume 定義文。書式は 6.1.2 を参照。
epsout =	0(省略時), 1, 2	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。 ファイル名は出力ファイルの拡張子を eps に変えたファイル名。 2 とすることで更に誤差棒を表示する。ただし、 axis= xy, yz, xz, wwg の場合は無効。
ctmin(i) = ctmax(i) =	(省略可、D=-9999) (省略可、D= 9999)	i-th カウンターの最小値。 i-th カウンターの最大値。
trcl =	(省略可)	rshow での座標変換。
gslat =	2（省略時）, = 0 = 1 = 2 = 3	gshow を指定した際の lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。(ただしボクセルファントムはきれいに描画されません。) 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。

表 156: [t-wwg] パラメータ (3)

name	値	説明
stdcut =	(省略可、D=-1)	統計誤差による計算打ち切り機能のしきい値。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

wwg の出力例を以下に示します。

```
 [ 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
```

ここでは、中性子に対して、2 群のエネルギービンの weight window のパラメータが定義されています。パラメータの定義には c**で表される定数が使われています。この場合、c71-c73 が第 1 エネルギー群（E <1keV）に対するパラメータ c74-76 が第 2 エネルギー群（1keV< E <1GeV）に対するパラメータです。これらのパラメータの中でユーザー自身が変更する可能性があるパラメータは、各群に対して左端で定義される初期設定値が 0 のパラメータ c71 や c74 などです。これらのパラメータは、各群の weight window にある一定値を加える際、利用します。

7.17 [T-WWBG] セクション

[t-wwbg] は、[ww bias] のパラメータを出力するタリーです。本タリーは Weight Window Bias Generator (WWBG) となっており、[weight window] で与えるウエイトの下限値にバイアス（傾斜）をかけることができる [ww bias] のパラメータを自動的に用意することができます。[ww bias] は、特に、[t-wwg] を用いて自動的に [weight window] を作成した際、更に特定の領域にウエイト値のバイアスをかけたい場合に利用できます。図 62 に、[weight window] と [ww bias] を組み合わせた輸送計算を実行する際の流れを示しました。最初に [t-wwg] を用いて [weight window] を作成します。これで十分な場合は不要ですが、注目しているタリー領域が限定的である場合は、[ww bias] を作成して特定の領域にバイアスをかけることにより、効率的に統計量を増やすことが可能となります。[ww bias] を作成する方法は 2 通りあり、本節でご紹介する [t-wwbg] を利用して自動的に作成する方法と、手動で作成する方法です。[t-wwbg] は、1 つのインプットファイルにおいて 1 つのみ指定でき、[parameters] において icntl=15 とすることで動作します。[ww bias] の設定方法や手動で作成する方法については、5.20 節 WW Bias セクションをご覧ください。また、作成した [ww bias] と [weight window] を組み合わせた輸送計算を実行する場合は、[parameters] において icntl=0 および iwwbias=1 としてください。

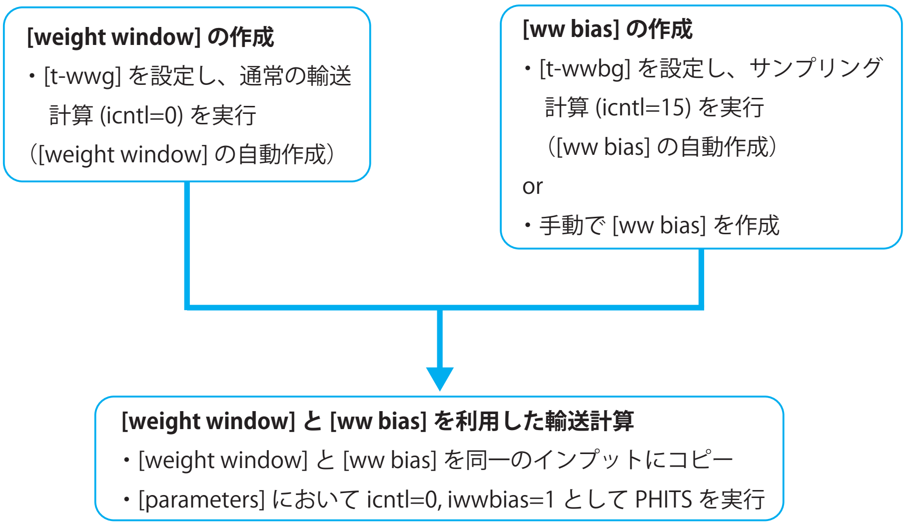


図 62: [weight window] と [ww bias] を組み合わせた輸送計算の概要。

[t-wwbg] で用意されるバイアス値は、図 63 にある始点 (x0,y0,z0) と終点 (x1,y1,z1) で決まるベクトルを中心軸とする複数の円柱領域において段階的に設定されます。図 63 に示したのは、3 つの大きさの違う円柱が入れ子構造となっている円柱領域の断面図です。円柱領域を指定するのに必要なパラメータは、上

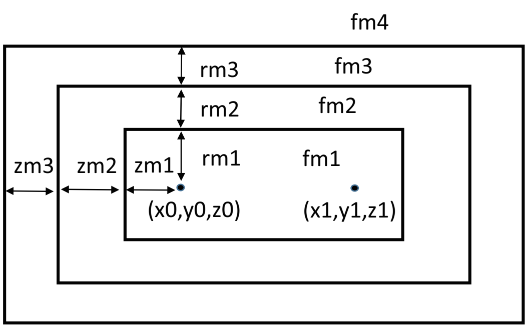


図 63: 円柱領域を指定する際のパラメータ。

記の始点と終点の座標の他、n-mesh，r-mesh，z-mesh，f-mesh です。n-mesh は入れ子構造を取る円柱の数です。r-mesh，z-mesh はそれぞれ円柱の半径と高さに関する差分量となっており、図 63 に示すように各円柱の大きさを定義します。これらはそれぞれ n-mesh の数と同じだけ指定する必要があります。また、f-mesh では、[t-wwbg] で指定したいバイアス値を円柱領域の内側から指定することができます。f-mesh として (n-mesh)+1 個のバイアス値が必要で、最後の値は定義された円柱領域外のバイアス値を意味します。これらの円柱領域は実際の輸送計算の体系とは無関係に定義することができ、指定された各円柱領域のバイアス値を参照しながら、実際の各領域のバイアス値を決定します。なお、このセクションで定義した円柱領域が [cell] で定義した外部ボイド領域に入ってしまう場合は、r-out を用いて外部ボイドを拡張することができます。

[t-wwbg] の入力書式は、以下の通りです。

表 157: [t-wwbg] パラメータ

name	値	説明
mesh =	reg	形状メッシュ。reg だけ指定可。
part =	all(省略時), 粒子名	ひとつの [t-wwbg] で最大 6 個まで指定可能。
e-type =	1, 2, 3, 4, 5	エネルギーメッシュ。エネルギーメッシュサブセクションが必要。
t-type =	1, 2, 3, 4, 5 (省略可)	時間メッシュ。時間メッシュサブセクションが必要。 エネルギーメッシュとの併用は不可。
axis =	xy，yz，xz，wwbg	再開始計算は不可。xy，yz，zx は rshow オプションが必要。
file =	file name	出力ファイル名。axis の数だけ定義する。
angel =	(省略可)	angel パラメータ。
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト。 y-軸テキスト。 z-軸テキスト。
rshow =	0(省略時), 1, 2, 3	axis=xy,yz,xz の時、領域境界 (1)、物質番号 (2)、 領域番号 (3) を表示。この下に xyz 形状メッシュセクションが必要。
ginfo =	0(省略時), 1, 2	rshow 時の領域チェックのオプション。
resol = width =	1(省略時) 0.5(省略時)	rshow 時の表示オプション。 rshow 時の表示オプション。
epsout =	0(省略時), 1	eps ファイルの出力オプション。ファイル名は、file で指定した出力ファイルの拡張子を eps に変えた名称。
x0，y0，z0 =		バイアスをかける円柱領域の始点の x, y, z 座標。
x1，y1，z1 =		バイアスをかける円柱領域の終点の x, y, z 座標。
n-mesh =	n	円柱領域の数。
r-mesh =	rm1,rm2,$\cdots$,rmn	円柱領域の半径の差分。n-mesh で指定した数だけ必要。
z-mesh =	zm1,zm2,$\cdots$,zmn	円柱領域の長さ方向の差分。n-mesh で指定した数だけ必要。
f-mesh =	fm1,fm2,$\cdots$,fmn,fm(n+1)	円柱領域のバイアス値。（n-mesh で指定した数）+1 だけ必要。
r-out =	(省略可、D=0)	拡張する外部ボイドの半径 [cm]。D=0 の場合は拡張なし。

[t-wwbg] の入力例を以下に示します。

例 61: [t-wwbg] の入力例

```
[ T - WWBG ]
     mesh =  reg
      reg = all
     axis = wwbg
     file = wwbg.out
     axis =  yz
     file = wwbyz.out
     part =  neutron
   e-type =    1
       ne =      2
             0.0  1e-3  1.0
       x0 =  0.0
       y0 = -20
       z0 =  50
       x1 =  0.0
       y1 =  20
       z1 =  150
   n-mesh =      3
   r-mesh =   10  10  10
   z-mesh =   10  10  10
   f-mesh =   1.0  0.5  0.1  0.05
    r-out =  1000
```

この例題では、始点と終点をそれぞれ (0, -20, 50), (0, 20, 150) としており、これらにより指定されるベクトルを中心軸とした 3 つの円柱領域が定義されています。図 64 に示したのが、この例題で用意されるバイアス値の空間分布です。3 つの円柱の半径は r-mesh により指定されており、一番内側のものから 10cm ずつ増加しています。また、これらの円柱の高さは z-mesh により指定されており、円柱の上面側と底面側にそれぞれ 10cm ずつ、すなわち、円柱の高さとして 20cm ずつ増加しています。バイアス値は、f-mesh により、円柱領域の内側から 1.0, 0.5, 0.1, 0.05 と段階的に設定されており、その変化の様子を図 64 の色の変化から確認することができます。

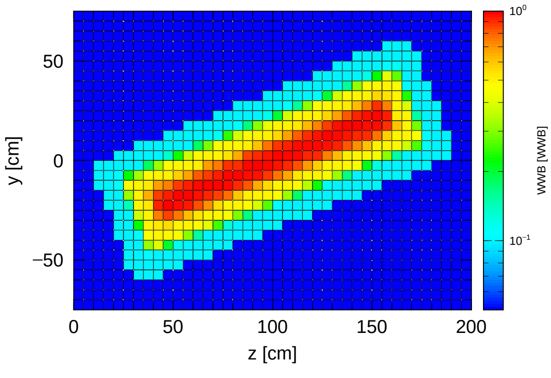


図 64: 例 61 により用意されるバイアス値の空間分布。

7.18 [T-Volume] セクション

各タリーにおいて mesh=reg とした場合、特定の unit の結果を正しく求めるためには、各セルの体積が必要となります。[t-volume] セクションは各セルの体積を自動で計算するためのタリーです。この体積自動計算機能では、各セルの物質を真空 (void) とし、特別な線源領域を設定したモンテカルロ積分を実行します。

本機能を使用する際は、[t-volume] セクションを設定し、[parameters] セクションにおいて icntl=14 としてください。本タリーは、1 つのインプットファイルにおいて 1 つのみ指定できます。

[t-volume] の入力書式は、以下の通りです。

表 158: [t-volume] パラメータ

name	値	説明
mesh =	reg	形状メッシュ。reg だけ指定可。
file =	file name	ファイ名を定義する。
resfile =	(省略可, D=file)	再開始計算時の過去タリーファイル名。
r-out =	(省略可, D=0.0)	外部ボイドの半径 (cm)。
title =	(省略可)	出力ファイルヘッダーのタイトル。
method =	(省略可,D=0)	モンテカルロ積分に関するオプション。 0: 粒子軌跡による積分。 1: 線源発生点による積分。
s-type =	1, 2	体積計算の線源の種類。 1: 球体系線源。原点 (x0, y0, z0) と半径 r0 が必要。 2: 直方体線源。x0, x1, y0, y1, z0, z1 が必要。
x0, y0, z0, x1, y1, z1, r0 =		線源の形状を定義する座標や半径。
stdcut =	(省略可, D=-1)	統計誤差による計算打ち切り機能のしきい値。

s-type により、体積計算時の線源の種類を指定できます。s-type=1 では、中心を (x0,y0,z0) とし半径を r0 とする球の表面から、法線内向きに cos 分布となるよう線源を発生させます。s-type=2 では、x=x0 と x1、y=y0 と y1、z=z0 と z0 の面により形作られる直方体の 6 面から、法線内向きに一様な線源を発生させます。いずれの種類の場合でも、対象とするセルが線源領域に十分な余裕をもって囲まれるようにしてください。

stdcut を指定することで、タリー結果の統計誤差を基準として、PHITS の計算を打ち切ることができます。stdcut を正とし、[parameters] セクションにおいて itall=0,1 とした場合に機能します。バッチ終了時、タリー結果の全ての統計誤差が相対値として stdcut よりも小さくなった場合に、そのバッチで計算を終了します。また、stdcut を指定したタリーセクションが複数ある場合は、全てのタリーセクションの結果が条件を満たした時に計算を終了します。

本機能を用いて PHITS を実行した場合、file(6) (D=phits.out) において出力される [source] には、体積計算で使用した線源の情報が書き出されます。インプットファイルにある [source] の内容は出力されませんのでご注意ください。

以下に [t-volume] の一例を示します。

例 62: [t-volume] の例題

```
[ T - V o l u m e ]
  mesh =  reg           # mesh type is region-wise
   reg = 101  102  103  104  105
  file = volume.out     # file name of output for [volume]
s-type =    1           # 1: Sphere source, 2: Rectangular source
     x0 =  0.0          # (D=0.0) x of sphere center
     y0 =  0.0          # (D=0.0) y of sphere center
     z0 =  0.0          # (D=0.0) z of sphere center
     r0 =  50.0         # radius of sphere
```

[t-volume] において mesh=reg とし、体積を求めたいセルの番号を reg=にて指定します。s-type の選択とこれに関連した x0 などのパラメータを設定することで、体積計算用の特別な線源領域を定義します。基本的に、対象とするセルを囲むように線源領域を決定してください。対象とするセルが線源領域を越えていると正しく計算できません。また、逆に線源領域を広く取り過ぎると、計算時間が長くなってしまいますのでご注意ください。求めた体積は、file=で指定したファイルに出力されます。この出力ファイルの形式は以下の通りです。

```
[ 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
          ....   ....     ....   ....
          ....   ....     ....   ....
```

[volume] セクションの形式で出力されるので、infl コマンド等により、icntl=0 とした本計算で使用できます。ここで、[volume] の最後の列にある数字は計算した体積の統計誤差（相対誤差）です。再開始計算のための情報も出力されていますので、統計誤差が大きい場合は、[parameters] に istdev=-1 or -2 を加えて、再開始計算を行なってください。

7.19 [T-Userdefined] セクション

ユーザー定義タリーは、通常のタリーでは得ることができない様々な物理量をタリーする機能です。各ユーザーが PHITS のソースファイル `usrtally.f` を Fortran 言語を用いて変更し、求める物理量を書き出します。このため、ソースファイルのコンパイルが必要となります。

インプットファイルで `[t-userdefined]` セクションを定義することにより、PHITS 実行時に `subroutine usrtally` を通過するようになります。通過のタイミングは `dumpall` の書き出しと同じで、あらゆるイベントが発生した時です。本タリーの基本的な考え方として、`subroutine usrtally` で示されている変数を if 文等の条件で選別してカウントすることで、求める物理量を引き出します。この `subroutine` で扱うことができる変数を以下で説明しますが、中には値を変更すると PHITS の計算がおかしくなる変数もあるので、十分に注意してください。本タリーの具体的な利用方法については、`\phits\utility\usrtally` フォルダにある readme やサンプルインプットをご参照ください。

`[t-userdefined]` セクションでは、書き出し用ファイル名を与える `file` と `subroutine usrtally` で使用できるパラメータの数を指定する `nudtvar` およびそのパラメータの値 `udtvar`(*i*) [102]が定義できます。これらは再コンパイルせずに変更できる変数となります。

表 159: `[t-userdefined]` パラメータ

name	値	説明
`file` =	file name	`subroutine usrtally` 内で使用できる書き出し用ファイル。 最大 50 まで定義できる（装置番号=151-200）。
`nudtvar` = `udtvar`(*i*) =	0（省略時） 数値 (D=0)	定義する `udtvar`(*i*) の数。 `subroutine usrtally` 内で使用できるパラメータ。 *i* =`nudtvar` まで定義できる。

例 63: `[T-Userdefined]` パラメータの入力例

```
       file = output1.dat       <-装置番号は 151 で、subroutine usrtally 内では iudtf(1)
       file = output2.dat       <-装置番号は 152 で、subroutine usrtally 内では iudtf(2)
    nudtvar = 2                 <-subroutine usrtally 内では nudtvar
 udtvar(1) = 20.0               <-subroutine usrtally 内では udtvar(1)
 udtvar(2) = -10.0              <-subroutine usrtally 内では udtvar(2)
```

[102] バージョン 3.01 以前の `udtpara`*i*（$i = 0 - 9$）と同様の役割を果たします。バージョン 3.02 以降の PHITS でも `udtpara`*i* は使用可能です。

以下が `subroutine usrtally` 内で示されている変数となります。

(1) NCOL:
これはプログラムの中の変数で、粒子輸送のタイミングの認識変数です。

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

(2) npe, me:
それぞれメモリ分散型並列計算時の並列 PE(Processor Element) 数と各 Element の識別番号。

(3) ipomp,npomp:
それぞれメモリ共有型並列計算時のスレッド識別番号と使用している全スレッド数。

(4) iusrtally:
`subroutine usrtally` への通過をコントロールするパラメータ。インプットファイルに `[T-Userdefined]` セクションがあれば 1 となっています。

(5) iudtf(50):
`file` で定義したファイルの装置番号 (151-200)。

(6) nudtvar:
`nudtvar` として定義した数。使用できる `udtvar`(*i*) の上限値。

(7) udtvar(*i*):
`udtvar`(*i*) で定義した値。*i*=`nudtvar` まで使用できる。インプットファイルで定義していない `udtvar`(*i*) には 0 が入っている。

(8) NOCAS, NOBCH, RCASC, RSOUIN:

NOCAS	:	current event 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 event
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:
電子の阻止能 (dE/dx)。

(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:
階層構造をもつ CELL の変数です。

(b) ILAT2:
階層構造をもつ CELL の変数です。

(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 Note that 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	:	weight 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:
粒子がもつ電荷 (charge state)。

(19) NCLSTS:
意味は、衝突もしくは崩壊による生成粒子の個数です。原子核も含みます。

(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, KCOLL はそれぞれ次のような意味を持ちます。

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:
これらの変数は、各生成粒子の情報を記述するもので、NCLSTS だけ繰り返し、また、それぞれの変数も以下のように配列を持ちます。

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] セクション

xyz メッシュで指定する面の領域境界を表示します。この機能は、[parameters] で icntl=7 を指定すれば、輸送計算を行うことなく、出力させることができます。

output=10（他のタリーの gshow=5 オプション）は，Lattice や Tetra 構造が細かすぎてジオメトリに空白が出来てしまう場合に選択してください。

表 160: [t-gshow] パラメータ (1)

name	値	説明
mesh =	xyz	形状メッシュ xyz のみ、形状メッシュサブセクションが必要
axis =	xy，yz，xz	2 次元表示
file =	file name	axis の数だけ定義する
output =	1，2	1: 境界線、2: 境界線+物質色
	3，4	3: 境界線+物質番号、4: 境界線+物質色+物質番号
	5，6	5: 境界線+領域番号、6: 境界線+物質色+領域番号
	7，8	7: 境界線+LAT 番号、8: 境界線+物質色+LAT 番号
	9，10	9: 全て表示しない、10: ピクセル形式で物質色を表示
resol =	1(省略時)	領域境界を求める分解能を各辺 resol 倍します。
width =	0.5(省略時)	領域境界を表示する線の太さを定義します。
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
x-txt =	(省略可)	x-軸テキスト
y-txt =	(省略可)	y-軸テキスト

output=7, 8 は、最下層のセルが Lattice それ自身の時のみ、 Lattice の成分番号を (4,1,2) の形式で出力します。例えば、5.6.4 節で示した図 47 は、次の様な [t-gshow] のインプットで出力しています。

例 64: [t-gshow] の例題

```
[ T - gshow ]
     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 = 8
     file = cell-example6.dat
   epsout = 1
```

表 161: [t-gshow] パラメータ (2)

name	値	説明
epsout =	0(省略時), 1	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。ファイル名は出力ファイルの拡張子を eps に変えたファイル名
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。 ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkout =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルを出力する。 ファイル名は出力ファイルの拡張子を vtk に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
vtkfmt =	0(省略時), 1	タリー出力結果の三次元可視化用ファイルのフォーマット。 テキスト形式（0）またはバイナリ形式（1）を選択する。
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
gslat =	2（省略時）, = 0 = 1 = 2 = 3	Lattice の境界線に関するオプション。 境界線を描画しない。 境界線を描画する。（ただしボクセルファントムはきれいに描画されません。） 同じセルの場合は境界線を描画しない。 同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。
ginfo =	2(省略時), 0 1 2	領域エラーチェックのオプション 領領域エラーチェックを行わない。 領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域を表示する。 エラー領域表示に加えて領域エラーファイル (.err) を出力。

7.21 [T-Rshow] セクション

xyz メッシュで指定する面の領域境界と、与えられた領域毎の物理量に比例した色でその領域を塗ります。通常は、reg メッシュで得られた物理量を value 定義文に書きこみ xyz メッシュで指定する面に領域の色として出力することを目的とします。従って、この機能は、パラメーターセクションで icntl = 9 を指定した時だけ 出力させることができます。

物理量に対して、色の変化を線形でするか、対数でするかは、ANGEL パラメーターで、zlog, zlin で指定して下さい。デフォルトは、zlin です。

表 162: [t-rshow] パラメータ (1)

name	値	説明
mesh =	xyz	形状メッシュ xyz のみ、形状メッシュサブセクションが必要
axis =	xy, yz, xz	2 次元表示
file =	file name	axis の数だけ定義する
output =	1, 2, 3	1: 境界線、2: 境界線+物質番号、 3: 境界線+領域番号
resol = width =	1(省略時) 0.5(省略時)	領域境界を求める分解能を各辺 resol 倍します。 領域境界を表示する線の太さを定義します。
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	*x*-軸テキスト *y*-軸テキスト *z*-軸テキスト
reg = value reg val		領域定義 各領域の物理量を定義します。 この行の下に value 定義文が必要。 value 定義文。書式は volume 定義文と同じ。6.1.2 を参照。
iechrl =	72(省略時)	value のインプットエコーの時の最大コラム数

表 163: [t-rshow] パラメータ (2)

name	値	説明
epsout =	0(省略時), 1	1 で出力ファイルを ANGEL で処理した eps ファイルを作成。ファイル名は出力ファイルの拡張子を eps に変えたファイル名
bmpout =	0(省略時), 1	二次元ビットマップ画像ファイルを出力する。 ファイル名は出力ファイルの拡張子を bmp に変えたファイル名。 mesh=xyz, axis=(xy, yz, xz のいずれか) のときのみ有効。
trcl =	(省略可)	r-z, xyz メッシュの座標変換番号もしくは座標変換定義
gslat =	2（省略時）,	Lattice の境界線に関するオプション。
	= 0	境界線を描画しない。
	= 1	境界線を描画する。（ただしボクセルファントムはきれいに描画されません。）
	= 2	同じセルの場合は境界線を描画しない。
	= 3	同じ物質の場合は線なし。この場合、Lattice や連続四面体以外でも、隣接セルが同じ物質の場合は境界線が描画されなくなります。また、gshow≥ 3 の場合でも、セル番号や Lat 番号は表示されなくなります。
ginfo =	2(省略時),	領域エラーチェックのオプション
	0	領領域エラーチェックを行わない。
	1	領域エラーチェックを行い 2 次元ジオメトリ図上にエラー領域を表示する。
	2	エラー領域表示に加えて領域エラーファイル (.err) を出力。

例えば、5.6.4 節の図 47 に色を付ける場合、次の例題で示すような [t-rshow] タリーを用いると、図 65 の結果を得ることができます。

例 65: [t-rshow] の例題

```
[ 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 )
```

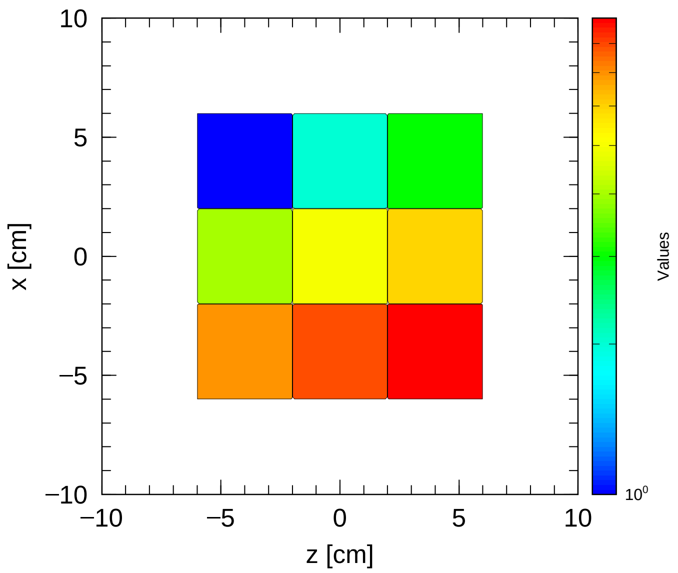


図 65: [t-rshow] の例題を用いて図 47 の各領域に色をつけた結果。

7.22 [T-3Dshow] セクション

幾何形状の 3 次元パース図を出力します。このタリーは、パラメーターセクションで icntl = 11 を指定した時だけ出力させることができます。

表 164: [t-3dshow] パラメータ (1)

name	値	説明
output =	0, 1, 2, 3 (省略時 3)	0: ドラフト、 1: 境界線のみ、 2: 境界線無し、 3: 境界線+カラー
material = (次行)	(省略可) all, 数 2 5 8	表示する material を限定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの material を次の行に記述する。 負の数にした場合は、それらの material を表示しないことを意味する。 material 番号
x0 = y0 = z0 =	(D=0.0) (D=0.0) (D=0.0)	視線、光線の原点の座標。 この点と視点を結んだ線が画面の中央になる。 視点、光源の座標は、この点を原点とする。
e-the = e-phi = e-dst =	(D=80) (D=140) (D=w-dst*10)	視点の z 方向からの角度 θ(degree) 視点の x 方向からの方位角 ϕ(degree) 視点の視点原点からの距離 (cm)
l-the = l-phi = l-dst =	(D=e-the) (D=e-phi) (D=e-dst)	光源の z 方向からの角度 θ(degree) 光源の x 方向からの方位角 ϕ(degree) 光源の光線原点からの距離 (cm)
w-wdt = w-hgt = w-dst = w-mnw = w-mnh = w-ang =	(D=100) (D=100) (D=200) (D=100) (D=100) (D=0.0)	画面フレームの幅 (cm) 画面フレームの高さ (cm) 画面フレームの視点原点からの距離 (cm)。画面フレームの中央と視点原点を結ぶ直線は、画面フレーム平面に垂直で、かつ視点を通る。 画面横方向のメッシュ数 画面高さ方向のメッシュ数 画面の回転 (degree)
heaven =	(D=y)	画面上方向の向き x, -x, y, -y, z, -z で指定
mirror =	(D=0)	=-1 で画面の左右鏡面変換

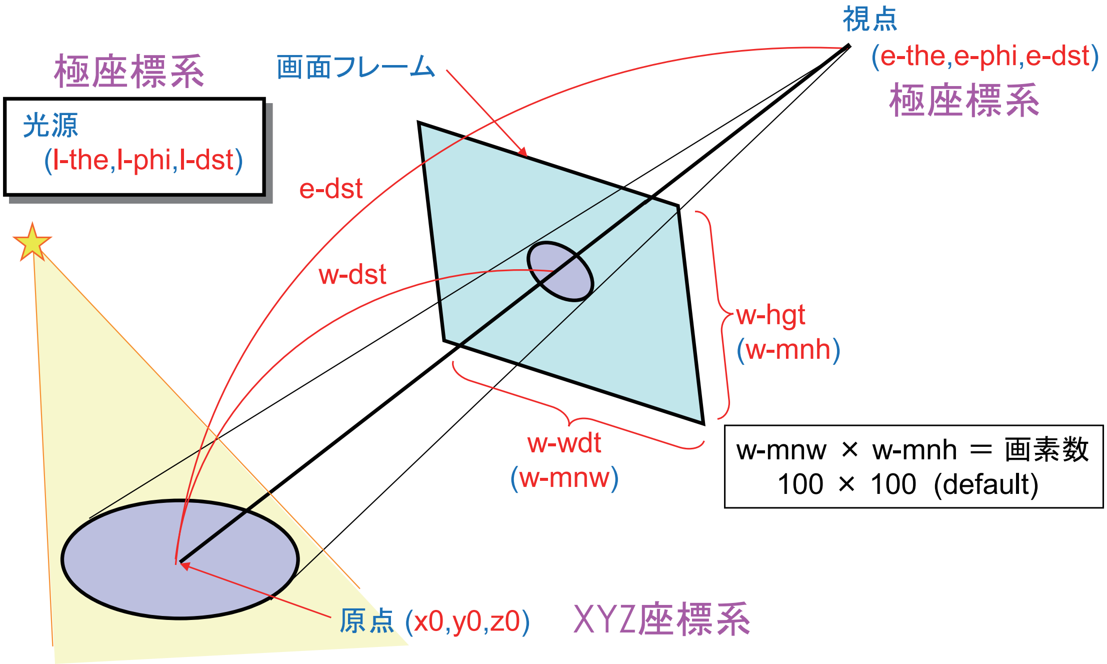


図 66: 3dshow タリー: 原点 (`x0,y0,z0`)、視点 (`e-the,e-phi,e-dst`)、光源 (`l-the,l-phi,l-dst`) 及び画面フレーム (`w-wdt,w-hgt,w-dst`) の与え方。

表 165: [t-3dshow] パラメータ (2)

name	値	説明
line =	(D=0) 0, 1	output = 1, 3 の時、0: 物質境界 + 面境界、 1: 物質境界 + 面境界 + 領域境界 を描く。
r-out =	(D=50000)	視点、光源を含む外部ボイドの半径 (cm)
shadow =	(D=0)	影の強さ、0 で影無し。2 ぐらいが適当。
bright = dark =	(D=0.8) (D=0.2)	面の明るさの制限、1 で最大、0 で明るさ無し 面の暗さの制限、1 で暗さ無し、0 で最大
box = box	(D=0) 10 個の数字	透過 box の数、最大 5 個まで。 透過 box の数だけ繰り返す
matinbox = (次行)	(省略可) all, 数 2 5 8	box 内で表示する material を指定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 数を指定した場合、その数だけの material を次の行に記 述する。 負の数は不可。 material 番号
reginbox =	(省略可) all, 領域番号	box 内で表示する領域を指定する。複数定義可。 all : デフォルト、この場合省略した場合と同じ。 matinbox の指定のある物質では、非表示にする
resol = width =	1(省略時) 0.5(省略時)	境界を求める分解能を各辺 resol 倍します。 域境界を表示する線の太さを定義します。
file =	file name	axis の数だけ定義する
title =	(省略可)	出力ファイルヘッダーのタイトル
angel =	(省略可)	angel パラメータ
sangel =	(省略可)	sangel パラメータ
x-txt = y-txt = z-txt =	(省略可) (省略可) (省略可)	x-軸テキスト y-軸テキスト z-軸テキスト
epsout =	0(省略時), 1	1 で出力ファイルを ANGEL で処理した eps ファイル を作成。ファイル名は出力ファイルの拡張子を eps に 変えたファイル名
axishow =	(D=1) 0, 1, 2	0:xyz 軸を表示しない。 1:xyz 軸を左下に表示する。 2:xyz 軸を中心に表示する。

reg = また、reginbox = の表式は、サブセクション 6.1.1 の領域メッシュの指定の仕方と同じです。

計算時間短縮のため、光源、視点が外部ボイドの内側に納まるように、半径 r-out の球を新たに設定してその外側を外部ボイド、その内側を内部ボイドに設定しています。r-out のデフォルトの値が小さい場合は、つまり、幾何形状がこの値より大きい場合、もしくは、光源、視点を非常に遠くに置きたい時など、r-out の値を大きく設定して下さい。この新しく加えられた外部ボイドを定義する球の設定は、インプットエコーで見ることができます。従って、この icntl = 11 の計算のインプットエコーの出力を再び計算のインプットに用いますとエラーします。視点と光源を一致させれば、影はできません。

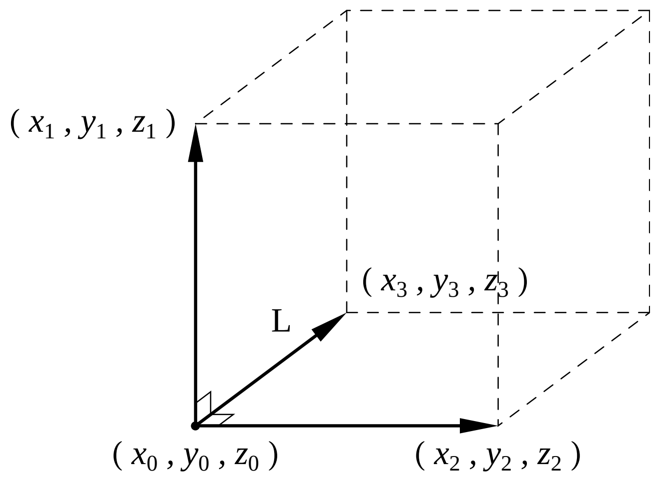
7.22.1 box の定義

透過 box は、最大 5 個まで定義できます。これらの box で定義された内部は、光は透過します。box の定義は、空間内の 3 点、$\mathbf{b}_0(x_0,y_0,z_0)$, $\mathbf{b}_1(x_1,y_1,z_1)$, $\mathbf{b}_2(x_2,y_2,z_2)$ を与えて、この 3 点で定義される平面に垂直な方向、つまり、$(\mathbf{b}_1 - \mathbf{b}_0) \times (\mathbf{b}_2 - \mathbf{b}_0)$ の方向に点 $\mathbf{b}_3$ を点 $\mathbf{b}_0$ から長さ L のところに取ります。この 4 点で下図のように box を定義します。この box の定義には、座標変換が使えます。box の各点を定義する前に trcl=で座標変換番号を定義するか、trcl=(·········) のように座標変換を直接定義します。

ただし、透過 box で定義した空間に真空 (void) があると適切に機能しない場合があるようです。そのときは、密度を薄くした空気等の物質でその領域を満たしてください。

box の定義の書式は、

```
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
```



7.22.2 3dshow の例題

3dshow の例題を見てみましょう。以下の例題の幾何形状は、以下のようなものです。

例 66: [t-3dshow] の例題 (1)

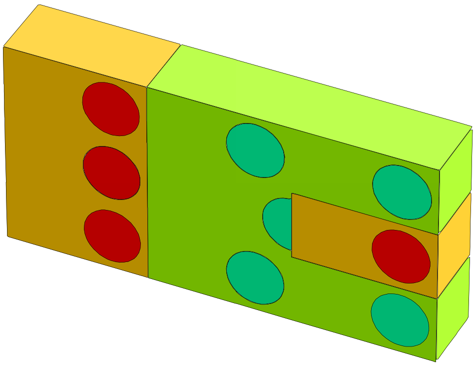
```
[cell]
   1  0 -1 fill=1
   2  0 -41 42 -43 44 -45 46  u=1 fill=5
  22  0 -41 42 -43 44 -45 46  u=1 trcl=(0 0 20) fill=6
  23  like 22 but trcl=(0 0 40) fill = 7
   5  0 -21 22 -23 24 -25 26  u=5 lat=1 fill=3
   6  0 -21 22 -23 24 -25 26  u=6 lat=1 fill= -1:1 0:0 0:0  2 2(0 0 5) 2
   7  0 -21 22 -23 24 -25 26  u=7  fill= -1:1 0:0 0:0  2 3 2 lat=1
   3  1  3.97300E-02  3 u=2
   4  6  4.18280E-02 -3 u=2
  13  5  8.47130E-04 -3 u=3
  14  3  1.23620E-01  3 u=3
  8  -1 +1
[surface]
   1   rpp -15 15 -5 5  -5 55
   21   px    5
   22   px   -5
   23   py    5
   24   py   -5
   25   pz   15
   26   pz   -5
   41   px   15
   42   px  -15
   43   py    5
   44   py   -5
   45   pz   15
   46   pz   -5
   5   rpp -20 20 -5 5 -5 35
   6   rpp -20 20 -5 5 -5 15
   7   rpp -20 20 -5 5 35 55
   3   c/y   0 10  4
```

全体は直方体、中にやはり直方体の lattice とその中に円柱が入っています。これを表示する 3dshow のインプットは次のようになります。

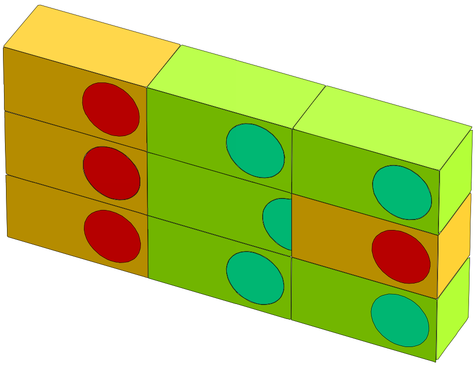
例 67: [t-3dshow] の例題 (2)

```
[t-3dshow]
   output = 3
   heaven = x
   resol = 2
   width = 0.1
   x0 =  0
   y0 =  0
   z0 = 25
   e-the = 70
   e-phi = 50
   e-dst = 1000
   l-the = 50
   l-phi = 25
   l-dst = 2000
   w-wdt = 60
   w-hgt = 40
   w-dst = 150
   file = dshow.dat
```

この結果は、



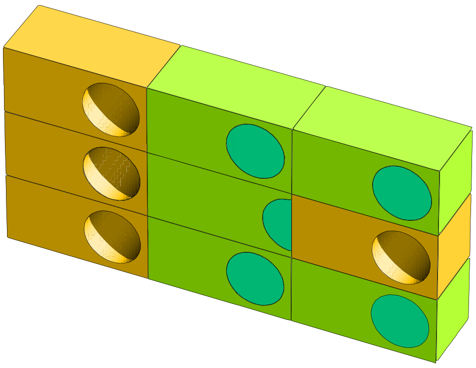
となります。これに、**line = 1** を加えて、領域境界の線も書き込んでみます。



lattice がどう組まれているかが分かると思います。次に、赤の領域、物質 5 番を透明にするために、また、影を付けるために、

```
material = -1
            5
  shadow =  2
```

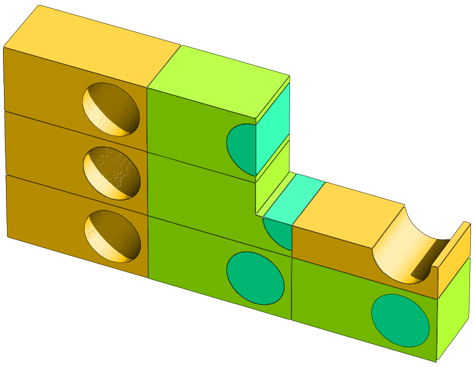
を加えます。



次に、box を設定します。

```
box = 1
box     0   10   30
      100   10   30
        0   10   100 100
```

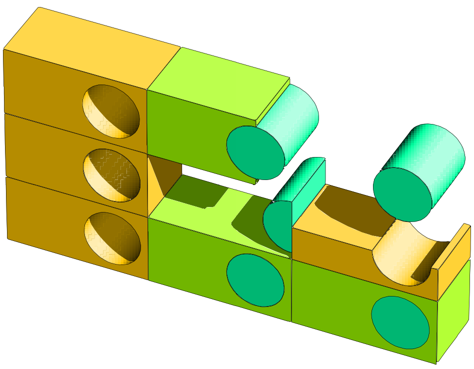
box により、一部が透明になり内部が見られます。



次に、

```
reg = ( 3 < 6[0 0 0] )
matinbox = 1
           6
```

を加え、box 外では、`reg = ( 3 < 6[0 0 0] )` を透明にし、box 内では、物質 6 番を見える様にします。



これらの組み合わせで、複雑な構造の内部を希望どおり表示することが可能になります。

8 dump ファイルの処理

[t-cross], [t-time], [t-product] タリーで、粒子の情報をファイルにダンプすることができます。このダンプファイルをソースとして新たな継続計算ができます。また、ダンプファイルの情報を処理することにより、既存のタリーにはない物理量の集計を取ることができます。そのためには、ユーザーがダンプファイルを読み込んで処理するプログラムを書かなければなりません。そのプログラムの雛形として、バイナリーダンプファイルをアスキーへ、アスキーダンプファイルをバイナリーへ変換するプログラムを添付しました。プログラムは、dump-a.f 、Windows の実行ファイルが dump_a.exe です。

このプログラムを書き換えることで、ダンプファイルを処理していろいろな物理量を求めるユーザープログラムを作ることができます。その助けのために、ソースリストと解説を以下に示します。

ファイル 3: dump-a.f

```
************************************************************************
*                                                                      *
*     This program exchanges the binary data and the ascii data        *
*     of dump file.                                                    *
*                                                                      *
*       modified by K.Niita on 2005/08/15                              *
*                                                                      *
*                                                                      *
*                                                                      *
*                                                                      *
************************************************************************
      implicit real*8 (a-h,o-z)
*-----------------------------------------------------------------------
      dimension isdmp(0:30)
      dimension jsdmp(0:30)
      data isdmp / 31*0 /
      data jsdmp / 31*0 /
      character chin*80
      character chot*80
      logical exex
      character dmpc(30)*4
      data dmpc / '  kf','   x','   y','   z','   u','   v','   w',
     &            '   e','  wt','  tm','  c1','  c2','  c3',
     &            '  sx','  sy','  sz','  n0','  nc','  nb','  no',
     &            '    ','    ','    ','    ','    ','    ',
     &            '    ','    ','    ','    '/
      dimension dmpd(30)
      dimension dmpp(30)
      data dmpp /  2112.,   0.0,   0.0,   0.0,   0.0,   0.0,   1.0,
     &              100.,   1.0,   0.0,   0.0,   0.0,   0.0,
     &               0.0,   0.0,   0.0,   0.0,   1.0,   1.0,   1.0,
     &               0.0,   0.0,   0.0,   0.0,   0.0,   0.0,   0.0,
     &               0.0,   0.0,   0.0/
*-----------------------------------------------------------------------
         in = 5
         io = 6
         id = 20
         ia = 21
         iserr = 0
*-----------------------------------------------------------------------
*        user program frag : 0 => no,  1 => with user program
*-----------------------------------------------------------------------
         iuser = 0
*-----------------------------------------------------------------------
*        read ascii or binary frag
*-----------------------------------------------------------------------
         write(io,*) ' ** 0 => read binary to ascii'
         write(io,*) ' ** 1 => read ascii  to binary'
         read(in,*,end=993) iasb
*-----------------------------------------------------------------------
*        read the name of input dump file
*-----------------------------------------------------------------------
         write(io,*)
```

```
         write(io,*) ' ** put the file name of input dump file'
         read(in,'(a80)',end=998) chin
            inquire( file = chin, exist = exex )
            if( exex .eqv. .false. ) then
              write(io,*) ' ** Error : the file does not exist'
              goto 999
            end if
            if( iasb .eq. 0 ) then
              open(id, file = chin,
     &             form='unformatted',status = 'old' )
            else
              open(id, file = chin,
     &             form='formatted',status = 'old' )
            end if
*-----------------------------------------------------------------------
*        read the number of data and data sequence
*-----------------------------------------------------------------------
         write(io,*)
         write(io,*) ' ** put the number of data in a record'
         read(in,*,end=997) isdmp(0)
         write(io,*)
         write(io,*) ' ** put the ID numbers of data in a record'
         read(in,*,end=996) ( isdmp(i), i = 1, isdmp(0) )
            do k = 1, isdmp(0)
               if( isdmp(k) .gt. 20 .or.
     &             isdmp(k) .le.  0 ) goto 992
                  jsdmp(isdmp(k)) = k
            end do
            write(io,*)
            write(io,'('' # dump data : '',30(a4))')
     &         ( dmpc(isdmp(j)), j = 1, isdmp(0) )
*-----------------------------------------------------------------------
*        read the name of output dump file
*-----------------------------------------------------------------------
         write(io,*)
         write(io,*) ' ** put the file name of output'
         read(in,'(a80)',end=998) chot
            inquire( file = chot, exist = exex )
            if( exex .eqv. .true. ) then
              write(io,*)
              write(io,*) ' ** Warning : the file already exists'
              write(io,*) ' ** Do you want to overwrite ?'
              write(io,*) ' ** Yes <= 0, No <= 1'
              read(in,*,end=995) iyes
              if( iyes .ne. 0 ) goto 999
            end if
            if( iasb .eq. 0 .or. iuser .ne. 0 ) then
              open(ia, file = chot,
     &             form='formatted',status = 'unknown' )
            else
              open(ia, file = chot,
     &             form='unformatted',status = 'unknown' )
            end if
*-----------------------------------------------------------------------
*        read the number of records to read
*-----------------------------------------------------------------------
         write(io,*)
         write(io,*) ' ** put the number of records to read'
         write(io,*) ' ** all <= 0, or positive integer'
         read(in,*,end=994) irec
*-----------------------------------------------------------------------
*        start reading the data
*-----------------------------------------------------------------------
         write(io,*)
         write(io,*) ' ** start read and write the data'
*-----------------------------------------------------------------------
         jrec = 0
  100    jrec = jrec + 1
         if( irec .gt. 0 .and. jrec .gt. irec ) goto 500
  687       continue
            if( iasb .eq. 0 ) then
               read(id,end=688,err=690)
```

```
     &         ( dmpd(isdmp(k)), k = 1, isdmp(0) )
            else
               read(id,'(30(1p1d24.15))',end=688,err=690)
     &         ( dmpd(isdmp(k)), k = 1, isdmp(0) )
            end if
            goto 689
  688       if( irec .gt. 0 ) then
               rewind id
               goto 687
            else
               goto 500
            end if
  690       continue
            iserr = iserr + 1
            write(io,'(''  ** Error in dump file no ='',i5)') iserr
            goto 687
  689       continue
*-----------------------------------------------------------------------
*     user program here
*-----------------------------------------------------------------------
      if( iuser .ne. 0 ) then
            do k = 1, 20
               if( jsdmp(k)  .gt. 0 ) dmpp(k) = dmpd(k)
            end do
               kf = nint( dmpp(1) )
               x  = dmpp(2)
               y  = dmpp(3)
               z  = dmpp(4)
               u  = dmpp(5)
               v  = dmpp(6)
               w  = dmpp(7)
               e  = dmpp(8)
               wt = dmpp(9)
               t  = dmpp(10)
               n1 = nint( dmpp(11) )
               n2 = nint( dmpp(12) )
               n3 = nint( dmpp(13) )
               sx = dmpp(14)
               sy = dmpp(15)
               sz = dmpp(16)
               n0 = nint( dmpp(17) )
               nc = nint( dmpp(18) )
               nb = nint( dmpp(19) )
               no = nint( dmpp(20) )
      end if
*-----------------------------------------------------------------------
*     write data on the file
*-----------------------------------------------------------------------
      if( iuser .eq. 0 ) then
            if( iasb .eq. 0 ) then
               write(ia,'(30(1p1d24.15))')
     &         ( dmpd(isdmp(k)), k = 1, isdmp(0) )
            else
               write(ia)
     &         ( dmpd(isdmp(k)), k = 1, isdmp(0) )
            end if
      end if
*-----------------------------------------------------------------------
         goto 100
*-----------------------------------------------------------------------
*        end of process
*-----------------------------------------------------------------------
  500    continue
         write(io,*)  ' ** end of read and write the data'
         write(io,'(''  ** number of processed records is '',
     &         i8)') jrec-1
         write(io,*)
         close( id )
         close( ia )
         goto 999
*-----------------------------------------------------------------------
  992 continue
```

```
      write(io,*) ' ** Error : ID should be 1 - 20'
      goto 999
  993 continue
      write(io,*) ' ** Error : the ascii or binary frag is wrong'
      goto 999
  994 continue
      write(io,*) ' ** Error : the number of records is wrong'
      goto 999
  995 continue
      write(io,*) ' ** Error : the answer should be 0 or 1'
      goto 999
  996 continue
      write(io,*) ' ** Error : the ID numbers is wrong'
      goto 999
  997 continue
      write(io,*) ' ** Error : the number of data is wrong'
      goto 999
  998 continue
      write(io,*) ' ** Error : file name is wrong'
      goto 999
  999 continue
      stop
      end
```

このプログラムは、アスキーもしくはバイナリーのダンプファイルを読み込んで、バイナリーもしくはアスキーのダンプファイルに書き換えるプログラムになっています。

入力パラメーターは、標準入力から読み込みます。通常は、入力を促すメッセージの後にパラメーターを入力する会話形式で入力します。まず最初が入力ダンプファイルがバイナリーかアスキーかのフラグです。

```
 ** 0 => read binary to ascii
 ** 1 => read ascii  to binary
```

バイナリーの時は 0、アスキーのときは 1 を入力します。次に、入力ダンプファイル名を聞いてきます。

```
 ** put the file name of input dump file
```

入力ダンプファイル名をを打ち込みます。次に、

```
 ** put the number of data in a record
```

のように、ひとつのレコードの dump データの個数を聞いてきます。バイナリー、アスキー両者ともここでは正の数で答えます。

```
 ** put the ID numbers of data in a record
```

ここで、dump データの個数分のデータの ID を打ち込みます。dump データの種類と ID 番号は、表 97, 98 にあります。

```
 ** put the file name of output
```

ここに、出力のファイル名を打ち込みます。もし、そのファイルが存在する場合には、上書きするかどうか聞いてきます。次に、

```
** put the number of records to read
** all <= 0, or positive integer
```

のように、読み込むデータのレコード数を聞いてきます。0 を打ち込むと全てのデータ、正の数を入れるとその個数だけ処理します。もしこの数がデータファイルのレコード数より多い場合は、データの最初に戻って処理を続けます。この後、データを読み込み、出力ファイルに書き込んで終了します。最後に、実際に何個のレコードを処理したかが打ち出されます。

このプログラムをもとにユーザープログラムを書くには、35 行目の iuser を 1 にします。そうすることによって、出力ファイルへの変換データの書き出しはしません。また、出力ファイルはアスキーとなります。次に、150 行目から 169 行までを見てください。ここに、読み込んだデータを、kf, x, y, z, u, v, w, e, wt, t, n1, n2, n3, sx, sy, sz, n0, nc, nb, no という変数で定義しています。ここで、kf は、粒子を識別する kf-code、表 4 を参照してください。x, y, z は、座標 (cm)、u, v, w は、運動量の単位ベクトル、e は、エネルギー (MeV)、原子核の場合は核子当たりのエネルギー、wt は粒子ウエイト、time は時間 (nsec)、n1, n2, n3 は、カウンターの値です。入力データにない変数は、29-33 行目で定義されているデフォルト値が入ります。必要ならここを書き換えてください。この部分に統計処理をするプログラム、また、最後に出力のための write 文を書きます。

9 出力中性子、光子データフォーマット

中性子、光子、電子、陽電子、陽子に関しては、計算打切エネルギー以下の情報を、MCNP, EGS4 などの他のコードで接続計算をするために、ファイル (file(12), file(13), file(10)) に書き出すことができます。ファイルにバイナリーで書き出す書式を以下に示します。これらは、中性子、光子、電子、陽電子、陽子とも同じです。

データ書式は、基本的に次の様な形になっています。

```
rd, rn, ( data(i), i = 1, nint(abs(rd)) )
rd, rn, ( data(i), i = 1, nint(abs(rd)) )
................
................
```

これらは、それぞれの場合に具体的に次の様になっています。
まず、incut = 1 でインポータンスを含まない場合 (rd<0)、

```
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), i = 1, n )
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), i = 1, n )
................
................
```

次に、incut = 1 でインポータンスを含む場合 (rd>0)、

```
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), wt(i), i = 1, n )
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), wt(i), i = 1, n )
................
................
```

次に、incut = 2 でインポータンスを含まない場合 (rd<0)、

```
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), t(i), i = 1, n )
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), t(i), i = 1, n )
................
................
```

次に、incut = 2 でインポータンスを含む場合 (rd>0)、

```
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), wt(i), t(i), i = 1, n )
rd, rn, x, y, z, ( e(i), u(i), v(i), w(i), wt(i), t(i), i = 1, n )
................
................
```

上記の記述で、n = nint(rn)、x, y, z は、座標 (cm)、e(i) は、エネルギー (MeV)、u(i), v(i), w(i) は、運動量の単位ベクトル、wt(i) は、粒子のウエイト、t(i) は、時間 (ns) です。

igcut = 3 の場合は、上の incut = 2 の場合の t(i) の変わりに、粒子の判別子 p(i) が入ります。

p(i) = 3.0 が光子、p(i) = 4.0 が電子、p(i) = 5.0 が陽電子です。

10 領域エラーチェック

3 次元体系を構築する際、ある場所を 2 つ以上の領域で定義してしまう多重定義や、1 つも領域を設定しない未定義領域が生じる可能性があります。これらの領域エラーが発生した状態では、輸送している粒子の現在位置がわからなくなってしまう場合があり、適切な輸送計算を実行することができません。領域エラーチェックは、構築した体系内に 2 重定義や未定義領域がないか確認するための機能です。この機能を利用して領域エラーのない 3 次元体系を作成してください。

領域エラーチェック機能は、ジオメトリを 2 次元表示するタリーで自動的に実行されます。具体的には `[t-gshow]`, `[t-rshow]` タリー、及び `icntl=8 or 10` の場合に `axis = xy, yz, or xz` として gshow or rshow オプションを指定したタリーが該当します。領域エラーチェック機能をコントロールするパラメータは各タリーの `ginfo` です。`ginfo=0, 1, 2` が指定でき、それぞれ、チェックしない、2 次元ジオメトリ図上にエラー領域を表示する（図 67 参照）、エラー領域表示に加えて後述の領域エラーファイル出力する、となります。ジオメトリを表示するタリーに対しては、自動的に `ginfo=2` となります。また、飛跡長などを 2 次元表示する通常のタリー（`icntl = 0` かつ `axis = xy, yz, or xz`）でも、`ginfo=1 or 2` と指定すれば領域エラーチェックを行います。

領域エラーがある場合、図 67 のような出力結果が得られます。多重定義領域がある場合はその領域を黒色で、未定義領域がある場合はその領域を紫色で表示します。ただし、未定義領域がある場合に、その周りの定義された領域が消えてしまう場合があります。

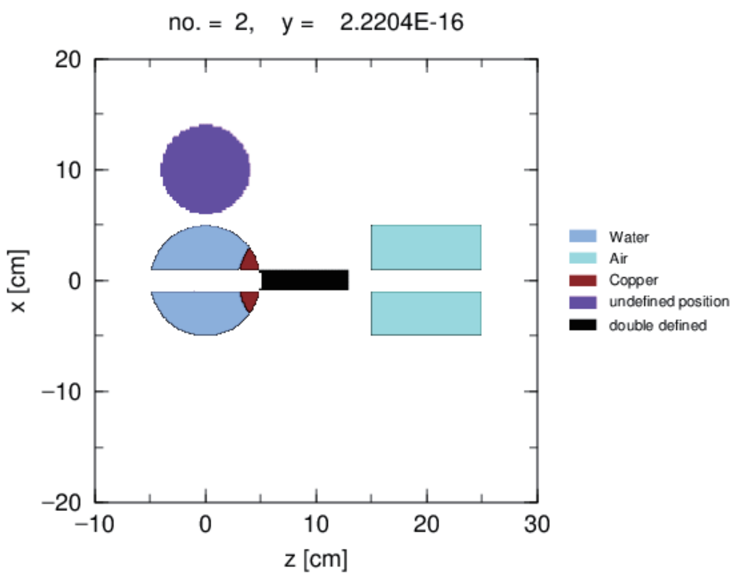


図 67: 領域エラーがある場合の出力結果

`ginfo=2` の場合、`file=`で指定したファイル名の拡張子を “.err” に変えた領域エラーファイルを作成します。このファイルには、エラーが起こっている領域の (x,y,z) 座標が次のような形式で出力されます。

```
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
```

1 行目のエラーは、セル 100 と 102 が x = -4.847761E+00, y = 1.234568E-11, z = -1.211940E+00 で重なって

いることを示しています。2 行目のエラーは、 x = -4.241791E+00, y = -2.500000E+00, z = -8.079602E-01 の場所が未定義領域であることを示しています。

この領域エラーチェック機能を用いれば、複雑な 3 次元体系を作成する場合でも、効率的にエラー領域を見つけることができます。ただしこの機能は、タリー領域で指定している範囲しかチェックしませんのでご注意ください。また、範囲内であっても、xyz メッシュで区切った格子点しかチェックしませんので、小さな領域エラーがある場合は見逃してしまう可能性がありますので、ご注意ください。

11 コンパイル

PHITS で大容量のボクセルファントムを使用するためにメモリ枠を拡張する場合や、“usrsors.f” ファイルを使って線源を定義する場合（5.3.18 参照）は、PHITS をコンパイルする必要があります。

PHITS 事務局で推奨しているコンパイラは Intel Fortran Compiler（11.1 以降）[103] と gfortran（4.8 と 7 シリーズ以降）です[104]。これら以外のコンパイラによる不具合に関しては十分なサポートができないことをご了承ください。もし、“makefile” に書かれていないオプションでコンパイル&実行に成功した方は、PHITS 事務局までご連絡ください。

表 166 に、各 OS においてコンパイル可能なコンパイラと並列計算に関する設定の組み合わせをまとめました。可能な組み合わせには ✓ をつけています。表にある single は並列計算を行わない場合を表しており、MPI と Open MP はそれぞれ MPI プロトコルと Open MP を利用した並列計算、hybrid はそれらを組み合わせた並列計算の設定を表しています。並列計算については、12 節をご覧ください。

表 166: 各 OS におけるコンパイル可能なコンパイラと設定の組み合わせ

OS	Intel Fortran				gfortran			
	single	MPI	Open MP	hybrid	single	MPI	Open MP	hybrid
Windows	✓	✓	✓	✓	✓	✓		
Mac	✓		✓		✓			
Linux	✓	✓	✓	✓	✓	✓		

11.1 Windows OS におけるコンパイル方法

11.1.1 Makefile を利用したコンパイル

“/src/” フォルダ内にある “makefile” を使って、PHITS をコンパイルすることができます。そのためには、“makefile” を各自の環境に合わせて書き換える必要があります。具体的には、このファイルの最初の辺りにある `ENVFLAGS` に計算環境に応じた変数をセットしてください（例えば、gfortran compiler を使用する場合は `WinGfort`）。作成される実行ファイル名は、phits_XXX.exe（XXX は `ENVFLAGS` にセットした変数）のようになります。例えば、`ENVFLAGS=WinGfort` の場合は、phits_WinGfort.exe が実行ファイル名です。“makefile” 中のコンパイラオプションなどは一例ですので、場合によってはそれらを変更することによりうまくコンパイルできる可能性があります。なお、付属の “makefile” が想定しているのは GNU make です。もし、“make” コマンドで動作しない場合は、“gmake” をお試しください。

Windows 用コンパイラとして gfortran をインストールする場合は、

- MinGW-w64 - for 32 and 64 bit Windows
  (`https://sourceforge.net/projects/mingw-w64/files/?source=navbar`)

にアクセスし、左上にある “Download Latest Version” をクリックしてインストーラーをダウンロードしてください。ダウンロードしたインストーラー（exe ファイル）を実行するとインストールが始まりますので、“Version” は最新版（もしくはそれ以外の適合 version）を選択し[105]、“Architecture” は 64bit であれば

[103] PHITS は Intel Fortran でコンパイルした方が高速化されますので，可能であれば Intel Fortran の利用をお勧めいたします。Intel Fortran は，学生であれば無償で利用可能です（`https://software.intel.com/en-us/qualify-for-free-software/student` 参照）。
[104] バージョン 2.50 以降、PHITS 内での大部分のメモリ使用方法を “static” から “dynamic” に変更したため、f77 や g77 ではコンパイルできないようになりました。
[105] gfortran のバージョン 4.9-5.4 で、適切にコンパイルされない場合があります。

“x86_64”, 32bit であれば “i686” を選択しながら進めてください。これら以外は変更する必要はありません。最後は “next” を押すとインストールが完了します[106]。次に、PHITS をコンパイルする場合は以下の手順で行ってください。

(1) インストールしたフォルダにある “mingw-w64.bat” をダブルクリックしてコマンドプロンプトを開く。

(2) cd コマンドなどにより “/phits/src/” フォルダに移動して、

```
mingw32-make
```

と、コマンドを入力するとコンパイルが始まります。ただしその際、あらかじめ “makefile” を編集し、`ENVFLAGS=WinGfort` と変更しておいてください。また、MPI や OpenMP は利用しない設定としてください。

作成した実行ファイル (phits_WinGfort.exe) を使用するには、gfortran をインストールしたフォルダに PATH を通す必要があります。最も簡単な方法は “phits.bat” を使う方法です。その場合は、テキストエディタで “mingw-w64.bat” を開き、その 2 行目にある

```
set PATH=C:\Program Files\mingw-64\...
```

の行を “phits.bat” の 2 行目にコピー＆ペーストし、`PHITS_Single_EXE` パラメータの箇所を作成した実行ファイル名（例えば、c:/phits/phits_WinGfort.exe）に変更します。以上により、従来と同じ方法（送る →PHITS、もしくは TeraPad から PHITS を実行）で PHITS を実行することができます。その際、OpenMP は使えませんのでご注意ください。また、PHITS 実行後に “Note: The following floating-point exceptions are signaling: IEEE_DENORMAL” のような警告が出る場合がありますが、無視しても問題ありません。

11.1.2 Microsoft Visual Studio と Intel Fortran を利用したコンパイル (Windows 用)

“/phits/bin/” フォルダ内に、Microsoft Visual Studio と Intel Fortran を組み合わせた環境に対するプロジェクトファイル (“phits-intel.proj”) が含まれています。以下の手順に従うことにより、このファイルを使って PHITS をコンパイルが実行できます。

(1) `phits-intel.proj` をダブルクリック (Visual Studio や Intel Fortran のバージョンが新しい場合は自動更新する場合があります。また、Visual Studio 2005 以前や Intel Fortran 11.1 より前のバージョンではうまく開けない場合があります)

(2) リリースモードで `phits-intel.proj` をビルドする（1 度で成功しない場合は何度か試してみてください。）

(3) `bin` フォルダ内に PHITS のインプットファイルを作成

(4) リリースモードで実行

(5) 起動した実行画面で `file=PHITS` インプットファイル名と入力

(6) `xz_track_all.eps` が作成されたことを確認

メモリ共有型並列計算用の実行ファイルを作成する場合は、ビルドする前に “/Qopenmp” オプションを追加してください。（Microsoft Visual Studio ウインドーの上部メニューから “プロジェクト” →“phits-intel のプロパティ” を選択肢、“Fortran” →“コマンドライン” の “追加のオプション” に追加します。）

もし、作成した実行ファイルを “送る” 機能で使用する場合は、“/phits/bin/” フォルダにある “phits.bat” をテキストエディタで開き、次のように書かれている環境変数の `PHITS_XXX_EXE` を書き換えてください（XXX には single, MPI, OpenMP が入る）。

[106] なお、このパッケージをインストールした場合、“make” コマンドとして “mingw32-make” が利用できるようになります。

```
set PHITS_Single_EXE="c:/phits/bin/phits308_win.exe"
set PHITS_MPI_EXE="c:/phits/bin/phits308_win_mpi.exe"
set PHITS_OpenMP_EXE="c:/phits/bin/phits308_win_openmp.exe"
```

Single, MPI, OpenMP は、それぞれ並列計算を行わない場合、MPI を利用したメモリ分散型並列計算、Open MP を利用したメモリ共有型並列計算を意味しており、作成した実行ファイルの設定に対応した変数を書き換えてください。実行ファイル名の方を “phits.bat” に書かれている名称（例えば、phits308_win.exe）に変更しても動作しますが、PHITS の更新を行う際に必要となりますので、元のファイルを消さないようにご注意ください。

11.2 Mac OS におけるコンパイル方法

11.2.1 Makefile を利用したコンパイル

“/src/” フォルダ内にある “makefile” を使って、PHITS をコンパイルすることができます。そのためには、“makefile” を各自の環境に合わせて書き換える必要があります。具体的には、このファイルの最初の辺りにある `ENVFLAGS` に計算環境に応じた変数をセットしてください（例えば、Intel Fortran compiler を使用する場合は `MacIfort`）。また、メモリ共有型並列を使用する場合は、`USEOMP` のある行の “#” を消して、これらを有効にしてください。作成される実行ファイル名は、phits_XXX.exe（XXX は `ENVFLAGS` にセットした変数）のようになります。例えば、`ENVFLAGS=MacIfort` の場合は、phits_MacIfort.exe が実行ファイル名です。更に、`USEOMP` を有効にした場合は phits_MacIfort_OMP.exe のようになります。“makefile” 中のコンパイラオプションなどは一例ですので、場合によってはそれらを変更することによりうまくコンパイルできる可能性があります。なお、付属の “makefile” が想定しているのは GNU make です。もし、“make” コマンドで動作しない場合は、“gmake” をお試しください。

11.3 Linux OS におけるコンパイル方法

11.3.1 Makefile を利用したコンパイル

“/src/” フォルダ内にある “makefile” を使って、PHITS をコンパイルすることができます。そのためには、“makefile” を各自の環境に合わせて書き換える必要があります。具体的には、このファイルの最初の辺りにある `ENVFLAGS` に計算環境に応じた変数をセットしてください（例えば、Linux で Intel Fortran compiler を使用する場合は `LinIfort`）。また、メモリ分散型並列及びメモリ共有型並列を使用する場合は、`USEMPI` や `USEOMP` のある行の “#” を消して、これらを有効にしてください。作成される実行ファイル名は、phits_XXX.exe（XXX は `ENVFLAGS` にセットした変数）のようになります。実行ファイルは “/src/” フォルダの親フォルダである “/phits/” フォルダに作られます。例えば、`ENVFLAGS=LinIfort` の場合は、phits_LinIfort が実行ファイル名です。更に、`USEOMP` を有効にした場合は phits_LinIfort_OMP のようになります。“makefile” 中のコンパイラオプションなどは一例ですので、場合によってはそれらを変更することによりうまくコンパイルできる可能性があります。なお、付属の “makefile” が想定しているのは GNU make です。もし、“make” コマンドで動作しない場合は、“gmake” をお試しください。

11.4 KURBUC を含めたコンパイル

陽子及びイオンに対する飛跡構造解析コード KURBUC のソースコード “kurbuc.f” は、著作権の関係上 “src” フォルダに含まれておりません。パッケージに含まれる “kurbuc.f” は、本物の “kurbuc.f” がなくてもコンパイルができるように作成したダミーファイルです。したがって、上記手順でそのままコンパイルすると、陽子やイオンに対する飛跡構造解析を行うことができない PHITS 実行ファイルが作成されます。

各自で作成した実行ファイルで陽子やイオンに対する飛跡構造解析を行いたい場合は、まずは通常のコンパイルを実施した後、作成された kurbuc 用オブジェクトファイル (“kurbuc.o” や “kurbuc.obj” など) を “src/kurbuc-obj” フォルダに含まれる各自の環境に合ったオブジェクトファイルに置き換えて、実行ファイルを再作成してください。通常のコンパイル方法で実行ファイルが再作成されない場合は、例えば “main.f” を再保存してタイムスタンプを更新してから再コンパイルしてください。各オブジェクトファイルのコンパイル環境は、“src/kurbuc-obj/readme.txt” をご参照ください。各自の環境に対応するオブジェクトファイルがない場合や、付属のオブジェクトファイルでは正しく動作しない場合は、PHITS 事務局までご連絡ください。

12 並列版のための指定方法

並列計算には、メモリ分散型並列とメモリ共有型並列計算があり、PHITS ではその両方に対応しています。また、両者を組み合わせたハイブリッド並列も可能です。メモリ分散型並列のためには、お使いのコンピュータに MPI をインストールする必要があります。メモリ共有型並列は、特にプロトコルなどをインストールする必要はありません。ただし、同じ数の CPU を使った場合はメモリ分散型並列の方が計算時間が短くなる場合が多いです。シングル・並列計算の切り替えはコンパイラオプションで行い、それぞれ別の実行ファイルを作る必要があります。詳しくは 11 節をご参照ください。なお、メモリ分散型並列（MPICH2 を利用）の場合は、Intel Fortran Compiler （11.1 以降）か gfortran（4.8 と 7 シリーズ以降）のコンパイラで実行ファイルを作成可能です。一方、メモリ共有型並列の場合は、Intel Fortran Compiler のみ作成可能で、gfortran は未対応ですのでご注意ください。

メモリ分散型並列では、各 CPU コアにそれぞれバッチ単位で Job を渡して、全てのコアがそのバッチ計算を終了した時点でメインのコアが結果を集計します。全てのコアがジオメトリやタリーの情報をそれぞれ読み込むため、メモリは、シングルで走らせるときのコア数倍必要となります。したがって、ボクセルファントムなど、巨大なメモリを必要とする計算には不向きです。また、全てのコアが計算終了するまで集計を待つため、バッチ当たりのヒストリー数 (`maxcas`) が少ないなど、各コア間での計算時間にばらつきが大きい場合は、計算時間が不必要に長くなってしまいます。

メモリ共有型並列では、各コアにヒストリー単位で Job を渡して、各コアは、ジオメトリやタリーなどの情報を共有しながら計算します。したがって、メモリの使用量は、シングルで走らせる場合と大差はありません。しかし、メモリへの書き込みアクセスが競合するため、タリー結果などを頻繁に書き込む計算（`[t-sed]` を使った計算など）では、計算時間が不必要に長くなってしまいます。

12.1 メモリ分散型並列

12.1.1 セットアップ方法

メモリ分散型 (MPI) 並列を実行するためには，MPI プロトコルをインストールする必要があります。ここでは，1 台の Windows PC を用いて並列計算を行う場合の MPI プロトコルインストール方法について解説します。複数台の Windows PC を接続して並列計算を行う場合は，“/phits/document/mpi/” フォルダに含まれる “Windows-MPI-setup-jp.docx” をご参照下さい。また，Mac 及び Linux に関しては，インターネットなどを参照して各自で MPI プロトコルをインストールして下さい。

Windows の場合は，MPI プロトコルとして Argonne National Laboratory が配布しているフリーソフト MPICH2 の利用を奨励します（パッケージに含まれる MPI 版実行ファイルが MPICH2 を利用してコンパイルされているため）。この最新バージョンは Windows へのインストーラが含まれていませんので，`http://www.mpich.org/static/downloads/1.4/` にある “mpich2-1.4-win-x86-64.msi” （64 ビット版）もしくは “mpich2-1.4-win-ia32.msi” （32 ビット版）をダウンロードして下さい。ダウンロードしたら msi ファイルを右クリックしてインストールを選択してください。その際，インストール先フォルダはデフォルト “C:/Program Files/MPICH2/” から変更せず，使用者は “Everyone” にして下さい。インストール時に “Passphrase” を要求されますが，何を入力しても問題ありません。ただし，複数台で並列計算を実行する場合は，共通の Passphrase が必要になりますので，入力した Passphrase を覚えておいてください。MPICH2 のインストールが完了したら，“/phits/document/mpi/” フォルダにある “smpd-install.bat” ファイルを右クリックして，“管理者として実行” としてください。これで設定が完了です。

12.1.2 Windows での実行方法

1 台の Windows PC で MPI 並列版の PHITS を実行するためには，PHITS 入力ファイルの最初のセクションの前に`$MPI=`M（M は並列数）と書きます。例えば CPU コアが 4 つ搭載された PC で最も効率よく計算

したい場合は

```
$MPI = 4
```

とします。なお，実際の並列実行 PE（processing element）数は，コントロール用に 1 つ追加されるため $M + 1$ となります。この状態で入力ファイルを保存し，通常の実行方法と同じく "送る" →"PHITS" とすることにより，MPI 並列で PHITS を実行できます。ただし，初めて実行する際は，Windows のユーザー名とパスワードを聞かれますので，ユーザー名は何も入力せずにそのままリターン（デフォルトで入ります），パスワードには Windows にログインする際のパスワードを入力して下さい。なお，現在のところ OpenMP と組み合わせたハイブリッド並列には対応していないため，OpenMP 用のコマンド`$OMP` と MPI 用のコマンド`$MPI` の両方が入力ファイルに書かれた場合は，後ろに書いた方を優先する仕様になっています。

12.1.3　コマンドラインでの実行方法

Linux や Mac，もしくは複数台の Windows PC で MPI 並列計算を行うためには，コマンドラインを用いる必要があります。その実行方法は，例えば

```
mpirun -np 5 phits_LinIfort_MPI.exe
```

のようになります。ここで "mpirun" はインストールした MPI プロトコルの実行ファイル名で，Windows の場合は "mpiexec.exe" となります。-np の後の数字は並列の実行 PE 数、"phits_LinIfort_MPI.exe" は PHITS の実行ファイル名です。このコマンドを qsub など各コンピュータに合わせた方法で投入してください。また、メモリ分散型並列版では、自動的に `phits.in` ファイルから入力ファイル名を読み込みます。この `phits.in` は固定です。このファイルの 1 行目に

```
file = input_file_name
```

のように入力ファイル名を記述します。従って、起動シェルでインプットファイルのリダイレクトはできませんので注意してください。これは、メモリ分散型並列だけの制約です。シングル版はこれまでの通りです。

なお、`phits.in` の 1 行目に `file=phits.in` と書けば、それ以下の行に通常のインプットファイルを書いて実行することもできます (2.3.2 節参照)。

12.1.4　maxcas, maxbch の指定方法

PHITS のメモリ分散型並列計算では、バッチ単位で並列化しています。従って、入力のバッチ数 `maxbch` は、並列の実行 PE 数 (コントロールに 1PE 使うので、全 PE 数 −1) の整数倍になるように指定して下さい。もし整数倍でないときは、整数倍になるように、またトータルヒストリー数が入力データとほぼ同じになるように、プログラムが自動で変換します。変換した場合は、出力のインプットエコーの最後にコメントが出力されます。後述しますが、バッチ毎の情報はメモリ分散型並列の場合は、バッチ数 × (PE −1) 毎に出力されます。途中停止もこの単位で行うことができます。

なお、再開始計算の場合 (`istdev < 0`) は、`maxcas` を過去の計算結果と一致するよう自動調整するため、全ヒストリー数を合わせる上記調整は行いません。`maxbch` を (PE−1) の整数倍にする調整のみ行います (端数は切り捨てます)。

12.1.5　異常終了の処理

プログラムが異常終了した場合、その異常終了した PE を切り離して実行します。最終的に残った PE の計算結果の合計を全体の結果として出力します。バッチ毎の情報と計算のサマリーの中に PE の状態が出力されます。MCNP などの他の計算コードとの接続計算の時は、異常終了の PE がある場合、ncut ファイルがその PE で不完全ですので、注意が必要です。

12.1.6 PHITS での ncut, gcut, pcut, dumpall ファイルの指定

ncut, gcut, pcut ファイル名は、通常のように入力ファイルで、

```
file(12) = temp/ncut.dat
```

のように指定します。1PE で走らせる場合は、カレントディレクトリの (パスを含む場合はそのディレクトリの)ncut.dat に書きこまれます。2PE 以上の計算では、デフォルトで各ノードの

```
/wk/j9999/temp/ncut.dat
```

に分散して書きこまれます。ここで `j9999` は、ユーザー名です。これは環境変数 `LOGNAME` から自動で読み込んでいます。この環境変数 `LOGNAME` は、デフォルトでユーザー名が入っています。計算の前に各ノードの `/wk` ディレクトリに `j9999` なるディレクトリを作ってください (作り方は並列コマンド `brsh` を使います)。もし、ユーザー名以外のディレクトリに ncut ファイルを作りたいときは、実行前に環境変数 `LOGNAME` を変更して運用して下さい。その時、その名前のディレクトリが、各ノードの`/wk` ディレクトリにあることを確認して下さい。

書きこみファイルのオプションとして、inpara, igpara, ippara があります (デフォルトで 0 です)。これらを 1 にすると、ファイル名の後に IP の番号を付けます。例えば、

```
/wk/j9999/temp/ncut.dat.005
```

これは、IP=5 のファイル名です。

inpara, igpara, ippara を 3 にするとデフォルトでついている ヘッダー

```
/wk/j9999/
```

が付きません。3 は 1 と同じくファイル名の最後に IP の番号を付けます。

12.1.7 PHITS での読み込みファイルの指定

PHITS で読み込みが必要となるファイルは、Decay-Turtle からの出力をソースとして使うときのソースファイルです。前者は、2.6MB のファイルで、各 PE が同じファイルを同時に読み込みを初めても、読み込みが最初の一回だけということもあり、それほどネットワークに負担を掛けないと思われますので、1 箇所に置いていても問題無いと思います。しかし、後者はファイルが 100MB 近く大きいもので、ソースの読み込みのたびにファイルをアクセスしに行きますので、1 箇所からの読み込みには問題が生じる可能性があります。そこで、Decay-Turtle のデータファイルを利用する場合は、あらかじめこのファイルを 各 PE のワークディレクトリ `/wk` の下に、例えば、`/wk/j9999/turtle/sours.dat` のようにコピーして、PHITS のインプットで、`file = /wk/j9999/turtle/sours.dat` のように指定して下さい。

12.2 メモリ共有型並列

12.2.1 実行方法

メモリ共有型並列版 PHITS は、Linux の場合やコマンドラインから直接実行ファイルを指定する場合を除き、PHITS のインプットファイルにおいて、最初のセクションの前に`$OMP=`N（N は使用する CPU コア数）を加えることで実行できます。その際、$N = 1$ の場合は並列計算を使用しません。また、$N = 0$ の場合は計算機が持つ全てのコアを使用します。これらは、`phits.bat`（Windows OS の場合）や `phits.sh`（Mac OS の場合）を利用して PHITS を実行する場合に動作します。

Linux の場合やコマンドラインから直接実行ファイルを指定する場合は、シングル版 PHITS の実行方法と同じく、

```
phits_LinIfort_OMP.exe < phits.inp
```

のようになります（phits_LinIfort_OMP.exe は実行ファイル名）。メモリ分散型並列と異なり、リダイレクトが使えるため、インプットファイル名 (phits.inp) は可変です。ただし、並列数を指定するために、環境変数 “OMP_NUM_THREADS” をコア数に合わせて設定する必要があります。通常、この環境変数はデフォルトでコア数に設定されていますが、最新のコンピュータは、1 つのコアで複数のスレッドを処理することが可能なため、コア数よりも多く設定されている場合があります。しかし、PHITS の並列処理では、“OMP_NUM_THREADS” にコア数以上の値を設定しても、ファイルの書き込み競合による計算遅延が起きてしまい、計算時間が長くなってしまいます。そのような場合は、手動で環境変数を変更してください。環境変数の変更は

```
export OMP_NUM_THREADS=8
```

のように行います。また、ハイブリッド並列の場合は、各実行ノードの環境変数をそれぞれ変更する必要があることにご注意ください。

バージョン 2.73 より、Windows OS については、メモリ共有型並列版 PHITS の実行ファイルとして 64bit 版をインストールするようにしました。32bit 版のメモリ共有型並列の実行ファイルの場合、使用するコア数が増えるとヒープメモリが不足してエラーが発生していましたが、64bit 版を使用することでこれが回避される可能性があります。

12.2.2　メモリ共有型並列計算の注意点

CPU コア数が 1 つの場合、メモリ共有型並列版 PHITS の計算時間は、シングル版 PHITS の計算時間よりも 2 倍程度長くなります。したがって、コア数が 2 以下のコンピュータでは、メモリ共有型並列計算による計算時間短縮のメリットはほとんどありません。

Linux で実行時にセグメンテーションエラーが生じる場合は、実行前にスタックサイズを広げることにより解決する場合があります。スタックサイズの広げ方は、

```
export OMP_STACKSIZE=1G
```

のようになります（スタックメモリを 1GB にする場合）。

メモリ共有型並列計算結果は、シングルでの計算結果と一致するように設計しています。両者で結果が異なる場合は、バグの可能性がありますので PHITS 事務局までご連絡ください。

13 FAQ

13.1 パラメータ設定関連

Q1.1 Ver.2.88 以前で動いていたインプットファイルがそれ以降の version で動きません。

A1.1 Ver.2.89 以降、ミスを誘発しやすい仕様をいくつか変更しています。基本的には、従来のインプットをそのまま使えるように設計していますが、場合によってはインプットの修正が必要になる場合があります。以下、その例です。

(1) Ver.2.89 以降では天然炭素の記号 C との混同を防ぐため [material] で c をコメントマークとして利用できません（それ以外のセクションでは引き続き利用可能です）。[material] でエラーが出るようになった場合は、コメントマークとして使っている c を$や#に変更するか、[parameters] セクションに icommat=1 と追記してください。

(2) 光子や電子の輸送のみを行う場合、従来は中性子の核データがなくても問題ありませんでしたが、Ver.2.93 以降は自動的に中性子の核データを利用する設定になっているため “There is no cross section table(s) in xsdir” というエラーが出てしまいます。この場合は、[parameters] に nucdata=0 と追記して核データを使わない設定に変更してください。

(3) Ver.2.96 以降では、[parameters] セクションが 2 つ以上ある場合、エラーを出して実行を止めるようにしました。2 つ以上定義している場合は、1 つに統合してください。

(4) Ver.2.96 以降、icntl に応じて不要なタリーを自動的に off にするようにしました。自動的に off にされたタリーセクション内に set や infl コマンドがある場合、それらも無効になりますのでご注意ください。なお、Ver.2.96 以降、off されたセクション内に set や infl コマンドがある場合は warning が出力されます。

Q1.2 電子・陽電子が輸送されません。

A1.2 電子・陽電子の計算には時間が掛かるためデフォルトでは輸送しない設定になっています。電子・陽電子を輸送したい場合は、negs=1 として EGS5 モードを利用するようにしてください。

Q1.3 核反応モデルに関して、デフォルトの設定は最も精度がよい設定ですか？

A1.3 基本的にはそうなるように設定していますが、重イオン核反応や高エネルギー核反応により生成される残留核の収率を精度よく計算したい場合は、irqmd=1 や ismm=1 のオプションを利用することを勧めています。ただし、これらのオプションを有効にすると、計算時間が長くなってしまう場合があります。

Q1.4 イベントジェネレータモードは、どのような場合に使えばよいのでしょうか？

A1.4 イベントジェネレータモード（以下、e-mode）を使った方がよい計算は、検出器の応答関数計算や、半導体ソフトエラー発生率の計算など、イベント毎の情報が必要となる計算です。具体的には、[t-deposit], [t-let], [t-yield] などを使う計算では、e-mode を使った方がよい場合が多いです。逆に、使わない方がよい計算は、中性子束やガンマ線束だけを求める遮蔽計算などです。[t-track], [t-cross] などしか利用しない場合は、一般的に e-mode を使う必要はありません。詳しくは「5.2.26 Event Generator Mode」をご参照ください。なお、version2.76 からイベントジェネレータモードを使う場合の奨励値は e-mode=2 となっていますので、ご注意ください。

Q1.5 統計誤差導出方法 (istdev の設定) はどういう条件で切り替えれば良いですか？

A1.5 バッチ分散モード (istdev=-1 or 1) しか選択できないメモリ共有型並列計算の場合を除き、基本的に全ヒストリー数 (maxcas*maxbch) で統計誤差を評価するヒストリー分散モード (istdev=-2 or 2) の方を推奨します。しかしながら、ヒストリー分散モードでは、ヒストリーの計算が終了する毎に誤差情

報を記録するため、タリーのメッシュ数が大きい場合 (xyz メッシュで `nx` や `ny` を数 1000 ずつ設定するなど) には、計算時間が異常に長くなる可能性があります。そのような条件で計算する際は、バッチ分散モードに切り替えて、`maxbch` を十分に大きく (10 以上) 設定するようにしてください。

Q1.6 JENDL 以外の核データを PHITS で使うことは可能でしょうか？

A1.6 ACE フォーマット（MCNP が読み込める形式）であれば可能です。以下、その手順です。

(1) 新しく入手した核データに付録されている xsdir ファイルをテキストエディタで開き、核データのアドレス情報に関する部分（例えば 1001.80c 0.999167 xdata/endf71x/H/1001.710nc 0 1 4 17969 0 0 2.5301E-08）を全て PHITS の data フォルダ内にある xsdir.jnd の末尾にコピー＆ペーストする。

(2) PHITS の XS フォルダ内に、上記アドレス情報が一致するように新しく入手した核データファイルをコピーする。例えば、上記のアドレス情報であれば、XS フォルダ内に xdata/endf71x/H フォルダを作って 1001.710nc ファイルをその中にコピーする。

(3) `[material]` セクションで、使う核データを明示的に指定する。例えば、上記のアドレス情報であれば 1H.80c もしくは 1001.80c と指定する。核データを明示的に指定しない場合は、xsdir.jnd の中でその核種に対して最初に書かれている核データが自動的に使われます。

Q1.7 光核反応により生成する粒子（中性子・放射性核種）の統計を効果的にためる方法はありますか？

A1.7 `pnimul` パラメータを使うことにより、他の反応（コンプトン散乱など）と比べて光核反応を起こしやすくすることができます。例えば、`pnimul=2.0` とすると、光核反応を引き起こす確率を 2 倍にする代わりに、光核反応で生成した粒子のウェイトを 1/2 にします。ウェイトを変化させるので、イベント毎の情報が必要な場合は使えません。また、`pnimul` パラメータを大きくしすぎると（100 以上）、光子・電子の挙動がずれてきてしまいますので、本パラメータを使う場合は、光子・電子の挙動（フラックス・線量など）が大きく変動しないことを確認しながら使ってください。

Q1.8 再開始計算を行おうとすると、タリーを変更していないにも関わらず “Error: inconsistent tally parameters” というエラーが出て止まります。対処方法はありますか?

A1.8 この問題は、桁落ちの関係で新旧タリーのパラメータが異なると PHITS が認識してしまうことにより発生します。タリーで使うパラメータを複雑な数式などで定義した場合によく発生します。この問題が生じた場合は `[parameters]` セクションで `ireschk = 1` と設定し、タリーの一致性をチェックしないモードで再開始計算を実行して下さい。

13.2 エラー、コンパイル関連

Q2.1 コンパイルがうまくできません。

A2.1 PHITS のコンパイルに関しては、基本的には「11 コンパイル」をご参照ください。PHITS は、Intel Fortran 11.1 以降及び gfortran 4.7 以降でコンパイル可能です。ただし、Windows の gfortran ver. 4.9 - 5.4 では正しくコンパイルできませんので 11.1.1 節をご参照の上、最新版をインストールしてください。なお、PHITS 事務局が奨励するコンパイラは、計算速度の観点から Intel fortran です（Q2.6 参照）。

Q2.2 巨大な体系について計算しようとすると、Segmentation エラーが起きてしまいます。

A2.2 PHITS では使用するメモリの最大値をあらかじめ定義しており、実際に必要となるメモリがその最大値を超えてしまうと、Segmentation エラーを引き起こします。使用するメモリの最大値 は、「src」フォルダにある「param.inc」中の mdas パラメータで定義されていますので、必要に応じてこの値を大きくし再コンパイルしてください。なお、Windows では 1 つの Fortran プログラムが 2GB 以上のメモリを使用することは許されていませんので、`mdas` の上限値は 268435450 です。それ以上のメモリ領域が必要となる計算には、64bit の Windows もしくは Linux をお使い下さい。

Q2.3 外部ファイルを取り込むためのコマンド「infl:」が使えません。

A2.3 infl:コマンドを使うためには、インプットファイルの 1 行目に「`file=`インプットファイル名」と書いておく必要があります。もしくは、通常の PHITS 用インプットファイルの他に、別のファイルを準備し、そのファイルの先頭行に「`file=`通常 PHITS のインプットファイル名」と書いて実行することも可能です。

Q2.4 Windows では実行できるインプットファイルが LINUX や UNIX では実行できません。

A2.4 いろいろな原因が考えられますが、Windows と LINUX 系 (UNIX 含む) では、リターンコードが違いファイルの転送に失敗している可能性があります。通常、Windows から LINUX 系にファイルを転送する際、FTP などのプロトコルを使うと思いますが、その転送の際、アスキーモードとバイナリーモードというのがあり、アスキーファイル (PHITS のインプットファイルなど) は、アスキーモードで転送しないと、ファイルが破損してしまう可能性があります。お使いの FTP ソフトの設定をご確認ください。なお、Windows で作られたバイナリファイル (PHITS の実行ファイルなど) を LINUX 系に転送しても、通常、正しく動作しません。

Q2.5 Cygwin 上で PHITS は動作しますか？

A2.5 はい。PHITS の makefile にある Cygwin の項目をご参照ください。

Q2.6 Intel fortran と gfortran、どちらが PHITS をコンパイルするのに適していますか？

A2.6 基本的には、Intel fortran を使うことをお勧めいたします。理由は、計算速度が速いためです。gfortran では、最適化オプションを付けると正しく動作しない場合があるため、奨励の最適化オプションは-O0（すなわち最適化しない）です。したがって、平均で 3-5 倍くらいの計算速度の差があると考えています。また、最新の gfortran では PHITS を正しくコンパイルできない問題もあります（Q2.1 参照）。ただし、Intel fortran は Linux 版を含めて有償ですので、その点、ご留意ください。

Q2.7 メモリ分散型並列（MPI）と共有型並列（OpenMP）はどのように使い分けるのがよいでしょうか？

A2.7 MPI プロトコルがインストールされたコンピュータであれば、基本的に MPI の方が計算速度は速くなります。ただし、通常の Windows や Mac には MPI プロトコルがインストールされていませんので、簡単に並列計算を実行したいのであれば、OpenMP を使うことをお勧めしています。また、ボクセルファントムや高エネルギー核データライブラリなどの巨大なメモリを使用する計算では、MPI だとメモリ容量をオーバーしてしまう場合があります。その場合は、MPI と OpenMP を組み合わせたハイブリッド並列を実施する必要があります。その際、メモリの許す限りできるだけ MPI 並列数を増やした方が計算速度は速くなります。ただし、ジョブコントロール用にも OpenMP 並列数分のコアを使用しますので、MPI 並列数の設定にはご注意ください。例えば、128 コアの計算機を使って OpenMP 並列数 8 で計算する場合、指定できる MPI 並列数は、128/8 – 1 = 15、となります。

Q2.8 DCHAIN を実行した際に、「指定されたファイルが見つかりません」といったエラーが表示されます。

A2.8 DCHAIN のインプットファイルや infl コマンドを含む PHITS のインプットファイルが全角文字を含むフォルダ名の中に置かれている場合、適切に動作しません。その場合は、半角文字のみのフォルダ名に変更してください。

Q2.9 ボクセルファントムなど Lattice 構造を [transform] で回転させると lost particle が頻発します。

A2.9 Lattice 構造を入れる外枠セルと Lattice 構造そのものの大きさが完全に一致していると、回転させた際の桁落ちの関係で Lost particle が頻発する場合があります。Lattice 構造を入れる外枠セルを少しだけ小さくすると回避できる可能性が高いです。詳細は`/phits/lecture/advanced/voxel/` をご参照ください。

13.3 タリー関連

Q3.1 [t-heat] と [t-deposit] は何が違うのでしょうか？

A3.1 共に放射線による付与エネルギー（発熱）を出力するタリーです。ただし、[t-heat] でしか実行できなかったカーマ近似による計算は、version 3.05 以降、[t-deposit] でも可能となったため、現在 [t-heat] の利用は奨励していません。

Q3.2 [t-track] で重イオンの track length を計算すると、結果が不自然なのですが。

A3.2 重イオンのエネルギーは、入射エネルギーや切断エネルギーに対しては核子あたりのエネルギー [MeV/n] で指定しますが、タリーの中でエネルギーを指定する場合は、核子あたりではなく、全エネルギー [MeV] で指定します。ただし、`[parameters]` において `iMeVperu=1` とすることで、タリーにおけるエネルギーの単位を MeV/n に変えることができます。

Q3.3 [t-let]、[t-sed] の結果が不自然なのですが。

A3.3 letmat で LET を計算する物質を指定する際、その物質が PHITS の体系内で使われている場合、密度はその体系内で使用している密度となります。例えば、水蒸気内の発熱量に対する LET 分布を、液体の水 (1g/cm^3) に対する LET の関数として計算したい場合、[Material] で２つの水を定義し、１つを PHITS の体系内で使用し、もう１つを letmat で使用する必要があります。また、letmat で使用する物質の密度は、原子数密度の絶対値 (1H 6.893407e-2 16O 3.446704e-2) を指定してください。詳しくは、奨励設定ファイルの ParticleTherapy をご参照下さい。また、`[t-let]` の場合、LET メッシュ幅を細かく設定しすぎると結果がガタガタしてしまいますので、あまり細かいメッシュ幅を設定しないようにしてください（1 桁 10 メッシュ程度まで）。

Q3.4 タリー結果の誤差はどのように計算されているのでしょうか？

A3.4 バージョン 2.50 より、正しく計算できるように修正しました。詳しくは「5.2.2 ヒストリー数、バンク配列の大きさ」をご参照ください。

Q3.5 並列計算で dump モードは使えるでしょうか？

A3.5 基本的には使えます。ただし、メモリ共有型（OpenMP）並列の場合は、1 つのファイルに全てのノードからの情報を書き込むため、書き込む順番がヒストリー番号毎に並ばなくなってしまいます。一方、メモリ分散型（MPI）並列の場合は、dump ファイル名にコア番号付けたファイルを出力しますので、そのような問題は起きません。

Q3.6 mesh = reg と mesh = xyz を用いて同じ直方体内のフラックスや発熱量を計算した場合に、それぞれの結果が異なるのですが。

A3.6 タリー結果を単位体積あたりに規格化して出力する場合、xyz や r-z メッシュではタリー領域の体積を自動的に計算して出力しますが、reg メッシュの場合は、[volume] セクションで体積を指定しない限り体積を 1cm^3 として出力します。例えば、mesh = reg で [volume] セクションがない場合、[t-track] の unit=1[1/cm^2/source] と unit=4[cm/source] は同じ値を与えます。

Q3.7 [t-deposit] で output=deposit とした場合に入射エネルギーよりも高いイベントが発生するのはなぜでしょうか？

A3.7 発熱する核反応（Q 値が正）が発生すると、入射粒子よりも 2 次粒子の合計エネルギーが高くなるため、入射エネルギーよりも付与エネルギーが大きくなる場合があります。また、xyz メッシュもしくは r-z メッシュかつエネルギー分散を考慮した場合（nedisp≠0）、エネルギー分散の考慮方法の仕様により、イベント毎の情報が正しく計算できなくなってしまいます。そのような場合は、mesh = reg として、[cell] セクションで領域を細かく分割して計算してください。

Q3.8 axis = xz で表示する際、横軸と縦軸を入れ替えることは可能でしょうか？

A3.8 可能です。まず、該当タリーの x-type と z-type に関連するパラメータを入れ替えます。次に下記 3 行をそのタリーに追加します。

```
trcl = 0 0 0 0 0 1 0 1 0 1 0 0
x-txt = x [cm]
y-txt = z [cm]
```

これで図が 90 度回転します。

13.4 線源設定関連

Q4.1 等方線源ソース (s-type=9, dir = −all) のときの規格化はどのようにすればよいですか？

A4.1 PHITS で作る等方線源は、指定した球の中でフラックスが一定値: $1/\pi/r_1^2$ (r_1 は球の内径) となるように作られています。したがって、PHITS の結果 (/source) を単位フルエンス当たりに変換するには、結果に πr_1^2 を乗じてください。また、等方線源の作成にはウェイトの概念を利用していますので、検出器の応答関数などイベント毎の分布計算には利用できませんので、ご注意ください。そのような計算が必要な方は、`dir = iso` としてください。

Q4.2 PHITS で認識する線源領域が、実際の領域とずれるのですが。

A4.2 面線源や点線源を定義する際、その面や点が、Cell で使う surface と完璧に一致している場合、PHITS の中で認識する線源領域が実際の領域とずれる場合があります。このような状態を避けるため、面線源や点線源は、cell で使う面とは少しずらして作成するようにしてください。また、特定の面に完璧に平行なビームを発生させた場合も同様のエラーが起きる可能性がありますのでご注意下さい。

13.5 その他

Q5.1 規制庁への加速器施設や廃棄物保管庫等の許認可申請において、PHITS による遮蔽計算の結果を使用することができるでしょうか?

A5.1 RI 法の申請においては、PHITS が認められています。J-PARC をはじめとして、放医研の新治療棟、北大の陽子線治療施設、神奈川がんセンターの重粒子線治療施設等で、実績が多数あります。一方、炉規法に関しては、申請において認められるかどうかに未確認です。許可取得例をご存知でしたら、事務局までご一報ください。これまでにユーザーの皆様からいただいた情報は、次のウェブページにまとめています。`https://phits.jaea.go.jp/image/shinsei-PHITS.pdf`

Q5.2 PHITS で得られた解析結果を商用目的で利用することは可能でしょうか？

A5.2 基本的には可能です。例えば，PHITS の解析結果を元に新しい検出器を開発して販売することや，外部機関から加速器の遮蔽設計を請け負って対価を得ることも問題ありません。ただし，PHITS を含む商品（例えば PHITS の実行ファイルが含まれる治療計画システムなど）を開発して販売する場合は，原子力機構と商用利用契約を結んでいただく必要があります。そのような計画がある場合は，早めに事務局までお知らせ下さい。

Q5.3 放射線に関連する物理現象で PHITS で扱えないものは何でしょうか?

A5.3 PHITS は個々の放射線の挙動をモンテカルロ法を用いて解析するため，複数の放射線による影響（電流や放射線自身が作る電場の影響など）を考慮することができません。また，物理過程の後に引き起こされる化学過程や材質の変化は模擬することができません。詳しくは，B 節をご参照ください。

APPENDIX

A　ANGEL

ANGEL は、簡単なインプットから、EPS (Enhanced PostScript) 形式のグラフを素早く描くために設計されたプログラムです。すなわち、ANGEL は、Angel 言語 (数値データファイルをグラフ化するために書き加える必要最低限の命令) から PostScript 言語 (Adobe 社のグラフィックコントロールプログラムの規格のプログラム言語) への翻訳機です。ANGEL は、PHITS のソースに含まれ、また PHITS の出力もアスキーファイルの他に eps ファイルの出力を得ることができますが、その後のグラフの整形などに ANGEL が必要になる場合があります。単体の ANGEL をコンパイルするためには、“/src/” フォルダにある “make.ang” を “makefile” に名称変更して “make” する必要があります。その他の ANGEL に関することは、ANGEL のマニュアルを参照してください。

A.1　タリー定義文

ANGEL パラメータ

タリー出力で、ANGEL のパラメータを追加します。

```
angel = xmin(1.0) ymin(1.3e-8)
```

ここで定義したパラメーターは、タリー出力の中で

```
p: xmin(1.0) ymin(1.3e-8)
```

と記載され、横軸と縦軸の最小値をそれぞれ指定した値に変更します。

表 167 に主要な ANGEL パラメータをまとめました。他の ANGEL のパラメータに関しては、ANGEL のマニュアルをご覧ください。

表 167: ANGEL パラメータ

ANGEL パラメータ	説明
xmin	横軸の目盛の最小値。
xmax	横軸の目盛の最大値。
ymin	縦軸の目盛の最小値。
ymax	縦軸の目盛の最大値。
xlin	横軸を線形表示。
xlog	横軸を対数表示。
ylin	縦軸を線形表示。
ylog	縦軸を対数表示。
cmin	2 次元表示（カラープロット）の最小値。
cmax	2 次元表示（カラープロット）の最大値。
cmnm	長さの単位を cm から nm に変換。
cmum	長さの単位を cm から μm に変換。
cmmm	長さの単位を cm から mm に変換。
cmmt	長さの単位を cm から m に変換。
cmkm	長さの単位を cm から km に変換。
nsps	時間の単位を nsec から psec に変換。
nsus	時間の単位を nsec から μsec に変換。
nsms	時間の単位を nsec から msec に変換。
nssc	時間の単位を nsec から sec に変換。
scmn	時間の単位を sec から min に変換。
schr	時間の単位を sec から hour に変換。
scdy	時間の単位を sec から day に変換。
scyr	時間の単位を sec から year に変換。

バージョン 2.89 以降、デフォルトの出力サイズを 4A から US レターに変更しました。もし、出力する eps ファイルの用紙サイズを変更する場合は、ANGEL パラメータとして、a4us(US レター)、a3pp(A3)、a4pp(A4)、a5pp(A5)、b3pp(B3)、b4pp(B4)、b5pp(B5) を指定してください。() の中は各々の用紙サイズです。

SANGEL パラメータ

タリー出力において、ANGEL 用のパラメータを追加できるようになります。ANGEL パラメータ定義文では `p:`で始まる ANGEL 用パラメータしか定義できませんが、本定義文を使用することで任意の ANGEL 用パラメータを定義できるようになります。例えば、`infl:`パラメータを定義することで、タリー出力ファイルに対して実験値などをまとめた外部ファイルを挿入し、計算値と比較した結果を図示することができます。

書式は、`sangel=`でタリー出力に書き出す行数を指定し、その次の行からその行数だけパラメータを加えます。例えば、

```
sangel = 2
infl:{exp.dat}
w: ($\theta=$ 0 deg) / X(10) Y(100)
```

とすることで、`exp.dat` にまとめた実験値を図示すると共に、`X=10, Y=100` の座標位置に “θ=0 deg” というコメントを加えることができます。

B　PHITS で扱えない物理現象リスト

「PHITS で扱えない物理現象は何ですか？」という質問を頻繁に受けましたので，以下にその一覧を示します。事務局等に質問する際，ご参照ください。

- 臨界計算
- 放射線同士の相互作用（例：高エネルギー粒子衝突実験の模擬）
- 放射線自身が生み出す電場の影響（例：レーザー加速，プラズマ）
- 電子が引き起こす核反応
- 1keV 以下の光子（可視光含む）の発生と輸送（例：チェレンコフ光，シンクロトロン放射光，シンチレーション発光）
- 1keV 以下の電子（δ 線含む）の発生と輸送[107]
- 電流に関連する物理現象（例：ガス検出器における電子なだれ）
- 電荷交換反応（イオンの電荷が物質で変化する事象，ただし有効電荷の変化は考慮）[107]
- 個々の電離・励起イベントの発生[107]
- 偏光の影響[108]
- 化学反応（例：ラジカル生成・輸送）
- 生物反応（例：DNA 損傷）
- 放射線照射による材料特性の変化，欠陥の時間発展
- 物質の状態の設定[109]（例：結晶・分子構造・温度）
- 微細な殻構造に依存する核反応[109]（例：Li(p,n) 反応）
- 電子以外の放射線による特性 X 線の発生（例：PIXE）
- 遅発中性子の発生

[107] 飛跡構造解析モードを利用した場合は可能
[108] Nuclear Florescence Resonance(NRF) では考慮可能
[109] 核データを利用した場合は可能

C　評価済み核データライブラリ

PHITS のパッケージに、エネルギー 20 MeV 以下の中性子に対応した JENDL-4.0、及びエネルギー 200 MeV 以下の中性子及び陽子に対応した JENDL-4.0/HE が格納されています。表 168 に JENDL-4.0 ライブラリに含まれる核種、表 169 に JENDL-4.0/HE ライブラリに含まれる核種をそれぞれ示します。ファイルサイズの大きさを考慮して、JENDL-4.0/HE については 22 核種のみ格納しています。他の核種をご希望の場合、PHITS 事務局までお問い合わせください。

表 168: エネルギー 20 MeV 以下の中性子に対応した JENDL-4.0 ライブラリ

PHITS パッケージ	核種									
格納	$^{1}_{1}$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					

表 169: エネルギー 200 MeV 以下の中性子及び陽子に対応した JENDL-4.0/HE ライブラリ

PHITS パッケージ	核種
格納	$^{1}_{1}$H $^{2}_{1}$H $^{12}_{6}$C $^{13}_{6}$C $^{14}_{7}$N $^{16}_{8}$O $^{27}_{13}$Al $^{28}_{14}$Si $^{29}_{14}$Si $^{30}_{14}$Si $^{54}_{26}$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 $^{9}_{4}$Be†
未格納	$^{19}_{9}$F $^{23}_{11}$Na $^{24}_{12}$Mg $^{25}_{12}$Mg $^{26}_{12}$Mg $^{31}_{15}$P $^{35}_{17}$Cl $^{37}_{17}$Cl $^{36}_{18}$Ar $^{38}_{18}$Ar $^{40}_{18}$Ar $^{39}_{19}$K $^{41}_{19}$K $^{40}_{20}$Ca $^{42}_{20}$Ca $^{43}_{20}$Ca $^{44}_{20}$Ca $^{46}_{20}$Ca $^{48}_{20}$Ca $^{46}_{22}$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 $^{59}_{27}$Co $^{58}_{28}$Ni $^{60}_{28}$Ni $^{61}_{28}$Ni $^{62}_{28}$Ni $^{64}_{28}$Ni $^{64}_{30}$Zn $^{66}_{30}$Zn $^{67}_{30}$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 $^{78}_{34}$Se $^{80}_{34}$Se $^{82}_{34}$Se $^{90}_{40}$Zr $^{91}_{40}$Zr $^{92}_{40}$Zr $^{94}_{40}$Zr $^{96}_{40}$Zr $^{93}_{41}$Nb $^{92}_{42}$Mo $^{94}_{42}$Mo $^{95}_{42}$Mo $^{96}_{42}$Mo $^{97}_{42}$Mo $^{98}_{42}$Mo $^{100}_{42}$Mo $^{113}_{49}$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 $^{124}_{50}$Sn $^{127}_{51}$I $^{133}_{55}$Cs $^{181}_{73}$Ta $^{180}_{74}$W $^{182}_{74}$W $^{183}_{74}$W $^{184}_{74}$W $^{186}_{74}$W $^{197}_{79}$Au $^{196}_{80}$Hg $^{198}_{80}$Hg $^{199}_{80}$Hg $^{200}_{80}$Hg $^{201}_{80}$Hg $^{202}_{80}$Hg $^{204}_{80}$Hg $^{235}_{92}$U $^{238}_{92}$U $^{237}_{93}$Np $^{238}_{94}$Pu $^{239}_{94}$Pu $^{240}_{94}$Pu $^{241}_{94}$Pu $^{242}_{94}$Pu $^{241}_{95}$Am $^{242}_{95}$Am $^{242m}_{95}$Am $^{6}_{3}$Li† $^{7}_{3}$Li†

† 陽子データのみ格納。

PHITS パッケージに未格納の核種をご希望の方は、PHITS 事務局までお問い合わせください。

索 引

PHITS *Ver.3.24 User's Manual*

2012 年 12 月 27 日　Ver.2.52　発行
2013 年 11 月 19 日　Ver.2.64　発行
2014 年 01 月 09 日　Ver.2.64a　発行
2015 年 03 月 23 日　Ver.2.76　発行
2015 年 12 月 16 日　Ver.2.82　発行
2016 年 09 月 29 日　Ver.2.88　発行
2017 年 10 月 31 日　Ver.3.01　発行
2017 年 12 月 01 日　Ver.3.02　発行
2018 年 02 月 01 日　Ver.3.03　発行
2018 年 02 月 16 日　Ver.3.04　発行
2018 年 03 月 14 日　Ver.3.05　発行
2018 年 05 月 29 日　Ver.3.06　発行
2018 年 07 月 05 日　Ver.3.07　発行
2018 年 08 月 20 日　Ver.3.08　発行
2019 年 04 月 03 日　Ver.3.10　発行
2019 年 05 月 16 日　Ver.3.11　発行
2019 年 06 月 19 日　Ver.3.12　発行
2019 年 08 月 02 日　Ver.3.13　発行
2019 年 08 月 19 日　Ver.3.14　発行
2019 年 09 月 12 日　Ver.3.15　発行
2019 年 09 月 26 日　Ver.3.16　発行
2019 年 10 月 29 日　Ver.3.17　発行
2020 年 03 月 19 日　Ver.3.20　発行
2020 年 09 月 14 日　Ver.3.21　発行
2020 年 10 月 09 日　Ver.3.22　発行
2021 年 01 月 12 日　Ver.3.23　発行
2021 年 03 月 12 日　Ver.3.24　発行

PHITS 開発チーム：
仁井田浩二[1]，佐藤達彦[2]，岩元洋介[2]，橋本慎太郎[2,†]，小川達彦[2]，古田琢哉[2]，
安部晋一郎[2]，甲斐健師[2]，松田規宏[2]，松谷悠佑[2]，Hunter Ratliff[2]，Lan Yao[2]，
Pi-En Tsai[2,*]，岩瀬広[3]，執行信寛[4]，Lembit Sihver[5]
[1](財) 高度情報科学技術研究機構,　[2](国研) 日本原子力研究開発機構,
[3](共) 高エネルギー加速器研究機構,　[4](国) 九州大学,
[5] ウィーン工科大学, オーストリア,　[*] 現所属: SLAC, USA
[†] 本マニュアルの編集担当