

---

---

# *PHITS*

---

*Ver. 2.76*

---

*User's Manual*

---

## 目次

<b>1</b>	<b>はじめに</b>	<b>1</b>
1.1	最近の改良点	1
1.2	開発者	7
1.3	PHITS の参考文献	7
<b>2</b>	<b>インストール、コンパイル及び使用方法</b>	<b>9</b>
2.1	動作環境	9
2.2	Windows でのインストール及び実行方法	9
2.3	Mac でのインストール及び実行方法	10
2.4	Makefile を利用したコンパイル	11
2.5	Microsoft Visual Studio と Intel Fortran を利用したコンパイル (Windows 用)	12
2.6	ANGEL のコンパイル	12
2.7	実行シェル	12
2.8	実行の途中中断	13
2.9	配列の大きさの変更	14
<b>3</b>	<b>入力ファイルの書式</b>	<b>15</b>
3.1	セクションの種類	15
3.2	読み込みコントロール	16
3.3	ファイルの挿入	17
3.4	ユーザー定義定数	18
3.5	数式の利用	18
3.6	粒子の表式	19
<b>4</b>	<b>セクション書式</b>	<b>21</b>
4.1	[Title] セクション	21
4.2	[Parameters] セクション	22
4.2.1	計算モード	22
4.2.2	ヒストリー数、バンク配列の大きさ	23
4.2.3	計算打切エネルギー、切り替えエネルギー	25
4.2.4	時間カット、ウエイトカット、ウエイトウインドウ	27
4.2.5	計算モデルオプション (1)	28
4.2.6	計算モデルオプション (2)	29
4.2.7	計算モデルオプション (3)	30
4.2.8	計算モデルオプション (4)	31
4.2.9	出力オプション (1)	32
4.2.10	出力オプション (2)	35
4.2.11	出力オプション (3)	36
4.2.12	出力オプション (4)	37
4.2.13	出力オプション (5)	38
4.2.14	幾何形状のエラー関係	39
4.2.15	入出力ファイル名	40
4.2.16	その他	41
4.2.17	低エネルギー中性子の物理パラメータ	41
4.2.18	光子・電子輸送オリジナルモデルの物理パラメータ	42

4.2.19	EGS5 用パラメータ	43
4.2.20	Dumpall オプション	45
4.2.21	Event Generator Mode	50
4.3	[Source] セクション	52
4.3.1	<Source> : マルチソース	53
4.3.2	共通パラメータ	54
4.3.3	円柱分布ソース	56
4.3.4	角柱分布ソース	56
4.3.5	ガウス分布ソース ( $x, y, z$ 独立)	57
4.3.6	一般パラボラ分布ソース ( $x, y, z$ 独立)	57
4.3.7	ガウス分布ソース ( $xy$ 平面)	58
4.3.8	一般パラボラ分布ソース ( $xy$ 平面)	58
4.3.9	球及び球殻分布ソース	59
4.3.10	s-type = 11 ソース	59
4.3.11	s-type = 12 ソース	60
4.3.12	円錐形状分布ソース	61
4.3.13	dump データソース	62
4.3.14	ユーザー定義ソース	64
4.3.15	エネルギー分布の定義	67
4.3.16	角分布の定義	73
4.3.17	マルチソースの例題	75
4.3.18	ダクトソースオプション	79
4.4	[Material] セクション	82
4.4.1	書式	82
4.4.2	核種の定義	83
4.4.3	密度の定義	83
4.4.4	物質パラメータ	83
4.4.5	$S(\alpha, \beta)$ の指定	84
4.4.6	例題	84
4.5	[Cell] セクション	86
4.5.1	書式	86
4.5.2	セルの記述方法	87
4.5.3	Universe 構造	90
4.5.4	Lattice 構造	93
4.5.5	繰り返し幾何形状	96
4.6	[Surface] セクション	101
4.6.1	書式	101
4.6.2	マクロボディー	103
4.6.3	マクロボディーの面定義	104
4.7	[Transform] セクション	105
4.7.1	書式	105
4.7.2	座標変換の定義	105
4.8	[Importance] セクション	108
4.9	[Weight Window] セクション	110
4.10	[Volume] セクション	111
4.11	[Temperature] セクション	112

4.12	[ Brems Bias ] セクション	113
4.13	[ Photon Weight ] セクション	114
4.14	[ Forced Collisions ] セクション	115
4.15	[ Magnetic Field ] セクション	116
4.15.1	荷電粒子	116
4.15.2	中性子	117
4.16	[ Electro Magnetic Field ] セクション	119
4.17	[ Counter ] セクション	120
4.18	[ Reg Name ] セクション	121
4.19	[ Mat Name Color ] セクション	122
4.20	[ Mat Time Change ] セクション	124
4.21	[ Super Mirror ] セクション	125
4.22	[ Elastic Option ] セクション	126
4.23	[ Timer ] セクション	127
4.24	[ delta ray ] セクション	128
4.25	[ Multiplier ] セクション	129
<b>5</b>	<b>タリー共通パラメータの書式</b>	<b>131</b>
5.1	形状メッシュ	131
5.1.1	領域メッシュ	131
5.1.2	階層構造の領域と体積の定義	132
5.1.3	r-z メッシュ	134
5.1.4	xyz メッシュ	134
5.2	エネルギーメッシュ	135
5.3	LET メッシュ	135
5.4	時間メッシュ	135
5.5	角度メッシュ	135
5.6	メッシュ定義文	136
5.6.1	メッシュタイプ	136
5.6.2	e-type = 1 の場合	136
5.6.3	e-type = 2, 3 の場合	137
5.6.4	e-type = 4 の場合	137
5.6.5	e-type = 5 の場合	137
5.7	他のタリー定義文	137
5.7.1	粒子定義文	137
5.7.2	axis 定義文	138
5.7.3	file 定義文	139
5.7.4	resfile 定義文	139
5.7.5	unit 定義文	140
5.7.6	factor 定義文	140
5.7.7	output 定義文	140
5.7.8	info 定義文	140
5.7.9	title 定義文	140
5.7.10	ANGEL パラメータ定義文	141
5.7.11	2d-type 定義文	141
5.7.12	gshow 定義文	142

5.7.13	rshow 定義文	142
5.7.14	x-txt, y-txt, z-txt 定義文	143
5.7.15	volmat 定義文	143
5.7.16	epsout 定義文	143
5.7.17	カウンター定義文	144
5.7.18	resol 分解能、width 線太さ定義文	144
5.7.19	trcl 座標変換	144
5.7.20	dump 定義文	144
5.8	複数のタリー結果の統合機能	146
<b>6</b>	<b>タリー入力書式</b>	<b>148</b>
6.1	[T-Track] セクション	148
6.2	[T-Cross] セクション	152
6.3	[T-Heat] セクション	156
6.4	[T-Deposit] セクション	159
6.5	[T-Deposit2] セクション	162
6.6	[T-Yield] セクション	164
6.7	[T-Product] セクション	167
6.8	[T-DPA] セクション	171
6.9	[T-LET] セクション	174
6.10	[T-SED] セクション	177
6.11	[T-Time] セクション	180
6.12	[T-Star] セクション	183
6.13	[T-Dchain] セクション	186
6.14	[T-Userdefined] セクション	190
6.15	[T-Gshow] セクション	195
6.16	[T-Rshow] セクション	197
6.17	[T-3Dshow] セクション	199
6.17.1	box の定義	202
6.17.2	3dshow の例題	203
<b>7</b>	<b>タリーを用いた体積、面積計算</b>	<b>206</b>
<b>8</b>	<b>dump ファイルの処理</b>	<b>208</b>
<b>9</b>	<b>出力中性子、光子データフォーマット</b>	<b>213</b>
<b>10</b>	<b>領域エラーチェック</b>	<b>215</b>
<b>11</b>	<b>並列版のための指定方法</b>	<b>217</b>
11.1	メモリ分散型並列	217
11.1.1	実行方法	217
11.1.2	maxcas, maxbch の指定方法	217
11.1.3	異常終了の処理	218
11.1.4	PHITS での ncut, gcut, pcut, dumpall ファイルの指定	218
11.1.5	PHITS での読み込みファイルの指定	218
11.2	メモリ共有型並列	219
11.2.1	実行方法	219

11.2.2	メモリ共有型並列計算の注意点 . . . . .	219
<b>12</b>	<b>FAQ</b>	<b>221</b>
12.1	パラメータ設定関連 . . . . .	221
12.2	エラー、コンパイル関連 . . . . .	221
12.3	タリー関連 . . . . .	222
12.4	線源設定関連 . . . . .	223
	索引	<b>224</b>

# 1 はじめに

*PHITS* コードは、日本原子力研究所(以下、「原研」と記す。現在の組織名は、日本原子力研究開発機構であり、以下、「原子力機構」と記す。)が開発した高エネルギー核反応モデル組込み核子中間子輸送コード NMTC/JAM ver.2 に、高度情報科学技術研究機構 (RIST)、東北大、原研/原子力機構の協力のもとに、重イオンの輸送計算機能を組み込んだ粒子、重イオン輸送統合コードシステム (Particle and Heavy Ion Transport code System; *PHITS*) です。

NMTC/JAM ver.2 では、低エネルギーの中性子、光子、電子の輸送を含むことにより、高エネルギーから低エネルギーまでの輸送を NMTC/JAM コードだけで計算することが可能です。低エネルギーの輸送部分に ACE 形式の断面積データを用いた場合、MCNP4C と同じ結果を与えます。従って、今までの NMTC/JAM と MCNP4C のつなぎ計算の結果も同じ結果を与え、しかも、タリーの結果は両者を統合したものが得られます。

これらの NMTC/JAM の成果に、重イオン核反応と物質中の重イオン輸送の機能を加え、MCNPX を超える機能を持つ粒子、重イオン輸送統合コードに発展しました。

このマニュアルは、JAERI-Data/Code 2001-007 のマニュアルの日本語訳抜粋を含んでいます。また、上記の英語のマニュアル完成以後のコードの改良に伴う変更も記述してあります。例えば、GG 幾何形状の導入、並列化、DPA タリーの導入、低エネルギーの中性子、光子、電子の輸送などです。しかしながら、コードの概要、物理モデルなどの解説は、コンパクトにするために省いています。

今後も、新しい機能が入り次第改訂していきますので、よろしく願いいたします。皆さんに使ってもらって、バグ潰しと改良を加えたいと思います。できるだけ多くの利用をお願いいたします。また、バグ、不都合の報告、改良の希望など、お待ちしております。

## 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 2.76 では主に次の改良を行いました。

- 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 `sum` タリー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。

バージョン 2.74 では主に次の改良を行いました。

- PHITS パッケージに含まれる DCHAIN-SP のバージョンを DCHAIN-SP2001 (`dchain264.exe`) から DCHAIN-SP2014(`dchain274.exe`) に変更しました。DCHAIN-SP2001 と比べた DCHAIN-SP2014 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) の足立将晶さん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“sum タリー”を追加しました（ただし、現在のところ [t-track] と [t-deposit] のみで有効）。詳しくは、5.8 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 の三浦孝充さんに実施していただきました。
- Kurotama モデルで、5GeV/n 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<sup>1</sup>をご参照下さい。
- $\gamma$  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 $\geq$ 1` や `igamma $\geq$ 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。

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、`na` や `nn` を負の値で与えられないようにしました。以前のバージョンでは、これらの値を負の値で与えると、角度を `degree` 単位で与えていた場合、適切に対数で内挿が行われておりませんでしたのでご注意ください。

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。

バージョン 2.70 では主に次の改良を行いました。

- 電磁カスケード計算コード EGS5<sup>2</sup> 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。詳しくは、4.2.19 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、100GeV 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、100GeV 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。

<sup>1</sup> L. Sihver *et al.*, Nucl. Instr. & Meth. B 334, 34-39 (2014).

<sup>2</sup> H. Hirayama *et al.*, SLAC-R-730 (2005) and KEK Report 2005-8 (2005).



- 核データライブラリに格納された荷電粒子放出断面積  $((n, p), (n, \alpha)$  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 R-JQMD モデルを開発しました。R-JQMD 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。

- 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 4.2.21 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=18,19 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。
- 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。

- 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  $\Delta$  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。

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 に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。

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。

- 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  $\gamma$  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 4.2.5 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。また、 ${}^6\text{Li}$ ,  ${}^{12}\text{C}$ ,  ${}^{14}\text{N}$ ,  ${}^{16}\text{O}$  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なりました。この改良により、これらの核種から  $\alpha$  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 4.16 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 4.3.15 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  $AN_{GEL}$  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さん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
  - $\delta$  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。

- 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切替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切替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-SP 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-SP 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-SP の接続が容易になります。詳細は 6.13 節をご覧ください。
- 新たなマクロボディとして、楕円柱 REC (Right Elliptical 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。

- 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 4.2.2 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さん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ください (11.2 節参照)。また、それに伴い、古いコンパイラ (f77, g77 など) ではコンパイルできなくなりました。詳しくは 2.4 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) 愛知淑徳大学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 6.14 節をご覧ください。
- <sup>35</sup>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 とリバモア評価済み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。

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  $\delta$  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  $\delta$  線を発生させます。

## 1.2 開発者

*PHITs* は以下の方々により開発が進められています。

(財) 高度情報科学技術研究機構 (RIST)

仁井田浩二

(独) 日本原子力研究開発機構 (JAEA)

松田規宏、橋本慎太郎、岩元洋介、佐藤達彦、古田琢哉、小川達彦、安部晋一郎、中島宏、深堀智生、奥村啓介、甲斐哲也

(共) 高エネルギー加速器研究機構 (KEK)

岩瀬広

(国) 東京工業大学 (TITech)

千葉敏

Chalmers University, Sweden

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、野田秀作

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, 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).

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。

<http://dx.doi.org/10.1080/00223131.2013.814553>

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).

## 2 インストール、コンパイル及び使用方法

*PHITS* は、WINDOWS, Mac 及び Linux 上で動作する FORTRAN プログラムです。WINDOWS 及び Mac に対しては、実行形式を含むインストーラを準備していますので、*PHITS* をコンパイルすることなく利用することができます。Linux に対しては実行形式を準備していませんので、`makefile` を用いて *PHITS* をコンパイルしてから利用する必要があります。また、WINDOWS や Mac でも、必要に応じて *PHITS* を再コンパイルすることができます。

### 2.1 動作環境

*PHITS* は、WINDOWS(XP 以降), Mac(OS X v10.6 以降), Linux, Unix など様々なコンピュータで動作しますが、快適に動作させるためには、メモリが 2GB 以上搭載されていることが望ましいです。また、*PHITS* をインストールするためには、約 4GB 以上のハードディスク空き容量が必要となります(推奨は 6GB 以上)。

*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 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/gview/index.htm>)

*PHITS* で使用するメモリ枠を拡張する場合 (2.9 参照) や、`usrsors.f` ファイルを使って線源を定義する場合 (4.3.14 参照) は、*PHITS* を再コンパイルする必要があります。事務局が奨励する *PHITS* 用コンパイラは Intel Fortran Compiler (11.1 以降) と `gfortran` (4.71 以降) です。それ以外のコンパイラでは、コンパイル時や実行時にエラーが発生する可能性が高いです。

### 2.2 Windows でのインストール及び実行方法

- (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) USB メモリもしくは DVD 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) `\phits\lecture\lec01\lec01.in` を右クリックして「送る」→「PHITS」
- (5) `xz_track_all.eps` が作成されたことを確認

メモリ共有型並列で実行する場合は `\phits\bin` フォルダにある `phits.bat` の 3 行目でコア数を指定してください。例えば 4 コアで計算する場合は、

```
set PHITS_PARALLEL=4
```

と設定してください。ただし、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。

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**、**angel.bat** と `\phits\dchain-sp\bin` フォルダにある **dchain.bat** のショートカットを **sendto** フォルダに作成する。
- (4) `\phits\data` フォルダにある核データリストファイル **xmdir.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。

## 2.3 Mac でのインストール及び実行方法

### インストール方法

USB メモリや DVD 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、インストール先のフォルダを指定すると、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。このフォルダには *PHITs* 本体とソース、講習会の資料、例題などがすべて入っています。

- (注 1) 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 2) 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、その名前を **phits[今日の日付].[現在の時刻]** に変更します。

### 使い方

インプットファイルを Dock にある青い PHITS アイコンにヘドラッグアンドドロップすることにより実行できます。計算結果はインプットファイルがあるフォルダに出力されます。

メモリ共有型並列で実行する場合は、次の手順で Dock にある PHITS の設定ファイルを書き換えてください。

- (1) DOC の PHITS アイコンを右クリックする。
- (2) “オプション”→“Finder に表示”を選択。( `/phits-office/phits/bin` が表示される。)
- (3) `bin` フォルダの PHITS アイコンを右クリックする。
- (4) “パッケージの内容を表示”を選択。
- (5) `Contents/document.wflow` をテキストエディタで開く。
- (6) 63 行目付近のシングルに対応する実行ファイルの行を次のようにコメントアウトする。

```
# phitsexec="/Users/phits-office/phits/bin/phits273_mac.exe"
```

- (7) その下の `openMP` に関する文章を次のように修正する。  
(`OMP_NUM_THREADS=`の後の数字で使用するコア数を指定してください。)



```
# If you would like to use OpenMP version, please use the following commands
phitsexec="/Users/phits-office/phits/bin/phits273_mac_openmp.exe"
# Please input your machine thread number when you use OpenMP version
export OMP_NUM_THREADS=4
```

(8) Contents/document.wflow を保存する。

## その他

ターミナルから *PHITs* を使う人は、ターミナルで以下を入力してください。

`/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/noda`

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

実行方法は、

```
phits250_mac.exe < your_input
```

となります。

## 2.4 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、`MPIFLAGS` や `OMPFLAGS` のある行の“#”を消して、これらを有効にしてください。なお、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。他に、コンパイル時に `-j` オプションを付けて `make` コマンドを実行するとコンパイルの途中で失敗しますのでご注意ください。これは、モジュールに関するオブジェクトファイルを先に作成する必要があるため、失敗した後、もう一度 `make` コマンドを実行するとうまくいきます。

Windows 用コンパイラとして `gfortran` をインストールする場合は、

- TDM-GCC (<http://tdm-gcc.tdragon.net/download>)

にアクセスし、“Bundle Installer” の欄にある最新版のインストーラーをダウンロードして行ってください。ダブルクリック等でこれを起動し、各 PC の環境にあわせた選択肢を選べばインストールが完了します。ただし、最初は“Create”を選択し、また、“New Installation: Choose Components”の段階では、“Select the type of install:”を“TDM-GCC Recommended, All Packages”に変更してください。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。

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。PHITS 事務局で推奨しているコンパイラは Intel Fortran Compiler (11.1 以降) と `gfortran` (4.71 以降) です。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下さい。なお、`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 事務局までご連絡下さい。

## 2.5 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ソリューションファイル (`bin.sln`) とプロジェクトファイル (`phits-intel.proj`) が含まれています。以下の手順に従って、これらのファイルを使って *PHITs* をコンパイル、実行できます。

- (1) `bin.sln` をダブルクリック (Visual Studio や Intel Fortran のバージョン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、`phits-intel.vfproj` ファイルの `Source files` にある一覧表で `a-angel.f` を `a-angel-winopenmp.f` に変更し、/Openmp オプションを追加してください。(Microsoft Visual Studio ウィ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、中に書かれている環境変数の `PHITS_EXE` を、例えば次のように、書き換えてください。

```
set PHITS_EXE=C:\phits\bin\Release\phits-intel.exe
```

実行ファイル名の方を `phits.bat` に書かれている名称 (例えば、`phits264.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ご注意ください。

## 2.6 ANGEL のコンパイル

*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。

## 2.7 実行シェル

*PHITs* を実行するための特別なシェルは必要ありません。*PHITs* のプログラムは、標準入力からインプットパラメータを読み込み、標準出力にサマリーとエラーメッセージを書き出します。その他の出力ファイルの指定は、インプットデータで行います。従って、最も簡単な実行コマンドは、

List 2.1 ● command line to execute *PHITs*

```
phits100 < input.dat > output.dat
```

となります。ここで、**phits100** は *PHITs* の実行ファイル、**input.dat** は *PHITs* の書式で書かれたインプットファイル、**output.dat** は標準出力の内容が書き出されるアウトプットファイルです。

WINDOWS 系の環境で実行する場合も同様です。ただし、WINDOWS 系では標準入力の **rewind** が使えないため、パラメータ **infl** を用いて **input.dat** 以外の外部ファイルを利用する場合はエラーとなります。そこで **infl** を使用する場合は、**input.dat** の 1 行目を

## List 2.2 ● the first line of the standard input

```
file = input.dat
```

としてください。このときプログラムは、**input.dat** というファイルを改めてオープンして、インプットデータを読み込みます。この方法は、WINDOWS 系以外でも使えます。**infl** の使い方については、3.3 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います。実行ディレクトリの **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.file
```

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。

## 2.8 実行の途中中断

プログラムを実行すると、カレントディレクトリに **batch.now** 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  $\times$  (**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。このファイルの 1 行目は、

```
1 <--- 1:continue, 0:stop
```

となっています。この最初の“1”を“0”に書きかえると、次のバッチで計算が終了し、そこまでのサマリー、タリーの出力をします。

これに関して、パラメータセクションに、新しいパラメータ **itall** を加えました。

```
itall = 2 # (D=0) 0:no tally at batch, 1:same, 2:different
```

のように指定します。**itall** = 1 では、バッチ毎に、メモリ分散型並列の場合には、バッチ数  $\times$  (**PE** - 1) 毎に、タリーの途中結果をユーザーが指定したファイルに、上書きで出力します。**itall** = 2 では、バッチ毎に、メモリ分散型並列の場合には、バッチ数  $\times$  (**PE** - 1) 毎に、タリーの途中結果をユーザーが指定したファイル名の後にバッチ番号をつけたファイルにその都度書き出します。いずれの場合も、最終結果はユーザーの指定したファイルに出力されます。

この二つの機能で途中経過を見ながら、計算を停止することが可能です。また、タリーの **epsout** のパラメータを使うと、途中結果をグラフでモニターすることができます。(5.7.16 を参照)

**batch.now** 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。

## 2.9 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、これは幾何形状とタリー、核データ関係、また、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、パラメータセクションで指定します。`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。

List 2.3 ● `param.inc`

```

1: *****
2: *
3: *      'param.inc'
4: *
5: *****
6:
7:      parameter ( mdas   =120000000 )
8:      parameter ( kvlmax =   3000 )
9:      parameter ( kvmmax = 1000000 )
10:     parameter ( itlmax =    60 )
11:     parameter ( inevt  =    70 )
12:     parameter ( isrc   =    50 )
13:     parameter ( latmax = 25000000 )
14:
15:     common /mdasa/ das( mdas )
16:     common /mdasb/ mmmmax
17:
18: *-----*
19: *
20: *      mdas   : total memory * 8 = byte
21: *      mmmmax : maximum number of total array
22: *
23: *      kvlmax : maximum number of regions, cell and material
24: *      kvmmax : maximum number of id for regions, cel and material
25: *
26: *      itlmax : number of maximum tally entry
27: *      inevt  : number of collision type for summary
28: *      isrc   : number of multi-source
29: *      latmax : maximum number of lattice in a cell + 1
30: *
31: *-----*
```

### 3 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。

#### 3.1 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。
[cell]	GG 体系を構成する cell を定義します。
[surface]	GG 体系を構成する面を定義します。
[transform]	GG 体系を構成する面の座標変換を定義します。
[importance]	領域の importance を定義します。
[weight window]	領域の weight window を定義します。
[volume]	体系を構成する領域の体積を定義します。
[temperature]	cell の温度を定義します。
[brems bias]	bremsstrahlung の bias を定義します。
[photon weight]	photon 生成の weight 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。
[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。
[counter]	counter を定義します。
[reg name]	領域表示の名前、大きさを指定します。
[mat name color]	物質表示の名前、色、大きさを指定します。
[mat time change]	物質の時間変化を定義します。
[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。
[timer]	粒子の固有時間をリセット、ストップを定義します。
[delta ray]	$\delta$ 線を発生させます。
[multiplier]	multiplier を定義します。

表 2: セクションの種類 (2)

name	説明
[t-track]	track length タリーのパラメータを定義します。
[t-cross]	面横断タリーのパラメータを定義します。
[t-heat]	発熱タリーのパラメータを定義します。
[t-deposit]	deposit タリーのパラメータを定義します。
[t-deposit2]	deposit2 タリーのパラメータを定義します。
[t-yield]	生成粒子タリーのパラメータを定義します。
[t-product]	生成粒子タリーのパラメータを定義します。
[t-dpa]	DPA タリーのパラメータを定義します。
[t-let]	LET タリーのパラメータを定義します。
[t-sed]	SED タリーのパラメータを定義します。
[t-time]	時間タリーのパラメータを定義します。
[t-star]	star density タリーのパラメータを定義します。
[t-dchain]	dchain タリーのパラメータを定義します。
[t-userdefined]	ユーザー定義タリーのパラメータを定義します。
[t-gshow]	領域境界表示のパラメータを定義します。
[t-rshow]	物理量領域表示のパラメータを定義します。
[t-3dshow]	3次元領域表示のパラメータを定義します。
[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。

### 3.2 読み込みコントロール

#### (1) 大文字、小文字、空白

ファイル名を除いて、大文字小文字は区別しません。行頭、行末の空白は無視します。セクション名は間に空白を入れてもかまいません。ただし、[\*\*\*\*\*] 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。

#### (2) タブ

タブは、8 個のブランクに置き換えます。

#### (3) 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。複数行が可能です。ただし、[cell]、[surface] の継続行は、先頭に 5 個以上の空白が必要です。

#### (4) 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
```

のようにです。但し、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ときは、使えません。

(5) コメント文字

コメント文字として、“#”, “%”, “!”, “\$” が定義されています。これらの文字以降行末まで無視されます。また、5 コラムまでの “c” で始まる少なくとも 1 個の空白を伴う行もコメント行となります。このため、[material] セクションで、“6000” すなわち炭素の自然同位体を定義するときは、“C” で始まるため、コメント行とされてしまいますので、“6000” で指定するようにしてください。

(注意: [cell], [surface] セクションの場合、セルの定義に “#” を用いますので、コメント文字として “\$” 以外は、このセクションでは利用できません。)

(6) 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。

(7) 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。

(8) 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。

(9) 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。

### 3.3 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 } [ n1 - n2 ]
```

です。{ } 内がファイル名、[ ] 内が行数指定です。n<sub>1</sub> 行から n<sub>2</sub> 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、

```
[ n1- ]
[ -n2 ]
```

という表現も可能です。上が n<sub>1</sub> 行から最後まで、下が先頭から n<sub>2</sub> 行までをインクルードします。インクルードは、何重にもネストできます。インクルードファイルを読み終わると、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、入力ファイルの 1 行目を

```
file = input.dat
```

としてください。ここで、*input.dat* は入力ファイルです。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、2.7 節をご参照ください。

### 3.4 ユーザ定義定数

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、

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

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。

### 3.5 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、

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		

例として、

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

この例のように、*param =* の後に 1 個の数字が期待される場合は、数式の途中で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。



### 3.6 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。粒子タイプ 20 番の nucleus は、全ての原子核から 15 番から 18 番、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	ityp	symbol	kf-code	particle name
1	proton	2212	proton	11	-	+ - 12	$\nu_e \bar{\nu}_e$
2	neutron	2112	neutron	11	-	+ - 14	$\nu_\mu \bar{\nu}_\mu$
3	pion+	211	$\pi^+$	11	-	-2212	$\bar{p}$
4	pion0	111	$\pi^0$	11	-	-2112	$\bar{n}$
5	pion-	-211	$\pi^-$	11	-	-311	$\bar{K}^0$
6	muon+	-13	$\mu^+$	11	-	+ - 221	$\eta \bar{\eta}$
7	muon-	13	$\mu^-$	11	-	331	$\eta'$
8	kaon+	321	$K^+$	11	-	+ - 3122	$\Lambda^0 \bar{\Lambda}^0$
9	kaon0	311	$K^0$	11	-	+ - 3222	$\Sigma^+ \bar{\Sigma}^+$
10	kaon-	-321	$K^-$	11	-	+ - 3212	$\Sigma^0 \bar{\Sigma}^0$
11	other	below	other particle	11	-	+ - 3112	$\Sigma^- \bar{\Sigma}^-$
12	electron	11	$e^-$	11	-	+ - 3322	$\Xi^0 \bar{\Xi}^0$
13	positron	-11	$e^+$	11	-	+ - 3312	$\Xi^- \bar{\Xi}^-$
14	photon	22	$\gamma$	11	-	+ - 3334	$\Omega^- \bar{\Omega}^-$
15	deuteron	1000002	deuteron				
16	triton	1000003	triton				
17	3he	2000003	$^3\text{He}$				
18	alpha	2000004	$\alpha$				
19	nucleus	Z*1000000+A	nucleus				
20	all	-	all particles				

表 5: 崩壊チャンネルと寿命

	分岐率	寿命 (sec)
$\pi^0 \rightarrow \gamma + \gamma$	100%	0
$\pi^+ \rightarrow \mu^+ + \nu_\mu$	100%	2.6029e-8*
$\pi^- \rightarrow \mu^- + \nu_\mu$	100%	2.6029e-8
$\mu^+ \rightarrow e^+ + \bar{\nu}_e + \nu_\mu$	100%	2.19703e-6
$\mu^- \rightarrow e^- + \bar{\nu}_e + \nu_\mu$	100%	2.19703e-6
$K^0 \rightarrow \pi^+ + \pi^-$	68.61%	8.922e-11
$\rightarrow \pi^0 + \pi^0$	31.39%	
$\rightarrow \gamma + \gamma$	other	
$K^+ \rightarrow \mu^+ + \nu_\mu$	63.51%	1.2371e-8
$\rightarrow \pi^+ + \pi^-$	other	
$K^- \rightarrow \mu^- + \nu_\mu$	63.51%	1.2371e-8
$\rightarrow \pi^+ + \pi^-$	other	
$\eta \rightarrow \gamma + \gamma$	38.9%	0
$\rightarrow \pi^0 + \pi^0 + \pi^0$	31.9%	
$\rightarrow \pi^+ + \pi^- + \pi^0$	23.7%	
$\rightarrow \pi^+ + \pi^- + \gamma$	other	
$\eta' \rightarrow \pi^+ + \pi^- + \eta$	44.1%	0
$\rightarrow \pi^0 + \pi^0 + \eta$	20.5%	
$\rightarrow \pi^+ + \pi^- + \gamma$	30.1%	
$\rightarrow \gamma + \gamma$	other	
$\Lambda^0 \rightarrow p + \pi^-$	64.1%	2.631e-10
$\rightarrow n + \pi^0$	other	
$\Sigma^+ \rightarrow p + \pi^0$	51.57%	7.99e-11
$\rightarrow n + \pi^+$	other	
$\Sigma^0 \rightarrow \Lambda^0 + \gamma$	100%	0
$\Sigma^- \rightarrow n + \pi^-$	100%	1.479e-10
$\Xi^0 \rightarrow \Lambda^0 + \pi^0$	100%	2.90e-10
$\Xi^- \rightarrow \Lambda^0 + \pi^-$	100%	1.639e-10
$\Omega^- \rightarrow \Lambda^0 + K^-$	67.8%	8.22e-11
$\rightarrow \Xi^0 + \pi^-$	23.6%	
$\rightarrow \Xi^- + \pi^0$	other	

\*2.6029x10<sup>-8</sup>

## 4 セクション書式

### 4.1 [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.  
.....
```

## 4.2 [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。

```
[ Parameters ]
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。

### 4.2.1 計算モード

表 6: パラメータ 1

パラメータ	値	説明
icntl	(D=0)	基本動作オプション
	= 0	normal <i>PHITs</i> 計算
	= 1	nuclear reaction 計算、 未完成
	= 2	CGVIEW インputファイル書き出し
	= 3	input echo だけの書き出し メモリー空間、ライブラリー、ファイルのチェック
	= 4	MARS-PF インputファイル書き出し
	= 5	no reaction, no ionization, 全てを void にする。 geometry check, 体積、面積計算に使う
	= 6	ソースのチェック。[t-product] でソース粒子をタリーできます
	= 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) でファイルを指定	

**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。  
**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put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。この **dumpall** オプションは、**GG** のときのみに有効です。**dumpall=1** でダンプする情報は、4.2.20 のところで解説します。

## 4.2.2 ヒストリー数、バンク配列の大きさ

表 7: パラメータ 2

パラメータ	値	説明
<b>irskip</b>	(D=0) irskip>0 irskip<0	乱数のコントロール 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 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)
<b>rseed</b>	(D=0.0) rseed<0 rseed=0 rseed>0	初期乱数オプション 計算開始時間情報より初期乱数を設定 6647299061401. デフォルトの初期乱数 rseed を初期乱数とする
<b>maxcas</b> <b>maxbch</b>	(D=10) (D=10)	1 バッチのヒストリー数。上限は 2147483647。 バッチ数。上限は 2147483647。
<b>maxbnk</b>	(D=10000)	バンク配列の大きさ
<b>istdev</b>	(D=0) =-2 =-1 =0 =1 =2	新規 / 再開計算及び統計誤差導出方法のコントロール。 再開計算 (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。 再開計算 (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。 新規計算。統計誤差は、メモリ共有型並列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。 新規計算。統計誤差は、各バッチ間の分散により導出。 新規計算。統計誤差は、各ヒストリー間の分散より導出。
<b>ireschk</b>	(D=0) =0 =1	再開計算時のタリーパラメータ一致性確認の有無。 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。 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。従って、入力バッチ数 **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。

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より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より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  $\sigma$  を計算し、その値とタリー量の平均値  $\bar{X}$  の比を相対誤差として出力します。

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

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、 $\bar{w}$  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勧めします。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誤差の範囲内に収まるわけではないのでご注意ください。また、核反応モデルに起因する系統誤差は評価されていないのでご注意ください。

**istdev<0** とした場合は、再開計算となります。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計算も可能です。再開計算の流れは、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。

### 4.2.3 計算打切エネルギー、切り替えエネルギー

表 8: パラメータ 3

パラメータ	値	説明
<b>emin(1)</b>	(D=1.0)	陽子の計算打切エネルギー (MeV)
<b>emin(2)</b>	(D=1.0)	中性子の計算打切エネルギー (MeV)
<b>emin(i)</b>	(D=1.0)	<b>i</b> -th 粒子の計算打切エネルギー (MeV)
	<b>i = 3-10</b>	粒子番号は、表を参照
<b>emin(11)</b>	(D=2.0)	その他の粒子の計算打切エネルギー (MeV)
<b>emin(i)</b>	(D=1.e+9)	<b>i</b> -th 粒子の計算打切エネルギー (MeV/u)
	<b>i = 12-19</b>	粒子番号は、表を参照
	<b>i = 15-19</b>	これらの粒子のエネルギーは、核子当たり
<b>esmin</b>	(D=0.001)	荷電粒子の <b>range</b> 計算の最小エネルギー (MeV)
<b>esmax</b>	(D=300000)	荷電粒子の <b>range</b> 計算の最大エネルギー (MeV)
<b>cmin(i)</b>	(D= <b>emin(i)</b> )	<b>i</b> -th 粒子の核反応計算打切エネルギー (MeV/u)
		これ以下のエネルギーの核反応は無視する。
	<b>i = 15-19</b>	これらの粒子のエネルギーは、核子当たり
<b>dmax(i)</b>	(D= <b>emin(i)</b> )	<b>i</b> -th 粒子のライブラリー利用の上限エネルギー

これらのパラメータで各々のエネルギー領域に制限を与える際は、下限の場合はその値以上、上限の場合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の上限値を与えた場合、丁度その値のエネルギー (デフォルトだと 1.0MeV) をもつ陽子は計算打切とはなりません。

$emin < energy < dmax$  の範囲がライブラリーによる計算になります。 $emin \geq dmax$  とすれば、ライブラリーを用いた計算をしません。陽子、中性子、光子、電子のライブラリーの上限は、現在のところ、それぞれ 3GeV、3GeV、100GeV、10GeV です。

荷電粒子の **range** 計算において、 $esmin < energy < esmax$  の範囲で **range** 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設定して下さい。通常はデフォルトのまま計算を行います。

表 9: パラメータ 4

パラメータ	値	説明
<i>ejamnu</i>	(D=20.)	核子の Bertini モデル (もしくは JQMD) から JAM への切り替えエネルギー (MeV/u)
<i>ejampi</i>	(D=20.)	パイオンの Bertini モデルから JAM への切り替えエネルギー (MeV)
<i>eisobar</i>	(D=0.0)	isobar=1 の時の isobar モデルの上限エネルギー (MeV)
<i>eqmdnu</i>	(D=20.)	核子の Bertini モデルから JQMD への切り替えエネルギー (MeV)
<i>eqmdmin</i>	(D=10.0)	JQMD 適用の下限エネルギー (MeV/u)
<i>ejamqmd</i>	(D=3500.0)	原子核反応の JQMD から JAMQMD への切り替えエネルギー (MeV/u)
<i>inclg</i>	(D=1) =0 =1 =2	核反応モデルとして INCL を使用する。 INCL を使用しない。 陽子、中性子、パイオン、軽イオン ( $d, t, {}^3\text{He}, \alpha$ ) が関与する核反応で INCL を使用する。 陽子、中性子、パイオンが関与する核反応で INCL を使用する。
<i>einclmin</i>	(D=1.0)	INCL が適用されるの下限のエネルギー (MeV/u)。
<i>einclmax</i>	(D=3000.0)	INCL が適用されるの上限のエネルギー (MeV/u)。
<i>incelf</i>	(D=0) =0 =1	核子が関与する核反応モデルとして INC-ELF を使用する。 INC-ELF を使用しない。 INC-ELF を使用する。
<i>eielfmin</i>	(D=1.0)	INC-ELF が適用されるの下限のエネルギー (MeV)。
<i>eielfmax</i>	(D=3500.0)	INC-ELF が適用されるの上限のエネルギー (MeV)。
<i>irqmd</i>	(D=0) =0 =1	核反応モデルとして JQMD か R-JQMD を使用する。 JQMD を使用する。 R-JQMD を使用する。

$d, t, \alpha$  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、通常物質中だ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 3.5GeV/u 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<sup>3</sup>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<sup>4</sup>を引用するようにしてください。

JQMD, R-JQMD 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 2.7 以降は、それまで使われていた JQMD に加えて、R-JQMD も選択できるようになりました。R-JQMD 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。

<sup>3</sup> A. Boudard, J. Cugnon, J.-C. David, S. Leray, and D. Mancusi, Phys. Rev C87, 014606 (2013).

<sup>4</sup> Y. Sawada, Y. Uozumi, S. Nogamine, T. Yamada, Y. Iwamoto, T. Sato, and K. Niita, Nucl. Instr. & Meth. B 291, 38-44 (2012).



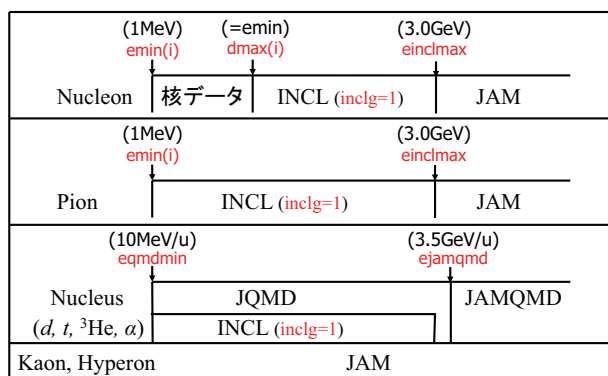


図 1: 核反応モデルの切り替えエネルギー

## 4.2.4 時間カット、ウエイトカット、ウエイトウインドウ

表 10: パラメータ 5

パラメータ	値	説明
tmax(i)	(D=1.e+9) i = 1-20	i-th 粒子の計算打切時間 (nsec) 粒子番号は、表を参照
wc1(i)	(D=-0.5)	i-th 粒子の minimum weight
wc2(i)	(D=wc1/2)	i-th 粒子の cutoff weight
swtm(i)	(D=1.0)	i-th 粒子の minimum source weight
wupn	(D=5)	ウエイトウインドウの上限値 = [Weight Window] セクションで指定した下限値 × wupn wupn ≥ 2
wsurv	(0.6*wupn)	サバイバルウエイト値 1 < wsurv < 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

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、 $|WC1| \times SWTM$  または、 $|WC2| \times SWTM$  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。

## 4.2.5 計算モデルオプション (1)

表 11: パラメータ 6

パラメータ	値	説明
<b>ielas</b>	(D=2) = 0 = 1 = 2	弾性散乱オプション 弾性散乱を考慮しない 中性子の弾性散乱を考慮する 中性子と陽子の弾性散乱を考慮する
<b>ielms</b>	(D=100)	弾性散乱角度分布分点数
<b>inmed</b>	(D=1) = 0 = 1 = 2	Bertini モデルの核子核子断面積オプション free (nmtclk25.dat) Cugnon old (nmtclk95.dat) Cugnon new (nmtclk30.dat)
<b>nevap</b>	(D=3) = 0 = 1 = 2 = 3	蒸発モデルのオプション 蒸発モデルを用いない DRES モデルを用いる SDM モデルを用いる GEM モデルを用いる
<b>ismm</b>	(D=0) = 0 = 1	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 を使う場合、JQMD から GEM への切り替え時間がデフォルトの 100fm/c から 75fm/c に変わります。
<b>igamma</b>	(D=0) = 0 = 1 = 2 = 3	残留核の $\gamma$ 崩壊オプション = 0 $\gamma$ 崩壊を考慮しない = 1 $\gamma$ 崩壊を考慮する。 = 2 $\gamma$ 崩壊を EBITEM モデルを用いて考慮する。 = 3 $\gamma$ 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 <code>file(14) = trxcrd.dat</code> を [parameters] セクションで設定する必要があります。
<b>isobar</b>	(D=0) = 0 = 1	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

**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。

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 100MeV/u 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<sup>5,6</sup>をご覧ください。

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<sup>7</sup>をご覧ください。

<sup>5</sup> T. Ogawa, T. Sato, S. Hashimoto, and K. Niita, Nuclear Instruments and Methods in Physics Research A 723 (2013) 36-46.

<sup>6</sup> J.P. Bondorf, A.S. Botvina, A.S. Iljinov, I.N. Mishustin, and K. Snepken, Physics Reports 257 (1995) 133-221.

<sup>7</sup> T. Ogawa, S. Hashimoto, T. Sato, and K. Niita, Nuclear Instruments and Methods in Physics Research B 325 (2014) 35-42.

## 4.2.6 計算モデルオプション (2)

表 12: パラメータ 7

パラメータ	値	説明
ipreeq	(D=0) = 0 = 1	nevap=1 の時の preequ モデルのオプション preequ モデルを用いない preequ モデルを用いる
level	(D=3) = 1 = 2 = 3	nevap=1 の時の 状態密度のオプション 8/A 馬場のパラメータ Ignatyuk のパラメータ
ieleh	(D=0) = 0 = 1	電子、陽電子の輸送オプション dmax(12) 以上のエネルギーでは、減速、反応を行わない。 dmax(12) 以上で $e=dmax(12)$ , $weight=e/dmax(12)$ とする。
ipnint	(D=0) = 0 = 1	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。) 光核反応を考慮しない。 光核反応を考慮する。
imucap	(D=0) = 0 = 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。 考慮しない。 考慮する。
imuint	(D=0) = 0 = 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。 考慮しない。 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。
npidk	(D=0) = 0 = 1	計算打切エネルギーに達した負電荷の崩壊粒子の取り扱い 吸収反応を強制的に考慮する 崩壊させる

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。  
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。その中で負電荷を持つ粒子は、npidk=0 の場合は、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。

## 4.2.7 計算モデルオプション (3)

表 13: パラメータ 8

パラメータ	値	説明
<b>imagnf</b>	(D=0) = 0 = 1	磁場のオプション 磁場を考慮しない 磁場を考慮する
<b>ielctf</b>	(D=0) = 0 = 1	電磁混合場のオプション 電磁混合場を考慮しない 電磁混合場を考慮する
<b>andit</b>	(D=0) = 0 = 1 = 2	Bertini モデルの $\Delta$ 角分布オプション 50%等方、50%前方 全て等方分布 全て前方分布
<b>icxsni</b>	(D=0) = 0 = 1 = 2	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 Pearlstein-Niita の式 KUROTAMA 模型 Sato の式
<b>icrhi</b>	(D=1) = 0 = 1 = 2	原子核・原子核反応の全断面積オプション Shen の式 NASA の式 KUROTAMA 模型
<b>idwba</b>	(D=0) = 0 = 1	DWBA スペクトルのオプション DWBA 計算で求めた離散スペクトルを考慮しない DWBA 計算で求めた離散スペクトルを考慮する
<b>gravx</b>	(D=0)	重力場の方向ベクトルの x 成分
<b>gravy</b>	(D=0)	重力場の方向ベクトルの y 成分
<b>gravz</b>	(D=0)	重力場の方向ベクトルの z 成分

KUROTAMA 模型は、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模型です。詳細はこの文献<sup>8</sup>をご覧ください。また、本模型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、必ず下記文献をご引用ください。

**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  ${}^7\text{Li}(p, n){}^7\text{Be}$  反応

10–50 MeV における  ${}^9\text{Be}(p, n){}^9\text{B}$  反応

10–50 MeV における  ${}^{6,7}\text{Li}(d, n){}^{7,8}\text{Be}$  および  ${}^{6,7}\text{Li}(d, p){}^{7,8}\text{Li}$  反応

5–25 MeV における  ${}^9\text{Be}(d, n){}^{10}\text{B}$  および  ${}^9\text{Be}(d, p){}^{10}\text{Be}$  反応

10–50 MeV における  ${}^{12,13}\text{C}(d, n){}^{13,14}\text{N}$  および  ${}^{12,13}\text{C}(d, p){}^{13,14}\text{C}$  反応

**gravx**, **gravy**, **gravz** は、重力場の方向ベクトル。重力を設定したときは、1eV 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。

<sup>8</sup> K. Iida, A. Kohama, and K. Oyamatsu, J. Phys. Soc. Japan 76, 044201 (2007).

## 4.2.8 計算モデルオプション (4)

表 14: パラメータ 9

パラメータ	値	説明
ndedx	(D=2)	荷電粒子の $dE/dx$ オプション
	= 0	重イオンは SPAR、他は NMTC オリジナル
	= 1	重イオン、陽子は ATIMA、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
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)
nspred	(D=0)	荷電粒子のクーロン散乱 (angle straggling) オプション
	= 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	クーロン散乱を考慮する。Moliere 理論に基づく Lynch の式 <sup>9</sup>
	= 10	クーロン散乱を考慮する。ATIMA の計算式
ascatl	(D=13.6)	Lynch の式の $S_2$ パラメータ。nspred = 2 の場合のみ有効
ascat2	(D=0.088)	Lynch の式の $\epsilon$ パラメータ。nspred = 2 の場合のみ有効
nedisp	(D=0)	荷電粒子の減速過程でのエネルギー分散 (energy straggling) のオプション
	= 0	エネルギー分散を考慮しない
	= 1	エネルギー分散を考慮する。Landau Vavilov
	= 10	エネルギー分散を考慮する。ATIMA の時
e-mode	(D=0)	Event generator mode オプション。
	= 0	通常モード。
	= 1	Event generator mode Ver.1.
	= 2	Event generator mode Ver.2. 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。
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

ndedx は、PHITS ver.2.00 以前は、ndedx=0 がデフォルトでしたので注意してください。

ATIMA のオプションは、テスト中です。また、計算時間が相当かかりますので注意してください。

e-mode オプションの詳細は、4.2.21 をご覧ください。ただし、Event generator mode Ver.2 は、メモリ共有型並列 (OpenMP) に一部未対応です。ご使用の際はご注意ください。

<sup>9</sup>  $\sigma = S_2 \frac{\sqrt{X/X_0}}{p\beta} [1 + \epsilon \log_{10}(X/X_0)]$ : G.R. Lynch and O.I. Dahl, Nucl. Instrum. Methods Phys. Res, B 58, 6-10 (1991) の式 (4).

## 4.2.9 出力オプション (1)

表 15: パラメータ 10

パラメータ	値	説明
<b>infout</b>	(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
<b>nrecover</b>	(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。

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)  
ウェイトを変化させる 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ェイト
- (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: 核反応モデルによる直接過程・前平衡過程計算およびライブラリーによる反応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。



## 4.2.10 出力オプション (2)

表 16: パラメータ 11

パラメータ	値	説明
<b>incut</b>	(D=0) = 0 = 1 = 2	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 出力なし ファイル名 <b>file(12)</b> の <b>ncut</b> ファイルに出力 <b>file(12)</b> に時間の情報も出力
<b>igcut</b>	(D=0) = 0 = 1 = 2 = 3	計算打切エネルギー以下の $\gamma$ 線、電子情報の出力オプション 出力なし $\gamma$ 線情報をファイル名 <b>file(13)</b> の <b>gcut</b> ファイルに出力 <b>file(13)</b> に $\gamma$ 線の時間の情報も出力 <b>file(13)</b> に $\gamma$ 線、電子、陽電子の情報も出力
<b>ipcut</b>	(D=0) = 0 = 1 = 2	計算打切エネルギー以下の陽子情報の出力オプション 出力なし ファイル名 <b>file(10)</b> の <b>pcut</b> ファイルに出力 <b>file(10)</b> に時間の情報も出力
<b>inpara</b>	(D=0) = 0 = 1 = 3	メモリ分散型並列計算の時、 <b>ncut</b> ファイル名のオプション /wk/uname/ファイル名 <b>file(12)</b> = 1 /wk/uname/ファイル名 <b>file(12)</b> の後に PE 番号を付加 = 3 ファイル名 <b>file(12)</b> の後に PE 番号を付加
<b>igpara</b>	(D=0) = 0 = 1 = 3	メモリ分散型並列計算の時、 <b>gcut</b> ファイル名のオプション /wk/uname/ファイル名 <b>file(13)</b> = 1 /wk/uname/ファイル名 <b>file(13)</b> の後に PE 番号を付加 = 3 ファイル名 <b>file(13)</b> の後に PE 番号を付加
<b>ippara</b>	(D=0) = 0 = 1 = 3	メモリ分散型並列計算の時、 <b>pcut</b> ファイル名のオプション /wk/uname/ファイル名 <b>file(10)</b> そのまま = 1 /wk/uname/ファイル名 <b>file(10)</b> の後に PE 番号を付加 = 3 ファイル名 <b>file(10)</b> の後に PE 番号を付加
	/wk/uname/	/wk/ はデフォルトのディレクトリ名 uname は環境変数 LOGNAME から読み込まれるユーザー名

**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。

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**=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** = 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。

## 4.2.11 出力オプション (3)

表 17: パラメータ 12

パラメータ	値	説明
<b>itall</b>	(D=0) = 0 = 1 = 2	タリーの出力をバッチ毎にするオプション 出力なし 同じファイルに出力 異なるファイルに出力 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
<b>itstep</b>	(D=0) = 0 = 1	magnetic field などの粒子の運動量が増加する場でのタリーを取るオプション 境界、核反応までタリーを取らない ステップごとにタリーを取る
<b>imout</b>	(D=0) = 0 = 1 = 2	[material] セクションでの物質表示オプション mat[12], 208Pb.33c タイプ mat[12], Pb-208.33c タイプ m12, 82208.33c MCNP 入力タイプ
<b>jmout</b>	(D=0) = 0 = 1	[material] セクションでの物質密度表示オプション 入力のまま 粒子密度に変換する
<b>kmout</b>	(D=0) = 0 = 1	核データ物質情報表示オプション 表示無し input echo に表示する
<b>matadd</b>	(D=1) = 0 = 1	GG の際、同じ物質で密度が異なる時のオプション 同じ物質番号で扱う 新たな物質番号を付加する

磁場など粒子の運動量を増加させる場のある時、タリーは通常境界面を横断するときと、核反応を起こすタイミングでしか呼ばれないので、磁場中の軌跡を track タリーで表示すると境界間は直線でしか表現できませんでした。そこで、バーチャルな境界面などを設定して表現していたのですが、この **itstep** のオプションを用いることにより、運動の各ステップごとにタリーを呼び、曲線の軌跡を追うことができるようになりました。磁場のある場合のステップの最大値は、**deltm** で指定します。

## 4.2.12 出力オプション (4)

表 18: パラメータ 13

パラメータ	値	説明
iggcm	(D=0) = 0 = 1	GG の warning 等のメッセージをエコーする エコーしない インプットエコー部分に書き出す
ivout	(D=0) = 0	input echo での volume 表示のオプション [volume] セクションに書き出す
ipout	(D=1) (D=0 for GG) = 0	input echo での importance 表示のオプション [importance] セクションに書き出す 但し、全粒子に同じ値が定義されているときに限る
icput	(D=0) = 0 = 1	CPU 時間のカウントオプション カウント無し カウント有り
ipara	(D=0) = 0 = 1	パラメータ表示オプション 入力のみを表示 全てを表示
nwsors	(D=0)	nwsors 個のソース情報を file(6) に書き出す

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、デフォルトで入力が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。

## 4.2.13 出力オプション (5)

表 19: パラメータ 14

パラメータ	値	説明
<b>ivoxel</b>	(D=0) =0 =1 =2	ボクセルデータのバイナリ形式による出力・読み込み 出力なし ファイル名 <b>file(18)</b> からバイナリ形式のボクセルデータを読み込む ファイル名 <b>file(18)</b> にバイナリ形式のボクセルデータを出力する
<b>dumpall</b>	(D=0) = 0 = 1 = -1	全ての情報のダンプオプション 出力なし ファイル名 <b>file(15)</b> の <b>dumpall</b> ファイルにバイナリ形式でデータを出力 ファイル名 <b>file(15)</b> の <b>dumpall</b> ファイルにアスキー形式でデータを出力
<b>idpara</b>	(D=3) = 0 = 1 = 3	メモリ分散型並列計算の時、 <b>dumpall</b> ファイル名のオプション /wk/uname/ファイル名 <b>file(15)</b> /wk/uname/ファイル名 <b>file(15)</b> の後に PE 番号を付加 ファイル名 <b>file(15)</b> の後に PE 番号を付加
	/wk/uname/	/wk/ はデフォルトのディレクトリ名 uname は環境変数 LOGNAME から読み込まれるユーザー名

**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、読み取った入力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となります。**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。

**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、4.2.20 のところで解説します。

メモリ分散型並列計算時は PE (Processer Element) 毎に別のファイルを用意して書き出し・読み込みを行います。**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。**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データを書き出し、そのファイルからのみデータの読み込みを行います。

## 4.2.14 幾何形状のエラー関係

表 20: パラメータ 15

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。
deltb	(D=1.e-5)	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)。
deltc	(D=2.012345)	エネルギー分散を考慮した場合 (nedisp=1) の荷電粒子の最大飛行メッシュ(cm)。
delt0	(D=0.1)	クーロン多重散乱を考慮した場合 (nspred=1) や磁場を考慮した場合 (imagnf=1) の最小飛行メッシュ(cm)。ただし、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=1.0)	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)。

igchk のデフォルトを変えましたので (計算時間短縮のため)、新しい形状をテストする時には、igchk=1 として下さい。deltb は、端数を入れないと GG でエラーの可能性があるのでデフォルトを変更しました。

## 4.2.15 入出力ファイル名

表 21: パラメータ 16

パラメータ	値	説明
file(2)	(D=cgview.in)	icntl=2 の時、EGS4 粒子飛跡表示プログラム CGVIEW 用出力ファイル名
file(3)	(D=cgview.set)	icntl=2 の時、CGVIEW 用セットアップ情報ファイル名 この部分が CGVIEW のセットアップになる
file(4)	(D=marspf.in)	icntl=4 の時、MARS-PF 用出力ファイル名
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=c:/phits/data/xsdir.jnd)	断面積ディレクトリファイル名
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 $\gamma$ 線情報の出力ファイル名
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(15)	(D=dumpall.dat)	dumpall=1 を指定した時、情報の出力ファイル名
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(20)	(D=c:/phits/XS/egs/)	EGS 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=c:/phits/dchain-sp/data/)	DCHAIN-SP 用データを格納したフォルダ名
dircha	(D=0) =0 =1	ディレクトリ文字 "/" "backslash"

file(7) は、他と同様にフルパス名で書いてください。

dircha は、Windows での運用を想定したパラメータです。UNIX 系では、ディレクトリ文字が '/' ですが、Windows 系では '\' なので、その違いを吸収します。ただし、WindowsXP では、'/' のままでも動いています。

バージョン 2.74 より、 $igamma \geq 1$  のときでも、 $\gamma$  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。

## 4.2.16 その他

表 22: パラメータ 17

パラメータ	値	説明
<b>inucr</b>	(D=1) = 1 = 2 = 3 = 4 = 5 = 6	核反応計算のオプション 二重微分断面積計算 全、弾性、弾性外断面積出力 シミュレーションによる弾性外断面積計算 弾性散乱の角分布計算 pp, np, $\pi$ -p, 断面積出力 シミュレーションによる pp, np, $\pi$ -p, 断面積計算
<b>idam(i)</b> <b>rdam(i)</b>	integer real*8 <b>i = 1 - 100</b>	ユーザー定義の整数変数 ユーザー定義の実数変数 これらは、各プログラムの中で <b>common /userp/ idam(100), rdam(100)</b> を引用すれば用いることができる

**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。

## 4.2.17 低エネルギー中性子の物理パラメータ

以下のパラメータは、20MeV 以下の中性子の計算に用います。

表 23: パラメータ 18

パラメータ	値	説明
<b>emcnf</b>	(D=0.0)	中性子捕獲のしきい値 (MeV) これ以上で <b>implicit capture</b> 、以下で <b>analog capture</b>
<b>iunr</b>	(D=0)	現在、0 で固定
<b>dnb</b>	(D=-1) =-1 = 0 > 0	核分裂からの delayed neutron 数 natural sampling no delayed neutron neutron number

## 4.2.18 光子・電子輸送オリジナルモデルの物理パラメータ

以下のパラメータは、`negs=0` のときのみ有効となります。

表 24: パラメータ 19

パラメータ	値	説明
<code>emcpf</code>	(D=100)	photon の詳細モデルの上限 (MeV)
<code>ides</code>	(D=0) = 0 = 1	photon の electron 生成のオプション electron を生成、もしくは、brems.photon の生成 electron を生成しない
<code>nocoh</code>	(D=0) = 0 = 1	photon の coherent scattering のオプション coherent scattering を起こす coherent scattering を起こさない
<code>iphot</code>	(D=0) = 0 = 1	electron の photon 生成のオプション photon を生成 photon を生成しない
<code>ibad</code>	(D=0) = 0 = 1	brems. の角分布のオプション full brems. tabular angular distribution simple brems. angular distribution approximation
<code>istrg</code>	(D=0) = 0 = 1	straggling のオプション sampled straggling for electron energy loss expected-value straggling for electron energy loss
<code>bnum</code>	(D=1) = 0 > 0	brems. photon のオプション brems. photon を生成しない analog brems. photon 数
<code>xnum</code>	(D=1) = 0 > 0	x-ray photon のオプション x-ray photon を生成しない analog x-ray photon 数
<code>enum</code>	(D=1) = 0 > 0	secondary electron のオプション secondary electron を生成しない analog secondary electron 数
<code>numb</code>	(D=0) = 0 > 0	brems. process のオプション nominal brems. production produce brems. on each substep



## 4.2.19 EGS5 用パラメータ

表 25: パラメータ 20

パラメータ	値	説明
<b>negs</b>	(D=0) = 0 = 1	光子・電子・陽電子の輸送に関するオプション オリジナルモデルを使う (EGS5 を使用しない)。この場合、表 24 のパラメータが有効になります。 EGS5 を使う。この場合、file(20) の指定が必要となります。
<b>iegsout</b>	(D=0) = 0 = 1 = 2	EGS5 用出力ファイルオプション 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。 再読込するファイル (pegs5.inp, pegs5.dat, pegs5.msfit) のみ残す 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 <sup>10</sup> をご参照ください。
<b>iegsrand</b>	(D=-1) < 0 = 0 > 0	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 並列計算や再開計算ができなくなることにご注意ください。 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) EGS5 のデフォルト乱数初期値 (314159265) を使う 指定した値を EGS5 乱数初期値として使う
<b>iedgfl</b>	(D=1) = 0 = 1	K 及び L 殻特性 X 線に関するオプション K 及び L 殻特性 X 線を扱わない K 及び L 殻特性 X 線を扱う
<b>iauger</b>	(D=1) = 0 = 1	K 及び L 殻オーージェ電子に関するオプション K 及び L 殻オーージェ電子を扱わない K 及び L 殻オーージェ電子を扱う
<b>iraylr</b>	(D=1) = 0 = 1	レイリー散乱 (Rayleigh scattering) に関するオプション レイリー散乱を扱わない レイリー散乱を扱う
<b>lpolar</b>	(D=0) = 0 = 1	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) 直線偏光を扱わない 直線偏光を扱う

EGS5 を使う場合、定義できる物質数が 100 に制限されます。また、1 つの物質で使用できる元素の数が 50 に制限されます。cygwin をお使いの場合など、コンピュータ環境によっては、メモリーオーバーのエラーがでる可能性があります。その場合は、src/include/egs5.h.f に書かれた物質数の最大値 (MXMED パラメータ) を小さくして再コンパイルしてください。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  $\text{emin}(12,13)=0.1$ ,  $\text{emin}(14)=0.001$ ,  $\text{dmax}(12-14)=1000.0$  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。

[t-track] タリーや [t-cross] タリーにおいて 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場合があります。EGS5 特有の輸送アルゴリズムによるもので異常ではありません。その場合はエネルギーに関するメッシュ幅を大きくしてください。

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。iprint=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。

<sup>10</sup> H. Hirayama *et al.*, SLAC-R-730 (2005) and KEK Report 2005-8 (2005).

表 26: パラメータ 21

パラメータ	値	説明
<b>incohr</b>	(D=1) = 0 = 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 非干渉散乱関数を使わない (自由電子との散乱) 非干渉散乱関数を使う (束縛電子との散乱) ( <b>ibound</b> が自動的に 1 に固定されます)
<b>iprofr</b>	(D=1) = 0 = 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 ドップラー効果を考慮しない ドップラー効果を考慮する ( <b>incohr</b> と <b>ibound</b> が自動的に 1 に固定されます)
<b>impacr</b>	(D=1) = 0 = 1	EII (Electron impact ionization) に関するオプション EII を考慮しない EII を考慮する
<b>ieispl</b>	(D=0) = 0 = 1	EII により発生する光子の扱いに関するオプション 分割しない (no splitting) 分割する (splitting)
<b>neispl</b>	(D=0)	<b>ieispl=1</b> のときに分割する光子の数
<b>ibrdst</b>	(D=1) = 0 = 1	制動放射の発生角度に関するオプション デフォルト値 サンプリング
<b>iprdst</b>	(D=1) = 0 = 1	電子対生成の発生角度に関するオプション デフォルト値 サンプリング
<b>iphter</b>	(D=1) = 0 = 1	光電子の角度分布に関するオプション デフォルト値 (入射光子の方向) サンプリング
<b>ibound</b>	(D=1) = 0 = 1	コンプトン断面積に関するオプション 自由コンプトン断面積 (free Compton cross section) 束縛全コンプトン断面積 (bound total Compton cross section)
<b>iaprim</b>	(D=1) = 0 = 1 = 2	制動放射断面積に関するオプション Motz <i>et al.</i> の経験式を使う ICRU-37 で与えられた放射阻止能 (radiative stopping power) に規格化する 規格化しない

### 4.2.20 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

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 $\geq$ 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 : wight 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 : wight 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

## 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 : px (GeV/c)
- = 2 : py (GeV/c)
- = 3 : pz (GeV/c)
- = 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)

### 4.2.21 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 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、 $(n, \gamma)$  反応なら必ず光子のみを放出、 $(n, \alpha)$  反応なら  $\alpha$  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となります。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となります。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。更に、熱中性子の温度依存の散乱の記述はできません。常に $T=0$ の取り扱いになります。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-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-heat]では、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発熱は、核データのKerma Factorを用いますが、e-modeでは、中性子からの直接の発熱の寄与はありません。中性子反応から生成された荷電粒子、残留核のIonization Lossから計算されます。光子は、electron=0を指定すると光子のKerma Factorを用いますが、電子の輸送も含みelectron=1を指定すれば、電子のIonization Lossから熱を計算します。[t-dpa]では、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でも、ライブラリーを用いずDPAを計算することが可能になります。

### 4.3 [Source]セクション

このセクションでは、入射粒子の情報を定義します。ソースタイプは、**s-type = 番号** で指定します。

表 27: ソースタイプ

ソースタイプ	説明
<b>s-type = 1</b>	円柱 (円面、ペンシルビーム)
<b>s-type = 4</b>	エネルギー分布を有する円柱
<b>s-type = 2</b>	角柱 (長方形)
<b>s-type = 5</b>	エネルギー分布を有する角柱
<b>s-type = 3</b>	ガウス分布 ( $x, y, z$ 独立)
<b>s-type = 6</b>	エネルギー分布を有するガウス分布 ( $x, y, z$ 独立)
<b>s-type = 7</b>	一般パラボラ分布 ( $x, y, z$ 独立)
<b>s-type = 8</b>	エネルギー分布を有する一般パラボラ分布 ( $x, y, z$ 独立)
<b>s-type = 9</b>	球及び球殻分布
<b>s-type = 10</b>	エネルギー分布を有する球及び球殻分布
<b>s-type = 11</b>	ビーム方向に垂直な位相空間で一様分布
<b>s-type = 12</b>	<code>decay-turtle</code> の出力を読み込む
<b>s-type = 13</b>	ガウス分布 ( $xy$ 平面)
<b>s-type = 14</b>	エネルギー分布を有するガウス分布 ( $xy$ 平面)
<b>s-type = 15</b>	一般パラボラ分布 ( $xy$ 平面)
<b>s-type = 16</b>	エネルギー分布を有する一般パラボラ分布 ( $xy$ 平面)
<b>s-type = 17</b>	<code>dump</code> ファイルの読み込み
<b>s-type = 18</b>	円錐形状
<b>s-type = 19</b>	エネルギー分布を有する円錐形状
<b>s-type = 100</b>	ユーザー定義のソースプログラム <code>usrsors.f</code> にプログラムを書きこみコンパイルする。

## 4.3.1 &lt;Source&gt; : マルチソース

s-type で指定するソースを複数定義することができます。各ソースは、<source> = 相対比、の指定で始まるサブセクションで定義します。相対比は、マルチソースのソース粒子の相対比を指定します。また、totfact = で全体の規格化定数を定義します。ただし、<source>の上の行で reg 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、各<source>に属するパラメータとして reg を最後に書かないように注意してください。

表 28: 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) マルチソース関連オプション。 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。

## 4.3.2 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 29: 共通ソースパラメータ

パラメータ	説明
<b>proj =</b>	入射粒子：粒子の指定方法は表を参照
<b>t-type = 0, 1, 2</b>	(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
<b>t0 =</b>	(D=0.0) 第 1 の時間分布の中心時刻 (ns)
<b>tw =</b>	時間分布の半値幅 (ns)
<b>tn =</b>	時間分布の数
<b>td =</b>	時間分布の間隔 (ns)
<b>tc =</b>	(D=10×tw) Gauss 分布の場合の cut off 時間 (ns)
<b>sx =</b>	(D=0) スピンの方向ベクトルの x 成分
<b>sy =</b>	(D=0) スピンの方向ベクトルの y 成分
<b>sz =</b>	(D=0) スピンの方向ベクトルの z 成分
<b>reg =</b>	(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 <b>reg = { 1 - 5 } 10 34</b> です。また <b>reg = ( 6 &lt; 10[1 0 0] &lt; u=3 )</b> などの lattice, universe 構造も指定できます。詳細は、タリーの領域指定箇所を参照して下さい。
<b>ntmax =</b>	(D=1000) 領域限定の際の最大再試行回数
<b>trcl =</b>	(D=なし) 座標変換番号もしくは座標変換定義
<b>wgt =</b>	(D=1.0) ソース粒子のウェイト。
<b>factor =</b>	(D=1.0) ソース粒子の規格化定数。全てのタリー結果を <b>factor</b> 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 <b>totfact</b> を用いて行って下さい。また、マルチソースでない場合、 <b>totfact * factor</b> が実質的な規格化定数になります。
<b>izst =</b>	(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る。ここで指定した値は輸送の途中で変化しない。ただし、核反応で生成された原子核の電荷は Z に戻る。

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2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  $\theta, \phi$  とした場合に、 $\cos\theta = \text{dir}$ ,  $\phi = \text{phi}$  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  $\psi = \text{dom}$ 、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シミュレートできます。

**dir** は **s-type=9,10** では定義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。この規格化定数 **factor** は、マルチソースの場合は、各ソースの規格化です。全体のソースの規格化は、**totfact** を用います。

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、初期スピンの定義されます。ここで、スピンの指定された場合は、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。

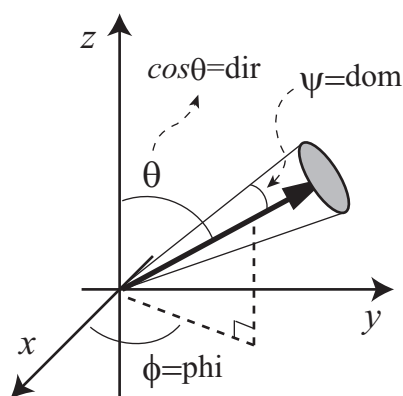


図2: 入射粒子の方向とパラメータ **dir**, **phi**, **dom** の関係

### 4.3.3 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 30: 円柱分布ソースパラメータ

s-type = 1, 4	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 =	円柱の z 軸方向の下限 (cm)
z1 =	円柱の z 軸方向の上限 (円面の場合は下限と同じ) (cm)
r0 =	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、中空部分を指定 (cm)
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=省略、ランダム) 入射粒子の方位角 (degree)。
dom =	(D=0.0) 入射粒子方向の立体角範囲 (degree)。 = -1 ; $\cos^2$ bias 分布
e0 =	入射粒子のエネルギー (s-type=1) (MeV/u)

### 4.3.4 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 31: 角柱分布ソースパラメータ

s-type = 2, 5	角柱のとき
x0 =	角柱の下限 x 座標 (cm)
x1 =	角柱の上限 x 座標 (cm)
y0 =	角柱の下限 y 座標 (cm)
y1 =	角柱の上限 y 座標 (cm)
z0 =	角柱の下限 z 座標 (cm)
z1 =	角柱の上限 z 座標 (長方形の場合は下限と同じ) (cm)
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=省略、ランダム) 入射粒子の方位角 (degree)。
dom =	(D=0.0) 入射粒子方向の立体角範囲 (degree)。 = -1 ; $\cos^2$ bias 分布
e0 =	入射粒子のエネルギー (s-type=2) (MeV/u)

4.3.5 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 32: ガウス分布 ( $x, y, z$  独立) ソースパラメータ

s-type = 3, 6	ガウス分布 ( $x, y, z$ 独立) のとき
x0 =	(D=0.0) ガウス分布中心の $x$ 座標 (cm)
x1 =	ガウス分布 $x$ 方向半値幅 (cm)
y0 =	(D=0.0) ガウス分布中心の $y$ 座標 (cm)
y1 =	ガウス分布 $y$ 方向半値幅 (cm)
z0 =	(D=0.0) ガウス分布中心の $z$ 座標 (cm)
z1 =	ガウス分布 $z$ 方向半値幅 (cm)
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=省略、ランダム) 入射粒子の方位角 (degree)
dom =	(D=0.0) 入射粒子方向の立体角範囲 (degree) = -1 ; $\cos^2$ bias 分布
e0 =	入射粒子のエネルギー (s-type=3) (MeV/u)

4.3.6 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 33: 一般パラボラ分布 ( $x, y, z$  独立) ソースパラメータ

s-type = 7, 8	一般パラボラ分布 ( $x, y, z$ 独立) のとき
x0 =	(D=0.0) X パラボラ分布中心の $x$ 座標 (cm)
x1 =	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 =	Y パラボラ分布の幅 (cm)
z0 =	(D=0.0) Z パラボラ分布中心の $z$ 座標 (cm)
z1 =	Z パラボラ分布の幅 (cm)
rn =	(D=2) 一般パラボラの次数
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=省略、ランダム) 入射粒子の方位角 (degree)
dom =	(D=0.0) 入射粒子方向の立体角範囲 (degree) = -1 ; $\cos^2$ bias 分布
e0 =	入射粒子のエネルギー (s-type=7) (MeV/u)

### 4.3.7 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 34: ガウス分布 ( $xy$  平面) ソースパラメータ

s-type = 13, 14	ガウス分布 ( $xy$ 平面) のとき
<b>x0</b> =	(D=0.0) ガウス分布中心の $x$ 座標 (cm)
<b>y0</b> =	(D=0.0) ガウス分布中心の $y$ 座標 (cm)
<b>r1</b> =	ガウス分布半値幅 (cm)
<b>z0</b> =	$z$ 軸方向の下限 (cm)
<b>z1</b> =	(D=z0) $z$ 軸方向の上限 (cm)
<b>dir</b> =	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
<b>phi</b> =	(D=省略、ランダム) 入射粒子の方位角 (degree)
<b>dom</b> =	(D=0.0) 入射粒子方向の立体角範囲 (degree) = -1 ; $\cos^2$ bias 分布
<b>e0</b> =	入射粒子のエネルギー (s-type=13) (MeV/u)

### 4.3.8 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 35: 一般パラボラ分布 ( $xy$  平面) ソースパラメータ

s-type = 15, 16	一般パラボラ分布 ( $xy$ 平面) のとき
<b>x0</b> =	(D=0.0) パラボラ分布中心の $x$ 座標 (cm)
<b>y0</b> =	(D=0.0) パラボラ分布中心の $y$ 座標 (cm)
<b>r1</b> =	パラボラ分布の幅 (cm)
<b>z0</b> =	$z$ 軸方向の下限 (cm)
<b>z1</b> =	(D=z0) $z$ 軸方向の上限 (cm)
<b>rn</b> =	(D=2) 一般パラボラの次数
<b>dir</b> =	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
<b>phi</b> =	(D=省略、ランダム) 入射粒子の方位角 (degree)
<b>dom</b> =	(D=0.0) 入射粒子方向の立体角範囲 (degree) = -1 ; $\cos^2$ bias 分布
<b>e0</b> =	入射粒子のエネルギー (s-type=15) (MeV/u)



## 4.3.9 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 36: 球及び球殻分布ソースパラメータ

s-type = 9, 10	球及び球殻分布のとき
x0 =	(D=0.0) 球及び球殻分布中心の x 座標 (cm)
y0 =	(D=0.0) 球及び球殻分布中心の y 座標 (cm)
z0 =	(D=0.0) 球及び球殻分布中心の z 座標 (cm)
r1 =	球及び球殻分布の内半径、ゼロの場合は球 (cm)
r2 =	球及び球殻分布の外半径 (cm)
dir =	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 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 <sup>2</sup> bias, 体積、面積計算に用いる data を指定した時は、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 の円から中心に向かって一様分布。r1=r2 の場合、dir = -all とほぼ同じ結果を与えるが、ウェイトを使っていないので等方線源を使ってエネルギー付与の分散を計算する ([t-deposit] で output = deposit) 場合に用いる。
e0 =	入射粒子のエネルギー (s-type=9) (MeV/u)

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  $1/\pi(r1)^2$  となります。また、dir = iso とした場合も同様となります。

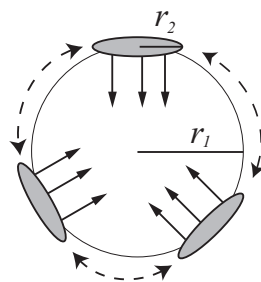


図 3: dir=iso の場合のイメージ図

## 4.3.10 s-type = 11 ソース

ビーム方向に垂直な位相空間で一様なソース分布を与えます。s-type = 11 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。

表 37: s-type = 11 のソースパラメータ

s-type = 11	ビーム方向に垂直な位相空間で一様分布のとき
x0 =	(D=0.0) ビーム中心の x 座標 (cm)
x1 =	x 方向の最大半径/最大角度の比 (cm/mrad)
y0 =	(D=0.0) ビーム中心の y 座標 (cm)
y1 =	y 方向の最大半径/最大角度の比 (cm/mrad)
z0 =	z 軸方向の下限 (cm)
z1 =	z 軸方向の上限 (cm)
rx =	x 方向における位相空間の楕円の傾き (rad)
ry =	y 方向における位相空間の楕円の傾き (rad)
wem =	エミッタンス ( $\pi$ cm $\times$ mrad)
dir =	入射粒子の軸方向の方向余弦、(1 or -1 only)
e0 =	入射粒子のエネルギー (MeV/u)

#### 4.3.11 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 38: 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 座標オフセット (cm)
y0 =	(D=0.0) ビームの y 座標オフセット (cm)
z0 =	(D=0.0) ビームの z 座標オフセット (cm)
dir =	入射粒子の軸方向の方向余弦、(1 or -1 only)
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 39: 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	ビーム粒子のウェイト
pz0	ビーム粒子の偏極度 (未使用)

## 4.3.12 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。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: 円錐形状分布ソースパラメータ

s-type = 18, 19	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 =	円錐頂点から底面へのベクトルの x 成分 (cm)
y1 =	円錐頂点から底面へのベクトルの y 成分 (cm)
z1 =	円錐頂点から底面へのベクトルの z 成分 (cm)
r0 =	(D=0.0) 頂点から円錐上面までの円錐側面での距離 (cm)
r1 =	頂点から円錐下面までの円錐側面での距離 (cm)
r2 =	円錐の母線から円錐側面の角度 (degree)
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=省略、ランダム) 入射粒子の方位角 (degree)
dom =	(D=0.0) 入射粒子方向の立体角範囲 (degree) = -1 ; cos <sup>2</sup> bias 分布
e0 =	入射粒子のエネルギー (s-type=18) (MeV/u)

## 4.3.13 dump データソース

dump ファイルに落とされたデータをソースとして取り込みます。s-type = 17 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 41: dump データのソースパラメータ

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
(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=省略、ランダム) 入射粒子の方位角 (degree)
dom =	(D=0.0) 入射粒子方向の立体角範囲 (degree) = -1 ; cos <sup>2</sup> bias 分布
e0 =	入射粒子のエネルギー (MeV/u)
e-type =	入射粒子のエネルギー分布指定
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
(以下省略可能)	以下のものも指定可能です
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

`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 42: `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 43: `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/u)、原子核の場合は核子当たりのエネルギー、`wt` は粒子ウェイト、`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。

### 4.3.14 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 44: 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 =	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=省略、ランダム) 入射粒子の方位角 (degree)
dom =	(D=0.0) 入射粒子方向の立体角範囲 (degree) = -1 ; cos <sup>2</sup> bias 分布
e0 =	入射粒子のエネルギー (MeV/u)
e-type =	入射粒子のエネルギー分布指定
wgt =	(D=1.0) ソース粒子のウェイト
factor =	(D=1.0) ソース粒子のウェイトの規格化定数
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

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。

List 4.1 ● `usrsors.f`

```

1: *****
2:      subroutine usrsors(x,y,z,u,v,w,e,wt,time,name,kf,nc1,nc2,nc3,
3:      &                sx,sy,sz)
4: *      sample subroutine for user defined source. *
5: *      variables : *
6: *      x, y, z : position of the source. *
7: *      u, v, w : unit vector of the particle direction. *
8: *      e      : kinetic energy of particle (MeV). *
9: *      wt     : weight of particle. *
10: *      time   : initial time of particle. (ns) *
11: *      name   : usually = 1, for Coulmb spread. *
12: *      kf     : kf code of the particle. *
13: *      nc1    : initial value of counter 1 *
14: *      nc2    : initial value of counter 2 *
15: *      nc3    : initial value of counter 3 *
16: *      sx,sy,sz : spin components *
17: *-----*
18: *      kf code table *
19: *      kf-code: ityp : description *
20: *      2212 : 1 : proton *
21: *      2112 : 2 : neutron *
22: *      211 : 3 : pion (+) *
23: *      111 : 4 : pion (0) *
24: *      -211 : 5 : pion (-) *
25: *      -13 : 6 : muon (+) *
26: *      13 : 7 : muon (-) *
27: *      321 : 8 : kaon (+) *
28: *      311 : 9 : kaon (0) *
29: *      -321 : 10 : kaon (-) *
30: *      kf-code of the other transport particles *
31: *      12 : nu_e *
32: *      14 : nu_mu *
33: *      221 : eta *
34: *      331 : eta' *
35: *      -311 : k0bar *
36: *      -2112 : nbar *
37: *      -2212 : pbar *
38: *      3122 : Lambda0 *
39: *      3222 : Sigma+ *
40: *      3212 : Sigma0 *
41: *      3112 : Sigma- *
42: *      3322 : Xi0 *
43: *      3312 : Xi- *
44: *      3334 : Omega- *
45: *-----*
46: *      available function for random number *
47: *      unirn(dummy) : uniform random number from 0 to 1 *
48: *      gaurn(dummy) : gaussian random number *
49: *      for exp( - x**2 / 2 / sig**2 ) : sig = 1.0 *
50: *****
51:      implicit real*8 (a-h,o-z)

```

```
52:  *-----
53:      parameter ( pi = 3.141592653589793d0 )
54:      data ifirst / 0 /
55:      save ifirst
56:      character filenm*50
57:  *-----
58:  *      example of initialization
59:  *-----
60:      if( ifirst .eq. 0 ) then
61:  c          filenm = 'input.dat'
62:  c          inquire( file = filenm, exist = exex )
63:  c          if( exex .eqv. .false. ) then
64:  c              write(*,*) 'file does not exist => ', filenm
65:  c              call parastop( 887 )
66:  c              end if
67:  c          open(71, file = file(i), status = 'old' )
68:
69:  c          close(71)
70:          ifirst = 1
71:      end if
72:  *-----
73:  *      example for 3 GeV proton with z-direction
74:  *-----
75:          x = 0.0
76:          y = 0.0
77:          z = 0.0
78:          u = 0.0
79:          v = 0.0
80:          w = 1.0
81:          e = 3000.0
82:          wt = 1.0
83:          time = 0.0
84:          name = 1
85:          kf = 2212
86:          nc1 = 0
87:          nc2 = 0
88:          nc3 = 0
89:          sx = 0.d0
90:          sy = 0.d0
91:          sz = 0.d0
92:  *-----
93:      return
94:      end
```



## 4.3.15 エネルギー分布の定義

ソースがエネルギー分布をもつ **s-type = 4, 5, 6, 8, 10, 14, 16**、もしくは **s-type = 17, 100** の場合は、**e-type** サブセクションを用いてソース粒子のエネルギー分布を指定する必要があります。原子核入射のときは、核子当たりのエネルギーです。

**e-type = 1, 4, 11, 14** の場合は、各ビンのエネルギー幅で積分した値を与えることで、エネルギー分布を表現します。一方、**e-type = 21, 24, 31, 34** の場合は、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。**e-type = 2, 3, 5, 6, 7, 12, 14, 15, 16** の場合は、エネルギーの微分スペクトルを様々な関数を用いて与えることができます。また、**e-type = 8, 9, 18, 19** の場合は、エネルギー分布がポイントワイズ、すなわち各点各点で与えられる不連続なものとなります。

(D=\*\*\*) のあるものは、省略可能です。

表 45: ソースエネルギー分布パラメータ (1)

パラメータ	説明
<b>e-type = 1, (11)</b>	エネルギー分点 <b>e(i)</b> と各ビンにおけるソース粒子の生成確率の積分値 <b>w(i)</b> を与えることにより、任意のエネルギー分布を指定する。統計的に <b>w(i)</b> 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。
<b>ne =</b>	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。 ( <b>e(i), w(i), i=1,  ne </b> ), <b>e( ne +1)</b> 各ビンに生成される粒子数の積分値は <b>w(i)</b> に比例する。
<b>e-type = 4, (14)</b>	<b>e-type = 1, (11)</b> と同じエネルギー分布を発生させる。ただし、 <b>e-type = 1, (11)</b> 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、 <b>e-type = 4, (14)</b> 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 <b>w(i)</b> 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、 <b>p-type = 1</b> とし生成個数比 <b>p(i)</b> 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。14 の時は、エネルギーを波長 (Å) で与える。
<b>ne =</b>	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。 ( <b>e(i), w(i), i=1,  ne </b> ), <b>e( ne +1)</b> デフォルト ( <b>p-type = 0</b> ) では各ビンに等しい個数が生成される。 <b>p-type = 1</b> で <b>p(i)</b> 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。
<b>p-type = 0, 1</b>	(D=0) 生成個数のオプション for 0, 全てのビンで <b>p(i)=1</b> 、以下のデータは無し for 1, 各ビンの生成個数比 <b>p(i)</b> を次の行からデータで与える ( <b>p(i), i=1, ne</b> )

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, e-type=2の代わりに e-type=11, e-type=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、入力に変換式を使って 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。

表 46: ソースエネルギー分布パラメータ (2)

パラメータ	説明
<b>e-type = 21, (31)</b>	エネルギー分点 $e(i)$ と各ビンにおけるソース粒子の生成確率の微分量 $[d\varphi/dE](i)$ を与えることにより、任意のエネルギー分布を指定する。統計的に $[d\varphi/dE](i)*\{e(i+1)-e(i)\}$ 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。
<b>ne =</b>	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。 $(e(i), [d\varphi/dE](i), i=1,  ne ), e( ne +1)$ 各ビンに生成される粒子数の積分値は $d\varphi/dE(i)*\{e(i+1)-e(i)\}$ に比例する。
<b>e-type = 24, (34)</b>	<b>e-type = 21, (31)</b> と同じエネルギー分布を発生させる。ただし、 <b>e-type = 21, (31)</b> 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、 <b>e-type = 24, (34)</b> 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、 <b>p-type = 1</b> 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。 34 の時は、エネルギーを波長 (Å) で与える。
<b>ne =</b>	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 ( <b>p-type = 0</b> ) では各ビンに等しい個数が生成される。 <b>p-type = 1</b> 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。
<b>p-type = 0, 1</b>	(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 $(p(i), i=1, ne)$

表 47: ソースエネルギー分布パラメータ (3)

パラメータ	説明
<b>e-type = 8, (18)</b>	エネルギー分点 $e(i)$ と各点のソース粒子の生成確率 $w(i)$ を与えることにより、任意のエネルギー分布を指定する。統計的に $w(i)$ に比例するように各点に粒子を生成させ、不連続なエネルギー分布を表現する。 18 の時は、エネルギーを波長 ( $\text{\AA}$ ) で与える。
<b>ne =</b>	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。
<b>e-type = 9, (19)</b>	<b>e-type = 8, (18)</b> と同じエネルギー分布を発生させる。ただし、 <b>e-type = 8, (18)</b> 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、 <b>e-type = 9, (19)</b> では、全てのエネルギー一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、 <b>p-type = 1</b> 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ソース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 ( $\text{\AA}$ ) で与える。
<b>ne =</b>	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 ( <b>p-type=0</b> ) 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。
<b>p-type = 0, 1</b>	(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 $(p(i), i=1, ne)$

表 48: ソースエネルギー分布パラメータ (4)

パラメータ	説明
<b>e-type = 2, (12)</b>	微分線源スペクトル ( $d\phi/dE$ ) をガウス分布で与える。 12 の時は、エネルギーを波長 ( $\text{\AA}$ ) で与える
<b>eg0 =</b> <b>eg1 =</b> <b>eg2 =</b> <b>eg3 =</b>	ガウス分布の中心値 (MeV) ガウス分布の半値幅 (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)
<b>e-type = 3</b>	微分線源スペクトル ( $d\phi/dE$ ) を Maxwellian 分布 $f(x) = x^{1.5} \exp(-x/T)$ で与える。
<b>nm =</b>	(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。
<b>et0 =</b> <b>et1 =</b> <b>et2 =</b>	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)
<b>e-type = 7</b>	<b>e-type = 3</b> と同じエネルギー分布を発生させる。ただし、 <b>e-type = 3</b> 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、 <b>e-type = 7</b> 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 $f(x) = x^{1.5} \exp(-x/T)$ 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、 <b>p-type = 1</b> 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。
<b>nm =</b>	(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 ( <b>p-type = 0</b> ) では各ビンに等しい個数が生成される。 <b>p-type = 1</b> 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。
<b>et0 =</b> <b>et1 =</b> <b>et2 =</b>	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)
<b>p-type = 0, 1</b>	(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、( $p(i), i=1, nm$ )

表 49: ソースエネルギー分布パラメータ (5)

パラメータ	説明
<b>e-type = 5, (15)</b>  <b>f(x) =</b>	微分線源スペクトル ( $d\varphi/dE$ ) を任意の関数 $f(x)$ で与える。 15 の時は、エネルギーを波長 ( $\text{\AA}$ ) で与える。 FORTRAN 形式で書いた関数。x はエネルギー (MeV/u) を表す。 内部変数や定数が使えます。例) $f(x) = \exp(-c1*x**2)$
<b>nm =</b>	(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。
<b>eg1 =</b> <b>eg2 =</b>	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)
<b>e-type = 6, (16)</b>  <b>f(x) =</b>	<b>e-type = 5, (15)</b> と同じエネルギー分布を発生させる。 ただし、 <b>e-type = 5, (15)</b> 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、 <b>e-type = 6, (16)</b> 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、 <b>p-type = 1</b> 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。 16 の時は、エネルギーを波長 ( $\text{\AA}$ ) で与える。 FORTRAN 形式で書いた関数。x はエネルギー (MeV/u) を表す。 内部変数や定数が使えます。例) $f(x) = \exp(-c1*x**2)$
<b>nm =</b>	(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 ( <b>p-type = 0</b> ) では各ビンに等しい個数が生成される。 <b>p-type = 1</b> 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。
<b>eg1 =</b> <b>eg2 =</b>	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)
<b>p-type = 0, 1</b>	(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、 ( $p(i), i=1, nm$ )

## 4.3.16 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 50: 角分布パラメータ (1)

パラメータ	説明
<code>a-type = 1, (11)</code>	角度分点 $a(i)$ と各ビンにおけるソース粒子の生成確率 $w(i)$ 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 <code>cos</code> で、11 の時は <code>degree</code> 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。
<code>na =</code>	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 $ ), $a( na +1)$ )
<code>a-type = 4, (14)</code>	<code>a-type = 1, (11)</code> と同じ角度分布を発生させる。ただし、 <code>a-type = 1, (11)</code> が生成数を調整して角度分布を表現するのに対し、 <code>a-type = 4, (14)</code> 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 <code>cos</code> で、14 の時は <code>degree</code> 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。
<code>na =</code>	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 ( <code>q-type = 0</code> ) では各ビンに等しい個数が生成される。 <code>q-type = 1</code> 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。
<code>q-type = 0, 1</code>	(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 51: 角度分布パラメータ (2)

パラメータ	説明
<b>a-type = 5, (15)</b>  <b>g(x) =</b>	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 $\text{degree}$ で与える。 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) $g(x) = \exp(-c1*x**2)$
<b>nn =</b>	角度群数。
<b>ag1 =</b>	角度分布のカットオフ最小値
<b>ag2 =</b>	角度分布のカットオフ最大値
<b>a-type = 6, (16)</b>  <b>g(x) =</b>	<b>a-type = 5, (15)</b> と同じ角度分布を発生させる。 ただし、 <b>a-type = 5, (15)</b> が生成数を調整して角度分布を表現するのに対し、 <b>a-type = 6, (16)</b> では、全ての角度ビンに同数の粒子を生成させ、粒子のウェイトを任意の関数 $g(x)$ 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 $\text{degree}$ で与える。 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) $g(x) = \exp(-c1*x**2)$
<b>nn =</b>	角度群数。 デフォルト ( <b>q-type = 0</b> ) では各ビンに等しい個数が生成される。 <b>q-type = 1</b> 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。
<b>ag1 =</b>	角度分布のカットオフ最小値
<b>ag2 =</b>	角度分布のカットオフ最大値
<b>q-type = 0, 1</b>	(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、 ( $q(i), i=1, nm$ )



## 4.3.17 マルチソースの例題

マルチソースの例題を紹介する。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る。まず、ソースセクションのリストを示す。

## List 4.2 ● マルチソース例題

```

1:  [ S o u r c e ]
2:  totfact = 3
3:  <source> = 9.72
4:  s-type = 4
5:  proj = proton
6:  z0 = 2
7:  z1 = 29
8:  r0 = 5
9:  r1 = 4
10: dir = 0.0
11: e-type = 6
12: eg1 = 1.e-6
13: eg2 = 1.e-3
14: nm = -200
15: set: c10[1.e-4]
16: f(x) = x**(1.5)*exp(-x/c10)
17: <source> = 1
18: s-type = 4
19: proj = photon
20: z0 = 1
21: z1 = 2
22: r0 = 5
23: dir = -1
24: e-type = 5
25: eg1 = 1.e-3
26: eg2 = 5.e-1
27: nm = 200
28: set: c10[1.e-1]
29: set: c20[1.e-1/2.35482]
30: f(x) = exp(-(x-c10)**2/2/c20**2)
31: <source> = 1
32: s-type = 4
33: proj = neutron
34: z0 = 29
35: z1 = 30
36: r0 = 5
37: e-type = 6
38: eg1 = 1.e-2
39: eg2 = 1.e+3
40: nm = -200
41: set: c10[92.469]
42: set: c20[5.644e+10]
43: f(x) = c10/c20*exp(-sqrt(x*(x+1876))/c10)*(x+938)/sqrt(x*(x+1876))
44: dir = data
45: a-type = 5
46: ag1 = 0
47: ag2 = 1
48: nn = 200
49: g(x) = exp(-(x-1)**2/0.3**2)

```

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。

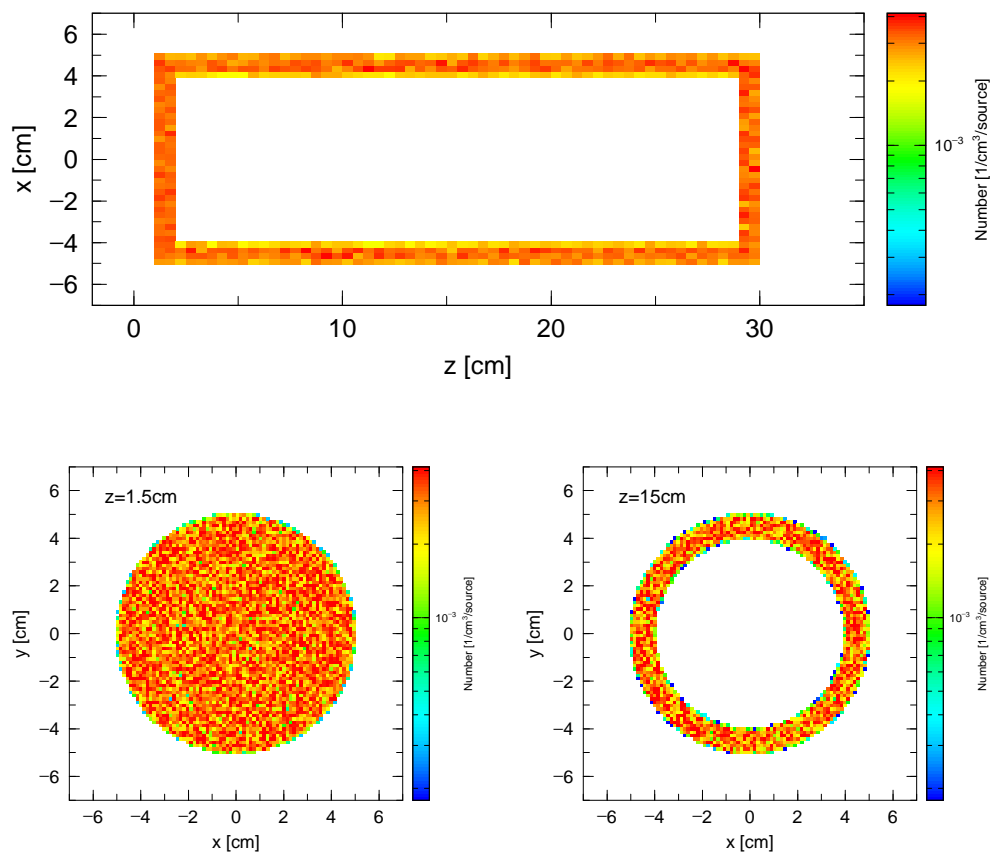


図 4: マルチソース、空間分布

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、各ソースごとのエネルギー分布が色別に表されている。

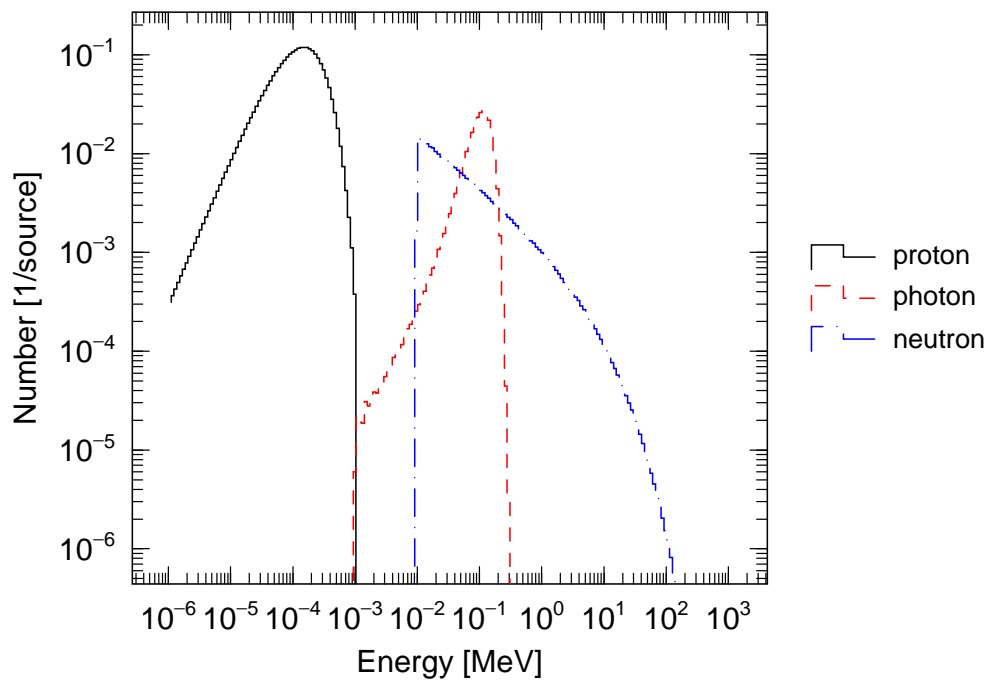


図 5: マルチソース、エネルギー分布

最初のソースは、 $\text{dir}=0$ , 即ち 90 度方向、2 番目が  $\text{dir}=-1$ , 180 度方向、3 番目が  $\text{dir}=\text{data}$  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。

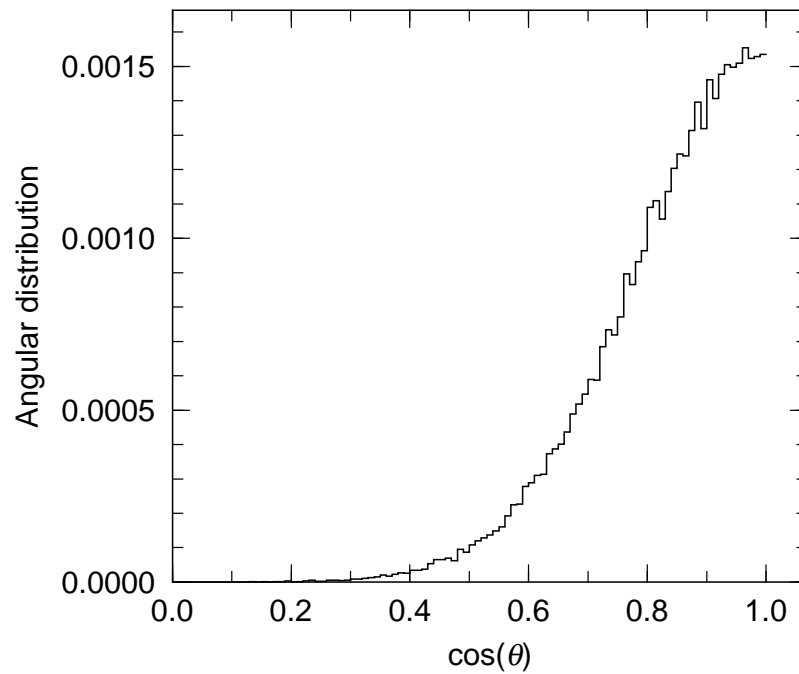


図 6: マルチソース、角度分布

## 4.3.18 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”と呼ばれますが、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、粒子のウェイトを変化させて、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, 4, 2, 5** 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 52: ダクトソースオプションのパラメーター

パラメーター	説明
<b>dom = -10</b>	ダクトソースの指定
<b>d10 =</b>	ビームラインの始点、 <b>z0</b> からの距離 (cm)
<b>d11 =</b>	ダクトソースの始点、 <b>z0</b> からの距離 (cm)
<b>d12 =</b>	ダクトソースの終点、 <b>z0</b> からの距離 (cm)
<b>dpf =</b>	<b>d12</b> を通過する粒子の割合
<b>drd =</b>	<b>s-type = 1, 4</b> 円柱の時のビームラインの半径 (cm)
<b>dxw =</b>	<b>s-type = 2, 5</b> 角柱の時のビームラインの X サイズ (cm)
<b>dyw =</b>	<b>s-type = 2, 5</b> 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, 4** の時は、円柱、**s-type = 2, 5** 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, 4** の時は、**r0** の内部、もしくは、**s-type = 2, 5** の時は、**x0, x1, y0, y1** の内部で生成されます。

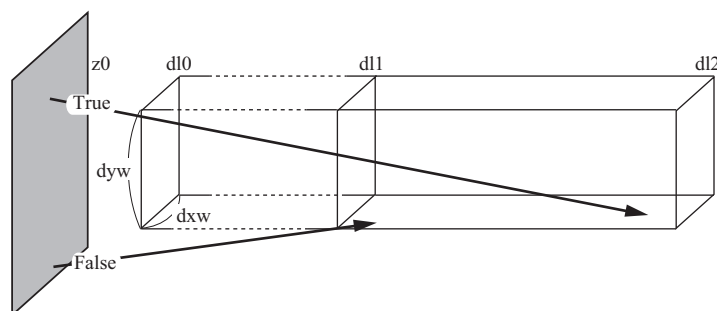


図 7: 角柱のダクトソースのイメージ図

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。**d11** から **d12** 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まります。そのとき、等方分布を仮定して粒子のウェイトが決まります。全体の規格化定数は、**z0** 地点で **d10** 地点のダクト面積と同じ領域で発生した粒子が **d10** 地点のダクトを通過する粒子数で規格化されます。通常、**d10** から **d12** 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、**d10** 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、

List 4.3 ● ダクトソースオプション例題 1

```

1  [ S o u r c e ]
2:
3:  set: c1[200]    $dl0
4:  set: c2[500]    $dl1
5:  set: c3[5000]   $dl2
6:  set: c4[5.0]    $x*2 at z0
7:  set: c5[5.0]    $y*2 at z0
8:  set: c10[5.0]   $dxw
9:  set: c20[5.0]   $dyw
10: set: c30[0.001] $dpf
11:
12: s-type = 2
13: proj  = neutron
14: e0    = 20.0
15: x0    = -c4/2
16: x1    = c4/2
17: y0    = -c5/2
18: y1    = c5/2
19: z0    = 0.0
20: z1    = 0.0
21: dir   = 1.0
22: phi   = 0.0
23:
24: dom = -10
25: dl0 = c1
26: dl1 = c2
27: dl2 = c3
28: dxw = c10
29: dyw = c20
30: dpf = c30

```

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。

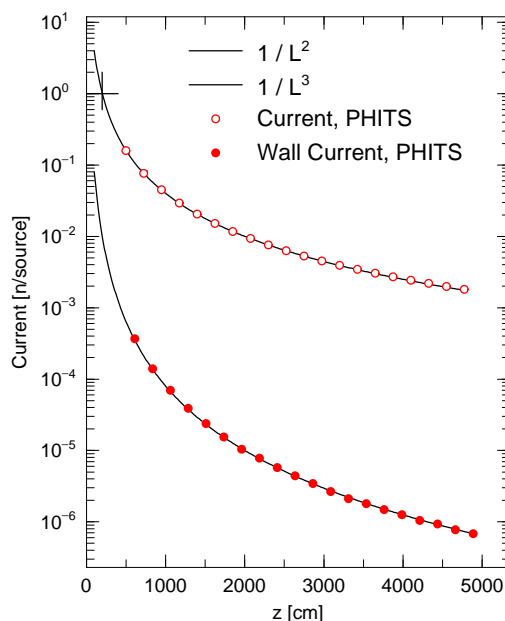


図 8: ダクトソースオプション例題 1

次の例題は、ソース領域の大きさを変えたものです。

List 4.4 ● ダクトソースオプション例題 2

```

1  [ S o u r c e ]
2:
3:  set: c1[200]    $d10
4:  set: c2[500]    $d11
5:  set: c3[5000]   $d12
6:  set: c4[10.0]   $x*2 at z0
7:  set: c5[10.0]   $y*2 at z0
8:  set: c10[5.0]   $dxw
9:  set: c20[5.0]   $dyw
10: set: c30[0.001] $dpf
11:
12: s-type = 2
13: proj  = neutron
14: e0    = 20.0
15: x0    = -c4/2
16: x1    = c4/2
17: y0    = -c5/2
18: y1    = c5/2
19: z0    = 0.0
20: z1    = 0.0
21: dir   = 1.0
22: phi   = 0.0
23:
24: dom = -10
25: d10 = c1
26: d11 = c2
27: d12 = c3
28: dxw = c10
29: dyw = c20
30: 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。

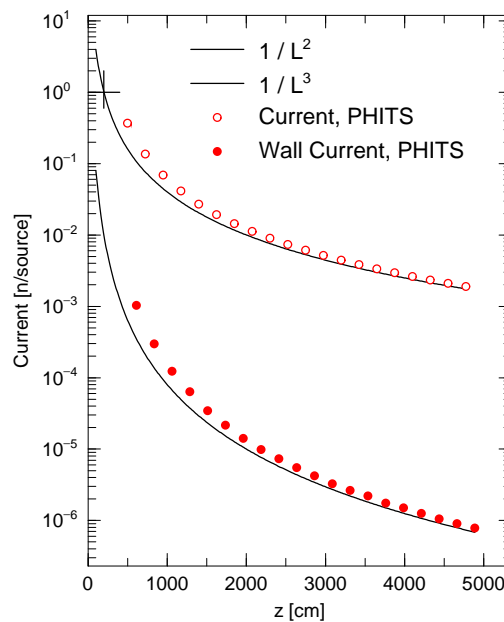


図 9: ダクトソースオプション例題 2

## 4.4 [Material] セクション

### 4.4.1 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。書式は次の2種類があります。

最初は、従来 of 書式で、

```
[ M a t e r i a l ]
  M A T [ m ]
    n u c l e u s   d e n s i t y
    n u c l e u s   d e n s i t y
    n u c l e u s   d e n s i t y
    n u c l e u s   d e n s i t y
  M A T [ m ]
    k e y w o r d = v a l u e   . . . .
    n u c l e u s   d e n s i t y
    n u c l e u s   d e n s i t y
    M T m   S ( a , b ) i d e n t i f i e r   . . .
  M A T [ m ]
    . . . . .   . . . . .
    . . . . .   . . . . .
```

この書式に加えて以下の入力ファイルの物質定義の書式も使えます。

```
[ M a t e r i a l ]
  M m   n u c l e u s   d e n s i t y           n u c l e u s   d e n s i t y
        n u c l e u s   d e n s i t y           n u c l e u s   d e n s i t y
  M m
    k e y w o r d = v a l u e   . . . .
    n u c l e u s   d e n s i t y           n u c l e u s   d e n s i t y
    n u c l e u s   d e n s i t y           n u c l e u s   d e n s i t y
    M T m   S ( a , b ) i d e n t i f i e r   . . .
  M m
    . . . . .   . . . . .
    . . . . .   . . . . .
```

ここで、*m* は、物質番号で重ならない限り 9999 まで自由です。



#### 4.4.2 核種の定義

*nucleus* は、核種の定義で、**208Pb**, **Pb-208**, **82208** どの形式でも可能です。水素は、**1H**, **H-1**, **1001** と指定します。質量数をゼロにすると、即ち、**Pb**, **82000** とすると、その原子の天然同位体比を仮定します。

核データを用いる時は、核種の記述の後のピリオドに続け、ライブラリーの番号 (2桁の数字)、データのクラス (1文字のアルファベット) で指定します。例えば、**208Pb.24c**, **Pb-208.24c**, **82208.24c** です。(ただし、天然同位体比を仮定する表記は、ライブラリーにその情報が含まれていない限り使用できません。) 指定しない場合は、断面積ディレクトリファイル **file(7)** (**D=xmdir**) 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。核種毎ではなく、物質番号毎にデフォルトのライブラリーを指定したい場合は、表 53 の **NLIB**, **PLIB**, **ELIB**, **HLIB** を利用してください。ただし、これらで指定していても、上記の書式で核種毎に示してあるライブラリーが優先されますのでご注意ください。どの核データが使用されたのかを知りたい場合は、**[parameters]** セクションの **kmout** を 1 にすると、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。

炭素を定義するとき、“C” で始めるとコメント行とされてしまいます。“6000”、“6012”または“12C”で指定するようにしてください。

#### 4.4.3 密度の定義

*density* は密度の定義で、単位は、*density* が正の時は、粒子密度 [ $10^{24}$  atoms/cm<sup>3</sup>]、負の時は、質量密度 [g/cm<sup>3</sup>] です。これらの密度は、**[cell]** セクションで密度が定義されていれば、それらによって再規格化されます。従って、**[cell]** セクションで密度が定義されている時は、このセクションでの定義は、密度である必要はなく粒子数の比でもかまいません。

#### 4.4.4 物質パラメータ

核データ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 53: 物質パラメータ

パラメータ	値	説明
GAS	D = 0 = 0 = 1	電子の阻止能の密度効果 液体もしくは固体の計算に適する 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	< 0 = 0 > 0	伝導体設定 絶縁体 (Default) 1 個でも絶縁体があれば絶縁体、それ以外は伝導体 1 個でも伝導体があれば伝導体

#### 4.4.5 $S(\alpha, \beta)$ の指定

低エネルギー中性子の輸送の際、 $S(\alpha, \beta)$  のライブラリーが必要になる場合があります。 $S(\alpha, \beta)$  ライブラリーは

MTm 物質 ID

で指定できます。m は、その物質の物質番号です。物質 ID は、`xmdir` に書かれている ID 番号 (lwtr.20t など) で、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。

#### 4.4.6 例題

幾つかの例題を下に示します。

List 4.5 ● material example (1)

```

1: [ M a t e r i a l ]
2: MAT[ 1 ]
3:   1H          1.0000000E-04
4:  208Pb       1.7238000E-02
5:  204Pb       4.6801000E-04
6:  206Pb       7.9430000E-03
7:  207Pb       7.2838000E-03
8: MAT[ 2 ]
9:   1H          1.0000000E-09
10:  14N         4.6801000E-05
11:  160         7.9430000E-06

```

核種と密度の順番は、核種-密度がデフォルトで、これを逆にしたいときは、

List 4.6 ● material example (2)

```

1: [ M a t e r i a l ]
2:   den          nuc          <-----
3: MAT[ 1 ]
4: 1.0000000E-04    1H
5: 1.7238000E-02    208Pb
6: 4.6801000E-04    204Pb
7: 7.9430000E-03    206Pb
8: 7.2838000E-03    207Pb
9: MAT[ 2 ]
10: 1.0000000E-09    1H
11: 4.6801000E-05    14N
12: 7.9430000E-06    160

```

のように、`den nuc` を入れて定義します。

List 4.7 ● material example (3)

```
1: [ M a t e r i a l ]
2: m1      80196.49c  5.9595d-5
3:        80198.49c  3.9611d-3
4:        80199.49c  6.7025d-3
5:        80200.49c  9.1776d-3
6:        80201.49c  5.2364d-3
7:        80202.49c  1.1863d-2
8:        80204.49c  2.2795d-3
9: c      ...Be...
10: m3     4009.37c  1.2362E-1
11: mt3    be.01
12: c      ...h2o (25C)...
13: m4     1001.37c  6.6658d-2      8016.37c  3.3329d-2
14: mt4    lwtr.01
15: c      ...b4c (natural boron; 25%-density)...
16: m5     6012.37c  6.8118d-3
17:        5011.37c  2.1825d-2
18: c      ...liquid-h2 (20K)...
19: m6     1001.49c  3.1371d-2      1011.49c  1.0457d-2
20: mt6    orthoh.00  parah.00
```

## 4.5 [Cell]セクション

### 4.5.1 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54を参照してください。

[Cell]				
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4.5.5節をご覧ください。セルパラメータの種類については表55にまとめています。

表 54: 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、[material]セクションで定義した density が正の場合は粒子密度 ( $10^{24}$ atoms/cm <sup>3</sup> )、負の場合は質量密度 (g/cm <sup>3</sup> ) となります。ここで指定した原子数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4.5.2節をご参照ください。
LIKE <i>n</i> BUT	セルの内容がセル番号 <i>n</i> 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 <i>n</i> 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55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。

表 55: セルパラメータ

項目	説明
VOL	そのセルの体積 (cm <sup>3</sup> ) 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 4.5.3 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 4.5.4 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 4.5.3, 4.5.4 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。

#### 4.5.2 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、

##### List 4.8 ● [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よう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 10 で示す仮想空間を構築します。図 10 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。

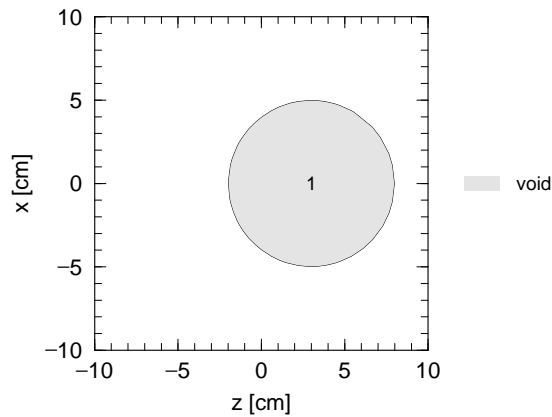


図 10: 例題(1)の空間を  $xz$  平面で切り取った断面図。セル番号1の領域はボイド(真空)。

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  $\sqcup$ (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、 $\sqcup$ (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。

List 4.9 ● **[cell]** セクションの例題(2)

```

1:  [ C e l l ]
2:   1  0  11 -12  13 -14  15 -16
3:   2 -1  #1
4:  [ S u r f a c e ]
5:   11 P X  -6
6:   12 P X   6
7:   13 P Y  -6
8:   14 P Y   6
9:   15 P Z  -6
10:  16 P Z   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  $\sqcup$ (空白) を用いて **11 $\sqcup$ -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  $\sqcup$  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11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。

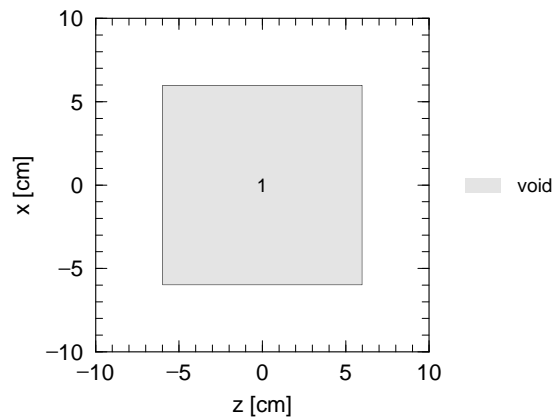


図 11: 例題 (2) の空間を  $xz$  平面で切り取った断面図。セル番号 1 の領域はボイド (真空)。

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。

List 4.10 ● [cell] セクションの例題 (3)

```

1: [ C e l l ]
2:   1  0  -10 : (11  -12  13  -14  15  -16)
3:   2  -1  #1
4: [ S u r f a c e ]
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 12 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次元平面では図のようになっています。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。

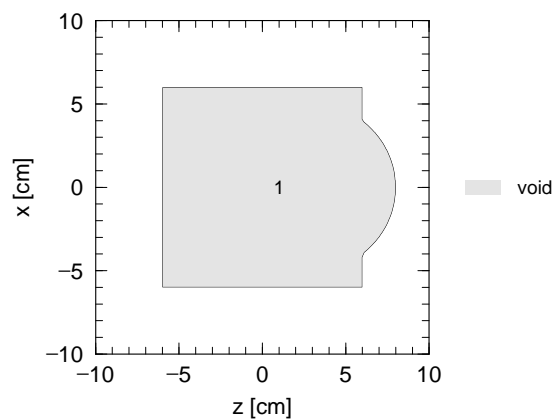


図 12: 例題 (3) の空間を  $xz$  平面で切り取った断面図。セル番号 1 の領域はボイド (真空)。

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。

List 4.11 ● [cell] セクションの例題 (4)

```

1: [ M a t e r i a l ]
2:   mat[1] 1H 2 160 1
3: [ C e l l ]
4:   1 0 -10
5:   2 1 1.0 10 (11 -12 13 -14 15 -16)
6:   3 -1 #1 #2
7: [ S u r f a c e ]
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  $\square$  (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  $1.0 \times 10^{24}$  atoms/cm<sup>3</sup> の“water”が満たされています。図 13 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。

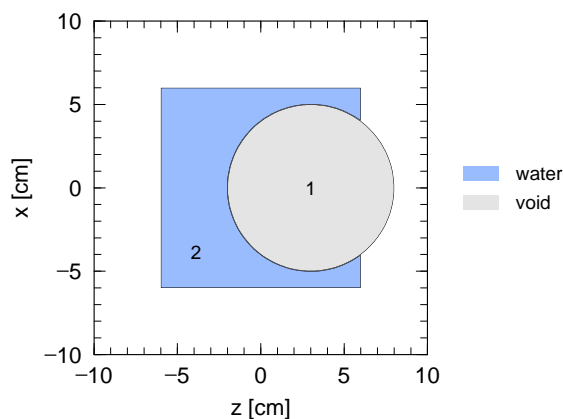


図 13: 例題 (4) の空間を xz 平面で切り取った断面図。セル番号 1 の領域はボイド (真空) とし、セル番号 2 の領域には水を満たした。

#### 4.5.3 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 4.5.5 節の繰り返し幾何形状を利用するときには有効となる機能です。

例題として、図 14 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、半径 12cm 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 5cm、高さ 12cm 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きます。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。

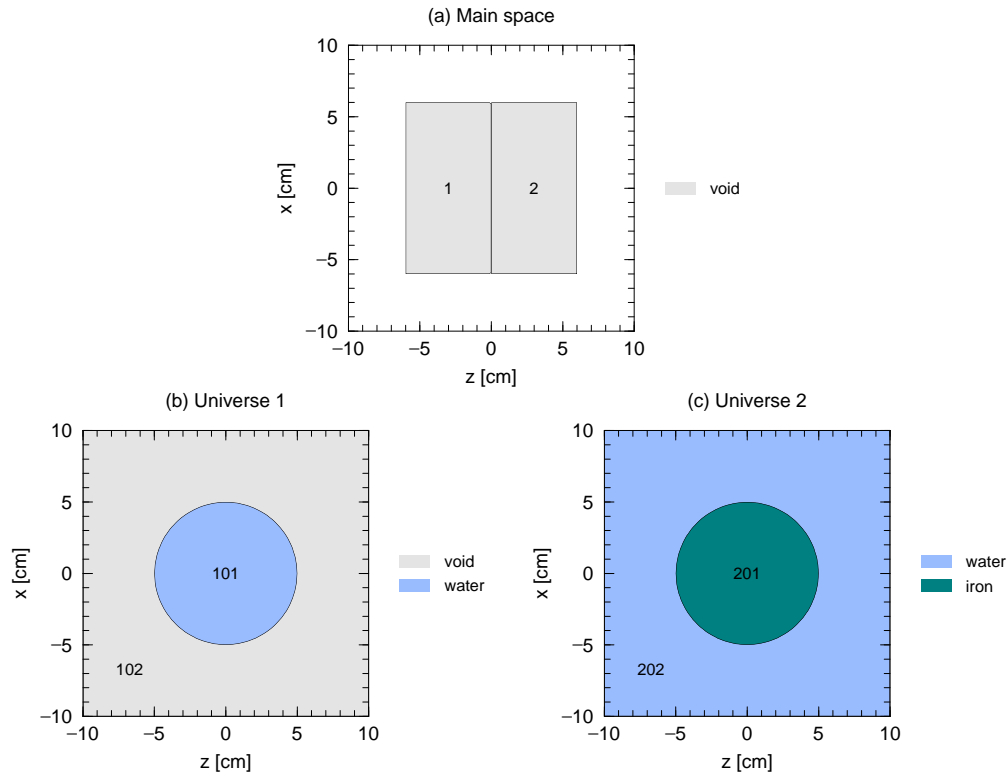


図 14: (a) 立方体を  $yz$  平面で分割した 2 つの直方体。(b) 円柱内部に水を満たし、それ以外はボイドとした universe 1。(c) 鉄の円柱の周りを水で満たした universe 2。

(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。

List 4.12 ● [cell] セクションの例題 (5)

```

1: [ M a t e r i a l ]
2:   mat[1] 1H 2 160 1
3:   mat[2] Fe 1
4: [ C e l l ]
5:   1 0      11 -12 13 -14 15 -17 FILL=1
6:   2 0      11 -12 13 -14 17 -16 FILL=2
7:  101 1 1.0 -10 13 -14 U=1
8:  102 0      #101 U=1
9:  201 2 10.0 -10 13 -14 U=2
10: 202 1 1.0 #201 U=2
11:   9 -1      #1 #2
12: [ S u r f a c e ]
13:  10 CY 5
14:  11 PX -6
15:  12 PX 6
16:  13 PY -6
17:  14 PY 6
18:  15 PZ -6
19:  16 PZ 6
20:  17 PZ 0

```

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 14(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 15 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。

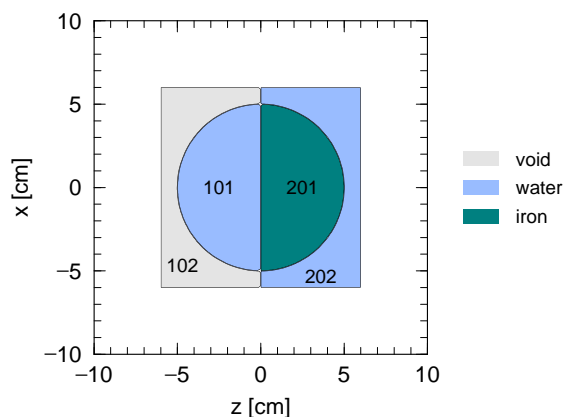


図 15: 例題 (5) の空間を  $xz$  平面で切り取った断面図。セル番号 1 の領域に **universe 1** のセル番号 101 と 102 が、セル番号 2 の領域に **universe 2** のセル番号 201 と 202 が入っている。

**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 16 の様に円柱の一部が立方体の中に入らなくなります。

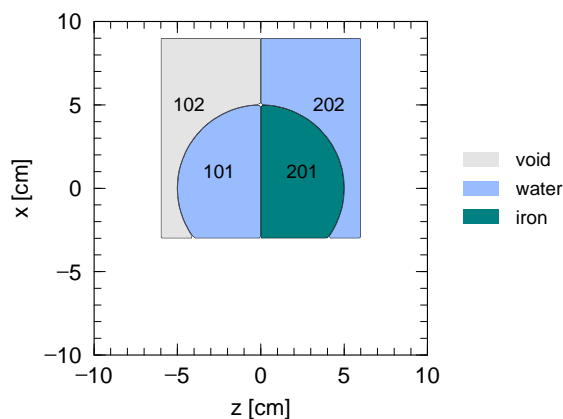


図 16: 例題 (5) において、セル番号 1 と 2 の範囲を  $x$  方向に関してずらした場合。

#### 4.5.4 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 4.5.5 節をご覧ください。

Lattice 構造は、図 17 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 17 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。

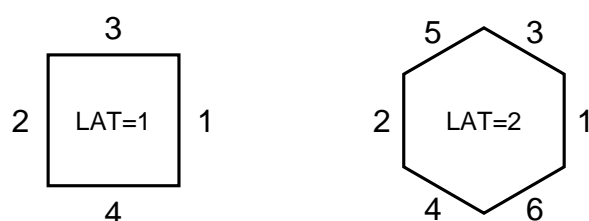


図 17: Lattice 構造の基本単位として使用できる四角形 (左) と六角形 (右)。各辺の番号は、LAT パラメータを定義する際の面の順番。

はじめに、四角形 (LAT=1) の lattice 構造を利用した例を考えます。

#### List 4.13 ● [cell] セクションの例題 (6)

```

1: [ M a t e r i a l ]
2:   mat[1] 1H 2 160 1
3: [ C e l l ]
4:   1 0      11 -12 13 -14 15 -16 FILL=1
5:  101 0      -26 25 -22 21 LAT=1 U=1 FILL=2
6:  201 1 1.0 -90 U=2
7:   2 -1      #1
8: [ S u r f a c e ]
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 17 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となります。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 18 のようになります。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。

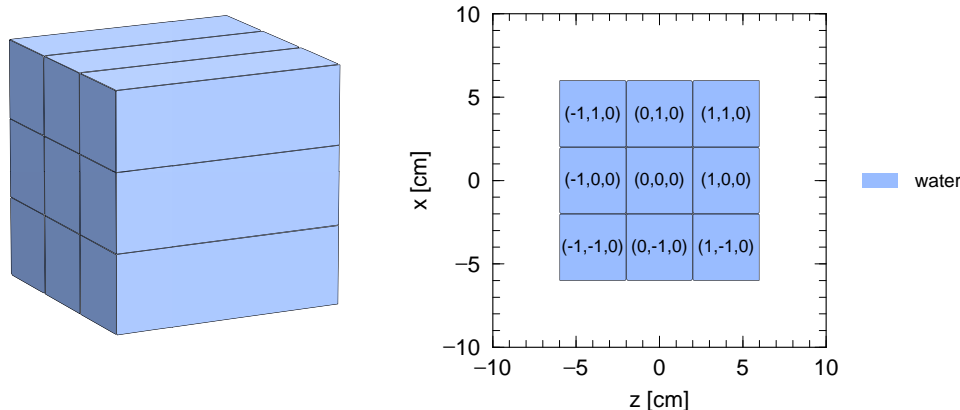


図 18: 例題 (6) の空間を立体的に見た図 (左) とこれを  $xz$  平面で切り取った断面図 (右)。各格子は、中に示された座標によって区別される。

図 18 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 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 17 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 18 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 5.1.2 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。

List 4.14 ● **[cell]** セクションの例題 (7)

```

1:  [ M a t e r i a l ]
2:  mat[1] 1H 2 160 1
3:  [ C e l l ]
4:  1 0 11 -12 13 -14 15 -16 FILL=1
5:  101 0 -31 32 -33 34 -35 36 -24 23 LAT=2 U=1 FILL=2
6:  201 1 1.0 -90 U=2
7:  2 -1 #1
8:  [ S u r f a c e ]
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  $2\text{cm}$  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  $\text{pi}$  (円周率  $\pi$ ) および面記号  $\text{PZ}$ ,  $\text{P}$  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 17 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  $4\text{cm}$ 、一辺が  $2\text{cm}$  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 が入っていることとなります。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 19 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きりません。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 17 の右図において、辺  $2 \rightarrow 1$  の方向が  $s$ 、 $4 \rightarrow 3$  が  $t$  の方向となります。 $u$  についても同様で、面番号  $24 \rightarrow 23$  の順番であれば  $y$  軸方向に、 $23 \rightarrow 24$  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  $\text{mesh=reg}$  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 5.1.2 節をご覧ください。なお lattice 座標  $(s, t, u)$  は、 $[\text{t-gshow}]$  タリーにおいて  $\text{output}=7$  or  $8$  を指定することで表示させることができます。

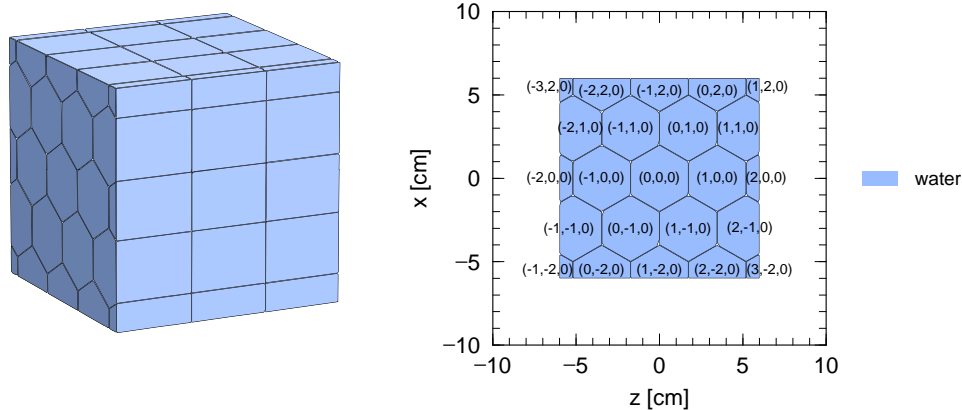


図 19: 例題 (7) の空間を立体的に見た図 (左) とこれを  $xz$  平面で切り取った断面図 (右)。各格子は、中に示された座標によって区別される。

### 4.5.5 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。4.5.4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。

#### 4.5.5.1 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 55 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られています。

List 4.15 ● [cell] セクションの例題 (8)

```

1: [Material]
2:   mat[1] 1H 2 160 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 20 のようになります。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 20 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。

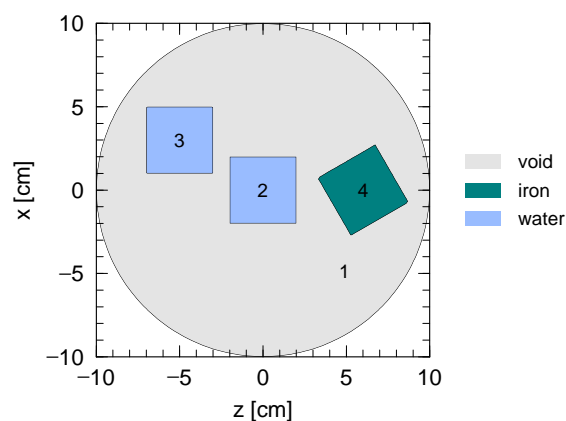


図 20: 例題 (8) の空間を  $xz$  平面で切り取った断面図。セル番号 1 の円柱の中にセル番号 2, 3, 4 が入っている。

## 4.5.5.2 Lattice を用いた階層構造

4.5.4 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。

## List 4.16 ● [cell] セクションの例題 (9)

```

1: [Material]
2:   mat[1] 1H 2 160 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、 $(-1, -1, 0), (0, -1, 0), (1, -1, 0), (-1, 0, 0), \dots$  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。

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となります。結果を示したのが図 21 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  $(0, 0, 0)$

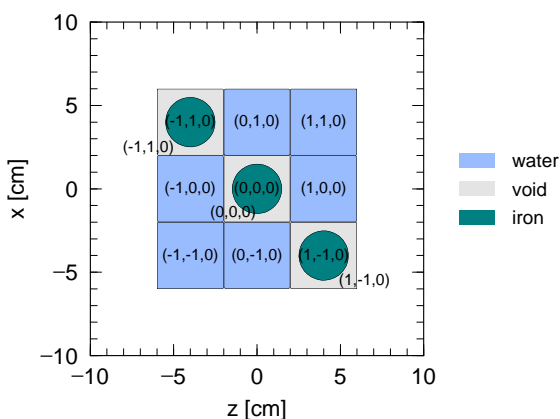


図 21: 例題 (9) の空間を  $xz$  平面で切り取った断面図。9 つの四角柱の内、3 つが鉄柱を中心に配した内部構造をもっている。

に含まれているセル番号 302 の領域” という意味で、図 21 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 5.1.2 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。

List 4.17 ● [cell] セクションの例題 (10)

```

1: [ M a t e r i a l ]
2:   mat[1] 1H 2 160 1
3:   mat[2] Fe 1
4: [ C e l l ]
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: [ S u r f a c e ]
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 22 のようになります。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 22 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

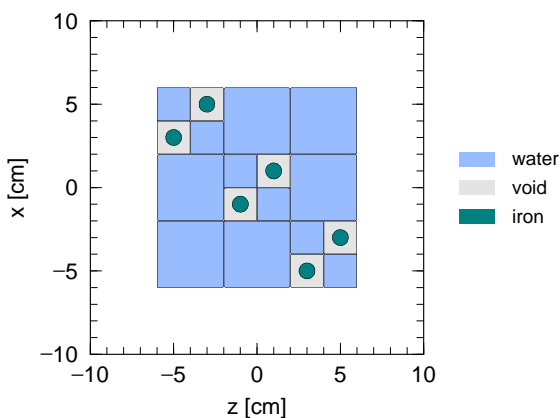


図 22: 例題 (10) の空間を  $xz$  平面で切り取った断面図。9 つの四角柱の内 3 つが構造をもち、更に 4 つの四角柱で構成されている。



いては 5.1.2 節をご覧ください。

### 4.5.5.3 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  $L_{000} L_{100} L_{200} \dots L_{stu} \dots$  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。

#### List 4.18 ● [cell] セクションの例題 (11)

```

1:  [ M a t e r i a l ]
2:   mat[1] 1H 2 160 1
3:   mat[2] Fe 1
4:  [ C e l l ]
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
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
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 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: [ S u r f a c e ]
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 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あり、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  $(-2, -2, -2), (-1, -2, -2), \dots, (2, 2, 2)$  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 23 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。少しずつ異なりますが、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 5.1.2 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 23: 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)。

*PhiTs* では、最初に取り出した入力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持ちます。*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* の入力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。

## 4.6 [Surface] セクション

### 4.6.1 書式

このセクションでは、入力ファイルの面 (surface) の定義を行います。コメント文字として **C** も使えます。ただし、ただし、**C** 以外のコメント文字としては **\$** のみ用いることができます。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、面番号、座標変換番号、面記号、面定義数値の順番です。面定義数値では定数や数式が使えます。マクロボディーによる面の定義も可能です。面番号は *PhiTs* の制約で 1 ~ 999999 までです。MCNP と同様、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。

書式と例題は以下の通りです。

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

表 56: 面記述書式

項目	説明
面番号	1 ~ 999999
座標変換番号	座標変換無しなら入力しない。 座標変換がある場合は、[transform] セクションの TRn の番号 n
面記号	面カード表の面記号もしくはマクロボディーの記号
面定義数値	面により 1 個から 15 個の入力値

List 4.19 ● [surface] セクションの例題 (1)

```

1:  [surface]
2:   1  cz   5.0
3:   2  cz  10.0
4:   3  cz  15.0
5:   4  cz  20.0
6:   5  pz   0.0
7:   6  pz   5.0
8:   7  pz  10.0
9:   8  pz  15.0
10:  9  pz  55.0
11: 10  pz  60.0

```

表 57: 面カード

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	$z - D = 0$	$D$
SO	球	中心は原点	$x^2 + y^2 + z^2 - R^2 = 0$	$R$
S		汎用	$(x - \bar{x})^2 + (y - \bar{y})^2 + (z - \bar{z})^2 - R^2 = 0$	$\bar{x} \bar{y} \bar{z} R$
SX		中心が X 軸上	$(x - \bar{x})^2 + y^2 + z^2 - R^2 = 0$	$\bar{x} R$
SY		中心が Y 軸上	$x^2 + (y - \bar{y})^2 + z^2 - R^2 = 0$	$\bar{y} R$
SZ		中心が Z 軸上	$x^2 + y^2 + (z - \bar{z})^2 - R^2 = 0$	$\bar{z} R$
C/X	円柱	X 軸に平行	$(y - \bar{y})^2 + (z - \bar{z})^2 - R^2 = 0$	$\bar{y} \bar{z} R$
C/Y		Y 軸に平行	$(x - \bar{x})^2 + (z - \bar{z})^2 - R^2 = 0$	$\bar{x} \bar{z} R$
C/Z		Z 軸に平行	$(x - \bar{x})^2 + (y - \bar{y})^2 - R^2 = 0$	$\bar{x} \bar{y} R$
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	$\sqrt{(y - \bar{y})^2 + (z - \bar{z})^2} - t(x - \bar{x}) = 0$
K/Y	Y 軸に平行		$\sqrt{(x - \bar{x})^2 + (z - \bar{z})^2} - t(y - \bar{y}) = 0$	$\bar{x} \bar{y} \bar{z} t^2 \pm 1$
K/Z	Z 軸に平行		$\sqrt{(x - \bar{x})^2 + (y - \bar{y})^2} - t(z - \bar{z}) = 0$	$\bar{x} \bar{y} \bar{z} t^2 \pm 1$
KX	X 軸上		$\sqrt{y^2 + z^2} - t(x - \bar{x}) = 0$	$\bar{x} t^2 \pm 1$
KY	Y 軸上		$\sqrt{x^2 + z^2} - t(y - \bar{y}) = 0$	$\bar{y} t^2 \pm 1$
KZ	Z 軸上		$\sqrt{x^2 + y^2} - t(z - \bar{z}) = 0$	$\bar{z} t^2 \pm 1$
				$\pm 1$ は 1 シートの円錐のみ指定
SQ	楕円 双曲面 放物面	X, Y, または Z 軸に 平行な軸	$A(x - \bar{x})^2 + B(y - \bar{y})^2 + C(z - \bar{z})^2 + 2D(x - \bar{x}) + 2E(y - \bar{y}) + 2F(z - \bar{z}) + G = 0$	$A B C D E$ $F G \bar{x} \bar{y} \bar{z}$
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	$(x - \bar{x})^2/B^2 + (\sqrt{(y - \bar{y})^2 + (z - \bar{z})^2} - A)^2/C^2 - 1 = 0$	$\bar{x} \bar{y} \bar{z} A B C$
TY			$(y - \bar{y})^2/B^2 + (\sqrt{(x - \bar{x})^2 + (z - \bar{z})^2} - A)^2/C^2 - 1 = 0$	$\bar{x} \bar{y} \bar{z} A B C$
TZ			$(z - \bar{z})^2/B^2 + (\sqrt{(x - \bar{x})^2 + (y - \bar{y})^2} - A)^2/C^2 - 1 = 0$	$\bar{x} \bar{y} \bar{z} A B C$
P	平面	座標により 定義された面		$x_1, y_1, z_1, x_2, y_2, z_2,$ $x_3, y_3, z_3$

円錐の場合、 $\bar{x}, \bar{y}, \bar{z}$  等で指定する頂点を中心に、各軸上に沿って上下に 2 つの円錐面が定義されます。 $\pm 1$  の部分の入力を、1 とすると軸の上側に広がる面のみを、 $-1$  とすると軸の下側に広がる面のみを、何の値も与えないとその両方を対象とします。ただし、片方の円錐面しか使用しない場合、[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。

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。

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。

## 4.6.2 マクロボディー

表 58: マクロボディーカード

記号	型式	入力数値	解説
BOX	任意の BOX 平面 (全ての角が 90 度)	$V_x V_y V_z$ $A1x A1y A1z$ $A2x A2y A2z$ $A3x A3y A3z$	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 xyz 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)	$V_x V_y V_z$ $R$	中心座標 球の半径
RCC	円柱	$V_x V_y V_z$ $H_x H_y H_z$ $R$	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	$v1 v2 v3$ $h1 h2 h3$ $r1 r2 r3$ $s1 s2 s3$ $t1 t2 t3$	基点の座標 基点から高さベクトル 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	$V_x V_y V_z$ $H_x H_y H_z$ $V_{1x} V_{1y} V_{1z}$ $V_{2x} V_{2y} V_{2z}$	シリンダーの底面の中心の座標 シリンダーの高さベクトル ( $\vec{H}$ ) $\vec{H}$ に垂直な楕円の長軸ベクトル ( $\vec{V}_1$ ) $\vec{H}$ と $\vec{V}_1$ に垂直な楕円の短軸ベクトル ( $\vec{V}_2$ )
TRC	カットされた円錐	$V_x V_y V_z$ $H_x H_y H_z$ $R_1$ $R_2$	円錐の底面の中心の座標 円錐の高さベクトル 底面の半径 上面の半径
ELL	楕円体	$Rm > 0$ のとき、 $V_{1x} V_{1y} V_{1z}$ $V_{2x} V_{2y} V_{2z}$ $Rm$ $Rm < 0$ のとき、 $V_{1x} V_{1y} V_{1z}$ $V_{2x} V_{2y} V_{2z}$ $Rm$	第 1 焦点の座標 第 2 焦点の座標 長軸の長さ  楕円体の中心 長軸のベクトル 短軸の長さ
WED	くさび形	$V_x V_y V_z$ $V_{1x} V_{1y} V_{1z}$ $V_{2x} V_{2y} V_{2z}$ $V_{3x} V_{3y} V_{3z}$	頂点の座標 第 1 三角面へのベクトル ( $\vec{V}_1$ ) 第 2 三角面へのベクトル ( $\vec{V}_2$ ) 高さベクトル ( $\vec{V}_3$ )

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。

## List 4.20 ● [surface] セクションの例題 (2) [cell] セクション例題 (2) に対応

```

1: [surface]
2: 1 rpp -15 15 -5 5 0 60
3: 2 rpp -5 5 -5 5 0 20
4: 4 rpp -15 15 -5 5 0 20
5: 5 rpp -20 20 -5 5 0 40
6: 6 rpp -20 20 -5 5 0 20
7: 7 rpp -20 20 -5 5 40 60
8: 3 c/y 0 10 4

```

## 4.6.3 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。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。その時は、マクロボディー番号の次にドットを入れ、それぞれの面の番号を付けます。それぞれの面の番号は以下の通りです。

表 59: マクロボディー面番号

記号	面番号	解説
BOX	1	$A1x A1y A1z$ の終点到垂直な面
	2	$A1x A1y A1z$ の始点到垂直な面
	3	$A2x A2y A2z$ の終点到垂直な面
	4	$A2x A2y A2z$ の始点到垂直な面
	5	$A3x A3y A3z$ の終点到垂直な面
	6	$A3x A3y A3z$ の始点到垂直な面
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		通常の面と同じ
RCC	1	半径 $R$ の円柱の側面
	2	$H_x H_y H_z$ の終点到垂直な面
	3	$H_x H_y H_z$ の始点到垂直な面
RHP or HEX	1	$r1 r2 r3$ の終点到垂直な面
	2	面 1 の対面
	3	$s1 s2 s3$ の終点到垂直な面
	4	面 3 の対面
	5	$t1 t2 t3$ の終点到垂直な面
	6	面 5 の対面
	7	$h1 h2 h3$ の終点到垂直な面
	8	$h1 h2 h3$ の始点到垂直な面
REC	1	楕円のシリンダー面
	2	$H_x H_y H_z$ の終点到垂直な面
	3	$H_x H_y H_z$ の始点到垂直な面
TRC	1	円錐面
	2	$H_x H_y H_z$ の終点到垂直な面
	3	$H_x H_y H_z$ の始点到垂直な面
ELL		普通の面として定義される
WED	1	底面と上面の斜辺を含む面
	2	$\vec{v}_2$ と $\vec{v}_3$ を含む面
	3	$\vec{v}_1$ と $\vec{v}_3$ を含む面
	4	$\vec{v}_1$ と $\vec{v}_2$ 及び $\vec{v}_3$ の終点を含む面
	5	$\vec{v}_1$ と $\vec{v}_2$ 及び $\vec{v}_3$ の始点を含む面

## 4.7 [Transform] セクション

### 4.7.1 書式

このセクションでは、座標変換カードの定義を行います。コメント文字として **C** も使えます。**C** 以外のコメント文字としては **\$** のみ用いることができます。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。ただし、**R**, **I**, **M**, **J** などの繰り返し、内挿、倍増、飛び越しのオペレーターは使えません。

このセクションで定義した座標変換は、ソース、[surface] セクション、[cell] セクション、タリーの r-z、xyz メッシュ、磁場の定義などで使うことができます。

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
```

表 60: 座標変換記述書式

項目	説明
<i>n</i>	座標変換番号 1 ~ 9999 * <i>TRn</i> は、 <i>B<sub>i</sub></i> が角度の余弦ではなく degree 単位の角度であることを意味する。
<i>O<sub>1</sub> O<sub>2</sub> O<sub>3</sub></i>	変換の転置ベクトル
<i>B<sub>1</sub> ~ B<sub>9</sub></i>	変換の回転行列
<i>M</i>	= 1 は、転置ベクトルが主座標系で定義した補助座標系の位置であることを意味する。 = -1 は、転置ベクトルが補助座標系で定義した主座標系の位置であることを意味する。

デフォルト値は、

```
TRn  0 0 0 1 0 0 0 1 0 0 0 1 1
```

### 4.7.2 座標変換の定義

回転行列、転置行列による座標変換の定義は以下の通りです。

$M = 1$  のとき

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_2 & B_3 \\ B_4 & B_5 & B_6 \\ B_7 & B_8 & B_9 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix}$$

$M = -1$  のとき

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} B_1 & B_2 & B_3 \\ B_4 & B_5 & B_6 \\ B_7 & B_8 & B_9 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} - \begin{pmatrix} B_1 & B_4 & B_7 \\ B_2 & B_5 & B_8 \\ B_3 & B_6 & B_9 \end{pmatrix} \begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix}$$

ここで、

$$B_1 = \cos(x', x)$$

$$B_2 = \cos(x', y)$$

$$B_3 = \cos(x', z)$$

$$B_4 = \cos(y', x)$$

$$B_5 = \cos(y', y)$$

$$B_6 = \cos(y', z)$$

$$B_7 = \cos(z', x)$$

$$B_8 = \cos(z', y)$$

$$B_9 = \cos(z', z)$$

です。直感的には、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。

List 4.21 ● [transform] セクションの例題 1

```

1:  [ T r a n s f o r m ]
2:  *tr1      0.0000000E+00  0.0000000E+00  1.4000000E+02
3:           1.3500000E+02  9.0000000E+01  4.5000000E+01
4:           9.0000000E+01  0.0000000E+00  9.0000000E+01
5:           2.2500000E+02  9.0000000E+01  1.3500000E+02      1
6:
7:  *tr2      0.0000000E+00  0.0000000E+00  2.5800000E+02
8:           3.0000000E+02  9.0000000E+01  2.1000000E+02
9:           9.0000000E+01  0.0000000E+00  9.0000000E+01
10:          3.0000000E+01  9.0000000E+01  3.0000000E+02      1

```

上の例題では、**tr1** は  $y$  軸周りに  $135$  度回転し  $z$  方向に  $140\text{cm}$  移動する座標変換、**tr2** は  $y$  軸周りに  $300$  度回転し  $z$  方向に  $258\text{cm}$  移動する座標変換となっています。ここでは \* が付いているため、 $B_i$  ( $i = 1, \dots, 9$ ) には余弦 (cos) の値ではなく回転角度 (単位: 度, degree) を入れることになります。



## List 4.22 ● [transform] セクションの例題 2

```
1: [ T r a n s f o r m ]
2: set: c10[90] $ angle of around Z (degree)
3: set: c20[30] $ angle of around Y (degree)
4: set: c30[0] $ angle of around X (degree)
5:
6: tr1  0 0 0
7:      cos(c10/180*pi)*cos(c20/180*pi)
8:      sin(c10/180*pi)*cos(c30/180*pi)+cos(c10/180*pi)*sin(c20/180*pi)*sin(c30/180*pi)
9:      sin(c10/180*pi)*sin(c30/180*pi)-cos(c10/180*pi)*sin(c20/180*pi)*cos(c30/180*pi)
10:     -sin(c10/180*pi)*cos(c20/180*pi)
11:     cos(c10/180*pi)*cos(c30/180*pi)-sin(c10/180*pi)*sin(c20/180*pi)*sin(c30/180*pi)
12:     cos(c10/180*pi)*sin(c30/180*pi)+sin(c10/180*pi)*sin(c20/180*pi)*cos(c30/180*pi)
13:     sin(c20/180*pi)
14:     -cos(c20/180*pi)*sin(c30/180*pi)
15:     cos(c20/180*pi)*cos(c30/180*pi)
16:     1
```

上の例題では、 $z$  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、任意の回転を定義できるので便利です。

## 4.8 [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 **itype** 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、

```
1: [ Importance ]
2:   reg          imp
3: ( 5 6 7 < 11 ) 2.0
4: ( 5 6 7 < 12 ) 4.0
5: ( 5 6 7 < 13 ) 8.0
6: ( 11 12 13 )  1.0
```

で定義される `importance` と、

```
1: [ Importance ]
2:   reg          imp
3: ( 5 6 7 )      1.0
4:   11           2.0
5:   12           4.0
6:   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。

## 4.9 [ 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、

```
[ Weight Window ]
  part =  proton  neutron
  eng =  5
  ( tim =  5 )
      6.00e-7  3.98e-1  1.00e+0  7.00e+0  5.00e+4
  reg
      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 **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 **itype** 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 ( $e_1, e_2, e_3, \dots$ ) 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。

領域番号 (**reg**) 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、最初の領域番号 (**reg**) のコラムは、省略できます。また、領域番号の並びは、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。**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。

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。

## 4.10 [Volume] セクション

このセクションでは、GG の cell の体積 ( $\text{cm}^3$ ) 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。両方で定義されない領域は、 $1.0 \text{ cm}^3$  にセットされます。書式は、

[ Volume ]	
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。

### 4.11 [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。両方で定義されない領域は、 $2.53 \times 10^{-8}$  MeV にセットされます。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。

## 4.12 [ Brems Bias ] セクション

このセクションでは、物質毎に **bremsstrahlung** 過程のエネルギーバイアスを定義します。**bremsstrahlung** 過程では、大量の低エネルギー光子が生成されますが、しばしば、高エネルギー光子により関心がある場合があります。そのような時に、エネルギー毎のバイアスを定義して、目的のエネルギー領域の光子の統計を上げることができます。

書式は、

```
[ Brems Bias ]
material = all or number of materials
          m1 m2 m3 .....
num      bias
{1-45}   1.0
46       2.0
47       3.0
48       4.0
49       5.0
```

まず、バイアスを設定する物質を定義します。**all** の場合は次の行は必要ありません。数を指定した場合は、その数だけの物質番号を次の行に定義します。

バイアスは、1 から 49 のグループについて定義します。大きい番号が高エネルギーを表します。バイアス値は相対値です。

グループ番号 (**num**) とバイアス (**bias**) の順番を変えたいときは、**bias num** とします。読み飛ばしコラム用の **non** も使えます。

### 4.13 [ Photon Weight ] セクション

このセクションでは、GG の cell の光子生成ウエイトを定義します。

セル  $i$  の光子生成ウエイトの値  $W_i$  が正の時は、生成 photon のウエイトが  $W_i \times I_s/I_i$  以上の時に生成されます。それ以下の時はロシアンルーレットが施されます。ここで  $I_s, I_i$  は、ソースセルと生成場所の中性子のインポートランスです。

セル  $i$  の光子生成ウエイトの値  $W_i$  が負の時は、生成 photon のウエイトが  $W_i \times W_s \times I_s/I_i$  以上の時に生成されます。それ以下の時はロシアンルーレットが施されます。ここで  $W_s$  は、生成する中性子の反応前のウエイトです。

もし、 $W_i = 0$  なら、1 個の光子が中性子反応から生成されます。もし、 $W_i = -1.0e6$  であれば、このセルでは光子の発生は無視されます。デフォルト値は、 $W_i = -1$  です。

[ Photon Weight ]	
reg	pwt
1	0.1
11	0.3
( { 2 - 5 } 8 9 )	0.5
( 11 12 15 )	0.1
16	0.9
....	.....
....	.....

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) とウエイト (pwt) の順番を変えたいときは、pwt 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。



## 4.14 [ 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、 $(1 - \text{透過確率})$  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 **itype** 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| ≤ 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てやらないと、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。注意して下さい。

## 4.15 [Magnetic Field] セクション

### 4.15.1 荷電粒子

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、パラメータセクションで、**imagnf=1** を指定しなければなりません。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 ( ) で括弧してください。

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、散乱が起きます。

**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。

[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場合があります。`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。

#### 4.15.2 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。このサンプルを動かすインプットファイル `sex03b.in` を添付しますので、参考にして下さい。

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、 $\phi_+$  と  $\phi_-$  は、スピンの磁場に平行な粒子数と反平行な粒子数です。

## 4.16 [ Electro Magnetic Field ] セクション

このセクションでは、指定した空間に一樣な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。電場と磁場は任意に併存させることができます。ただし、設定できるのは共に一樣な場のみです。**[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。

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。書式は以下のようになります。

[ Electro Magnetic Field ]				
reg	elf	mgf	trcle	trclm
1	100	1	1	2
2	100	1	1	2

**[t-track]** タリーを用いて電磁場中の粒子の軌跡をプロットする場合は、**[parameters]** セクションにおいて **itstep=1** とすると、より正確なタリー結果を得られるようになります。

**[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場合があります。**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。

### 4.17 [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、カウンターの進捗 (-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、

```
[ 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 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、つまりカウントなしと見なされます。数字は、カウンターの進捗を表します。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。

## 4.18 [ Reg Name ] セクション

このセクションでは、[t-gshow]、[t-gshow]、[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。このため、名前の中で括弧 { } は使えません。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 30 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。

### 4.19 [ 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。名前として定義できる文字数の上限は 30 です。

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 61: 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 62: 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 63: 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

## 4.20 [ 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。

## 4.21 [ 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。

$$R = \begin{cases} R_0 & \text{if } Q \leq Q_c \\ \frac{1}{2}R_0(1 - \tanh[(Q - mQ_c)/W])(1 - \alpha(Q - Q_c)) & \text{if } Q > Q_c \end{cases}$$

ここで  $Q$  は、散乱ベクター ( $\text{\AA}^{-1}$ ) 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  $\alpha$  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 ( $\text{\AA}^{-1}$ )、 $\alpha$  を am ( $\text{\AA}$ )、 $W$  を wm ( $\text{\AA}^{-1}$ ) 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。

## 4.22 [ 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。

### 4.23 [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。

#### 4.24 [ 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 ( $\delta$  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  $\delta$  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。 $\delta$  線生成断面積は、Butts と Katz の式<sup>11</sup> 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

真空 (void) 以外の領域毎に、 $\delta$  線が発生させるしきい値エネルギー  $E_{th}$  (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  $\delta$  線を実際に 2 次粒子として発生させます。 $E_{th}$  以下の  $\delta$  線の寄与は通常の LET で評価しており、設定できる  $E_{th}$  の最小値は 0.001MeV (=1keV) です。ただし、 $E_{th}$  を 10keV 以下に設定したり、 $\delta$  線が発生させる領域の厚さをおおよそ  $10\mu\text{g}/\text{cm}^2$  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ください。具体的には、荷電粒子の実効的な阻止能が、 $\delta$  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。[delta ray] セクションを利用しない場合は、 $E_{th}$  にデフォルトの値として  $1.e+10$  が入っており、事実上  $\delta$  線は発生しません。領域番号と  $E_{th}$  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。

<sup>11</sup> J. J. Butts and R. Katz, "Theory of RBE for Heavy Ion Bombardment of Dry Enzymes and Viruses", Radiation Research 30, 855-871 (1967).

## 4.25 [ Multiplier ] セクション

このセクションで定義した係数は、[t-track] セクションの **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、[t-track] タリーの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。書式は以下のようになります。

```
[ Multiplier ]
number = -201
interpolation = log
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 番号は -299 から -200 までの負の数です。次に、この係数のテーブルから係数を抽出するときの内挿方法を **interpolation** で決定します。**log** の場合は **log-log** 内挿、**lin** のときは **linear-linear** 内挿です。そして、エネルギーの群数を **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。ただし、エネルギーのデータは昇順で書く必要があります。

定義した係数をタリーで利用する場合は、[t-track] セクションの **multiplier** サブセクションを使います。基本的な表式は (*C* *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、[t-track] セクションにおいて、以下の様な書式で使います。

```

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 [t-track] 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。



## 5 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 64: タリーセクションの種類

name	説明
[t-track]	track length タリーのパラメータを定義します。
[t-cross]	面横断タリーのパラメータを定義します。
[t-heat]	発熱タリーのパラメータを定義します。
[t-deposit]	deposit タリーのパラメータを定義します。
[t-deposit2]	deposit2 タリーのパラメータを定義します。
[t-yield]	生成粒子タリーのパラメータを定義します。
[t-product]	生成粒子タリーのパラメータを定義します。
[t-dpa]	DPA タリーのパラメータを定義します。
[t-let]	LET タリーのパラメータを定義します。
[t-sed]	SED タリーのパラメータを定義します。
[t-time]	時間タリーのパラメータを定義します。
[t-star]	star density タリーのパラメータを定義します。
[t-dchain]	dchain タリーのパラメータを定義します。
[t-userdefined]	ユーザー定義タリーのパラメータを定義します。
[t-gshow]	領域境界表示のパラメータを定義します。
[t-rshow]	物理量領域表示のパラメータを定義します。
[t-3dshow]	3次元領域表示のパラメータを定義します。

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。

### 5.1 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。この3種類のメッシュを

```
mesh = [ reg, r-z, xyz ]
```

で定義します。

#### 5.1.1 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。

### 5.1.2 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 が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  $1 \leq s \leq 2, 3 \leq t \leq 6, u = 1$  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。

#### List 5.1 ● mesh = reg の例題 (1)

```
1:      mesh = reg
2:      reg = (all)
3:      ( { 201 - 205 } )
4:      ( 161 < 160[1:2 3:6 1:1] )
5:      ( (201 202 203 204) < (161 162 163) )
6:      ( ( 90 100 ) 120 < 61 ( 62 63 ) )
```

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。

#### List 5.2 ● mesh = reg の例題 (2)

```
1:      mesh = reg          # mesh type is region-wise
2:      reg = ( all ) ( { 201 - 205 } ) ( 161 < 160[ 1:2 3:6 1:1 ] ) ( (
3:          { 201 - 204 } ) < ( { 161 - 163 } ) ) ( ( 90 100 ) 120 < 61
4:          ( 62 63 ) )
5:      volume             # combined, lattice or level structure
6:      non  reg          vol  # reg definition
7:      1  10001          8.1000E+01 # ( all )
8:      2  10002          5.0000E+00 # ( { 201 - 205 } )
9:      3  10003          1.0000E+00 # ( 161 < 160[ 1 3 1 ] )
10:     4  10004          1.0000E+00 # ( 161 < 160[ 2 3 1 ] )
11:     5  10005          1.0000E+00 # ( 161 < 160[ 1 4 1 ] )
12:     6  10006          1.0000E+00 # ( 161 < 160[ 2 4 1 ] )
13:     7  10007          1.0000E+00 # ( 161 < 160[ 1 5 1 ] )
14:     8  10008          1.0000E+00 # ( 161 < 160[ 2 5 1 ] )
15:     9  10009          1.0000E+00 # ( 161 < 160[ 1 6 1 ] )
16:    10  10010          1.0000E+00 # ( 161 < 160[ 2 6 1 ] )
17:    11  10011          4.0000E+00 # ( ( { 201 - 204 } ) < ( { 161 - 163 } ) )
18:    12  10012          2.0000E+00 # ( ( 90 100 ) < 61 )
19:    13  10013          1.0000E+00 # ( 120 < 61 )
```

```
20:      14 10014  2.0000E+00 # ( ( 90 100 ) < ( 62 63 ) )
21:      15 10015  1.0000E+00 # ( 120 < ( 62 63 ) )
```

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として組み込まれています。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。このような場合は、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。

### 5.1.3 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。

### 5.1.4 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。

## 5.2 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。

## 5.3 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。

## 5.4 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。

## 5.5 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。

## 5.6 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。

### 5.6.1 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のです。

表 65: 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。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。

### 5.6.2 e-type = 1 の場合

書式は、

<pre> 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) </pre>
---

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。

### 5.6.3 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
```

### 5.6.4 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
```

### 5.6.5 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。

## 5.7 他のタリ一定義文

### 5.7.1 粒子定義文

タリ一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。

### 5.7.2 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せんが、

```
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 以降についての制限です。

### 5.7.3 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。**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。

### 5.7.4 resfile 定義文

再開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。

### 5.7.5 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。

### 5.7.6 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。

### 5.7.7 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。

### 5.7.8 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
```

### 5.7.9 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。

### 5.7.10 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)
```

と記載されます。*ANGEL* の使い方は、*ANGEL* のマニュアルをご覧ください。

### 5.7.11 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。

### 5.7.12 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 ANGLE 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。

### 5.7.13 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。

#### 5.7.14 x-txt, y-txt, z-txt 定義文

AN<sub>G</sub>L 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、AN<sub>G</sub>L パラメータ定義文では、変えられません。

<p>x-txt = テキスト y-txt = テキスト z-txt = テキスト</p>
---

#### 5.7.15 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 頂点と同じ物質ならスキャンしません。

#### 5.7.16 epsout 定義文

epsout = 1 を指定すると、出力ファイルを AN<sub>G</sub>L 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。

### 5.7.17 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。

### 5.7.18 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。

### 5.7.19 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 = O1 O2 O3 B1 B2 B3 B4 B5 B6 B7 B8 B9 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。

### 5.7.20 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。

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

表 66: 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 67: dump データの種類と番号 (2)

物理量	name	nocas	nobch	no
番号	17	18	19	20

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。

## 5.8 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

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。ただし、バージョン 2.74 で本機能を利用できるのは `[t-track]` と `[t-deposit]` (`output=deposit` の場合は不可) のみです。また、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。

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。

Sumtally サブセクションで使用できるパラメータは表 68 の通りです。

表 68: sumtally サブセクションパラメータ

name	値	説明
<code>isumtally =</code>	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
<code>nfile =</code> (次行)	数 <code>file name</code> 数値	タリーファイル数 タリーファイル名、重み付けの値 <code>isumtally=2</code> の場合は自動的に規格化されます
<code>sfile</code>	<code>file name</code>	<code>sumtally</code> 機能で足しあわせた結果の出力ファイル名
<code>sumfactor</code>	(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。

### List 5.3 ● sumtally 機能を用いた例題 (`isumtally=1`) の場合)

```

1: sumtally start
2: isumtally = 1      $(D=1) sumtally option, 1:integration, 2:weighted sum
3: nfile = 2        $ number of tally files
4:   result-1.out  1.0
5:   result-2.out  1.0
6: sfile=result-s.out $ file name of output by sumtally option
7: sumfactor = 1.0   $ (D=1.0) normalization factor
8: sumtally end

```



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ェイトの平均値を変えられることになります (`[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計算を実行することが可能です。

`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。

$$\bar{X} = F \sum_{j=1}^N \frac{r_j}{r} \bar{X}_j \quad (2)$$

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  $\sigma_X$  は、 $j$  番目のタリー結果の標準偏差を  $\sigma_{X_j}$  とし、

$$\sigma_X = F \sqrt{\sum_{j=1}^N \left(\frac{r_j}{r}\right)^2 \sigma_{X_j}^2} \quad (3)$$

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。

**List 5.4 ● sumtally 機能を用いた例題 (`isumtally=2`) の場合)**

```

1: sumtally start
2: isumtally = 2      $(D=1) sumtally option, 1:integration, 2:weighted sum
3: nfile = 2         $ number of tally files
4:   result-l.out  2.0
5:   result-r.out  3.0
6: sfile=result-s.out $ file name of output by sumtally option
7: sumfactor = 5.0   $ (D=1.0) normalization factor
8: sumtally end

```

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、乱数等の再開計算に必要な情報は出力されず、再開計算ができないようになっています。

## 6 タリー入力書式

### 6.1 [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 24 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 ( $[\text{cm}^2]$  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で、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。

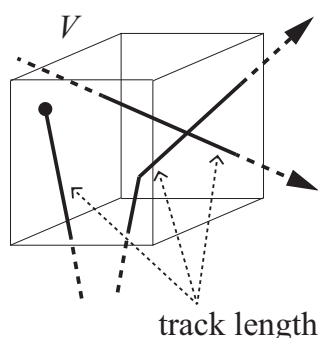


図 24: Track タリー: 空間内の飛跡 (実線) の長さを計算する。

表 69: [t-track] 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-track] 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
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

表 70: [t-track] パラメータ (2)

name	値	説明
unit =	1, 2, 3, 4  11, 12, 13, 14	1: [1/cm <sup>2</sup> /source] 2: [1/cm <sup>2</sup> /MeV/source] 3: [1/cm <sup>2</sup> /Lethargy/source] 4: [cm/source] 11: [1/cm <sup>2</sup> /nsec/source] 12: [1/cm <sup>2</sup> /nsec/MeV/source] 13: [1/cm <sup>2</sup> /Lethargy/nsec/source] 14: [cm/nsec/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。
multiplier =	物質数 (省略可)	multiplier を物質毎に指定します multiplier サブセクションの書式は下に示します
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
2d-type =	1,2,3,4,5,6,7 (省略可)	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
gshow =	0(省略時), 1, 2, 3, 4	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
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

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  $E_{\text{high}}, E_{\text{low}}$  のときに、各 Lethargy の幅を  $\ln(E_{\text{high}}/E_{\text{low}})$  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  $1\text{cm}^3$  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。

表 71: [t-track] パラメータ (3)

name	値	説明
<b>ginfo =</b>	0(省略時), 1 2	<b>gshow, rshow</b> 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 ( <b>.err</b> ) を出力。
<b>resol =</b>	1(省略時)	<b>gshow, rshow</b> オプションの時、領域境界を求める分解能を各辺 <b>resol</b> 倍します。
<b>width =</b>	0.5(省略時)	<b>gshow, rshow</b> 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。
<b>volume</b> <b>reg vol</b>	(省略可)	<b>reg</b> メッシュの時、各領域の体積を定義します。省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に <b>volume</b> 定義文が必要 <b>volume</b> 定義文。書式は 5.1.2 を参照。
<b>iechrl =</b>	72(省略時)	<b>volume</b> のインプットエコーの時の最大コラム数
<b>volmat =</b>	(省略可、D=9)	<b>xyz</b> メッシュで <b>material</b> が指定されている時、各メッシュの <b>volume correction</b> を行う。(0 で無し) <b>volmat</b> の値は、 <b>xyz</b> メッシュ1辺のスキャン数
<b>epsout =</b>	0(省略時), 1	1 で出力ファイルを <b>AnGEL</b> で処理した <b>eps</b> ファイルを作成。ファイル名は出力ファイルの拡張子を <b>eps</b> に変えたファイル名
<b>ctmin(i) =</b> <b>ctmax(i) =</b>	(省略可、D=-9999) (省略可、D= 9999)	<b>i-th</b> カウンターの最小値 <b>i-th</b> カウンターの最大値
<b>trcl =</b>	(省略可)	<b>r-z, xyz</b> メッシュの座標変換番号もしくは座標変換定義
<b>gslat =</b>	1(省略時) 0	1 で <b>gshow</b> で <b>lattice</b> の境界を表示 0 で <b>gshow</b> で <b>lattice</b> の境界を非表示

**multiplier** サブセクションを用いて、本タリーの結果にエネルギーに依存した係数を乗じることができます。任意の関数を用意する場合は [**multiplier**] セクションを使用します。係数のセットを表す ID 番号を  $k$  として (ただし  $k < 0$ )、基本的に ( $C \ k$ ) の様な表式を用いて指定します。ここで、 $C$  は規格化定数です。**multiplier** サブセクションの書式は以下の通りです。

<b>multiplier = number of material</b>			
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 [t-track] 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、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。他に、PHITS に内蔵されている線量換算係数のセットが利用できます。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。<sup>12</sup> 線量換算係数の単位は ( $\mu\text{Sv/h}/(\text{n/sec/cm}^2)$ ) です。(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、以下に示す MCNP の FM カードと同様の書式も利用できます。

```

multiplier = number of material
part = proton
emax = 150

mat          mset1  mset2
1  ( 0.1236 1 1 -4 ) ( 0.0 )
2  ( 0.0060 2 1 -4 ) ( 0.0 )
3  ( 0.0032 3 1 -4 ) ( 0.0 )
....
....

multiplier = number of material
part = neutron
emax = 150

mat          mset1          mset2
1  ( 0.1236 1 1 -4 : -6 -8 ) ( 1.0 -1 33 0.543 )
2  ( 0.0060 2 1 -4 : -6 -8 ) ( 1.0 -1 34 0.321 )
3  ( 0.0032 3 1 -4 : -6 -8 ) ( 1.0 -1 35 0.678 )
....
....

```

この例題の mset1 は、heat をとるセットです。また、mset2 は、陽子がゼロで、中性子は attenuator セットとなっています。

<sup>12</sup> 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)

## 6.2 [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  $\theta$  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異なることとなり (図 25 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 2 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。

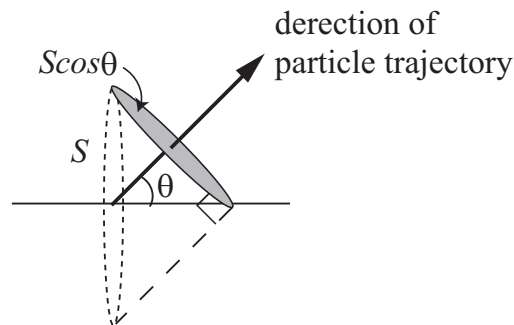


図 25: 指定した面の面積  $S$  と粒子の運動の方向に対して有効な面素  $S \cos\theta$  の関係。

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 ( $[\text{cm}^2]$  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 72: [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
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 <sup>2</sup> /source] 2: [1/cm <sup>2</sup> /MeV/source] 3: [1/cm <sup>2</sup> /Lethargy/source] 4: [1/cm <sup>2</sup> /sr/source] 5: [1/cm <sup>2</sup> /MeV/sr/source] 6: [1/cm <sup>2</sup> /Lethargy/sr/source] 11: [1/cm <sup>2</sup> /nsec/source] 12: [1/cm <sup>2</sup> /MeV/nsec/source] 13: [1/cm <sup>2</sup> /Lethargy/nsec/source] 14: [1/cm <sup>2</sup> /sr/nsec/source] 15: [1/cm <sup>2</sup> /MeV/sr/nsec/source] 16: [1/cm <sup>2</sup> /Lethargy/sr/nsec/source]
axis =	eng, reg, x, y, z, r, cos, the, t xy, yz, xz	出力データの x 軸 角度 (cos, the), 時間 (t) 2次元表示

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  $\theta$  毎の current もカウントできます。図 26 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。

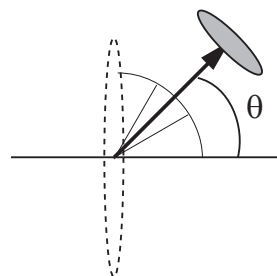


図 26: 角度メッシュを用いた current の角度分布の評価。

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  $E_{\text{high}}, E_{\text{low}}$  のときに、各 Lethargy の幅を  $\ln(E_{\text{high}}/E_{\text{low}})$  で与えます。

表 73: [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
2d-type =	1,2,3,4,5,6,7 (省略可)	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 =	(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
gshow =	0(省略時) 1, 2, 3, 4	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
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 =	1(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。
width =	0.5(省略時)	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	1 で出力ファイルを $ANGL$ 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
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。
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



`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。

[t-cross] で、`dump` を用いることができるのは、`reg` メッシュだけです。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。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。そのためには、`e-type, a-type, t-type` のメッシュをひとつにすると便利です。(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。)

形状メッシュで、`mesh = reg` を選択した場合、それに引き続く形状メッシュサブセクションは、領域横断面を定義するため、領域番号を用いて入ってくる領域、出て行く領域、またその面積 (cm<sup>2</sup>) を次の書式で定義する必要があります。

```

mesh = reg
reg = number of crossing surfaces
  r-in  r-out  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
  ...    ...    ....
  ...    ...    ....

```

このとき、データの並びは、`r-in r-out area` がデフォルトで、この定義文は省略できますが、順番を変えたいときは、`area r-in r-out` のように定義文を明示します。読み飛ばしコラム用の `non` も使えます。領域をまとめる、( { 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。

`mesh = reg` の時は、上の様に入る領域と出る領域を指定しますので、`flux` を得る時は注意が必要です。上の例題の 3 番目の領域以外は、片方向の `flux` になります。従って、`flux` を取る時は、3 番目の例題のように隣り合う領域をそれぞれ `r-in r-out` に合わせて指定して下さい。

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。

### 6.3 [T-Heat] セクション

生成熱を出力します。核データを用いる低エネルギーの中性子、光子、電子による発熱も、Kerma factor を用いて同時にタリーすることができます。e-mode=1 を用いると、核データを用いた中性子の Kerma factor は用いず、放出荷電粒子や、残留核の運動エネルギーから発熱を計算します。光子の発熱は、通常は核データの Kerma factor から計算しますが、electron=1、かつ、電子の輸送を含んでいる時は、電子のエネルギーロスから計算します。

表 74: [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 =  (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
output =	heat simple	total: 全発熱量 total: 全発熱量 ncut, gcut, pcut: 計算打切エネルギー以下の中性子、光子、陽子の発熱量。(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: 全発熱量 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ロスによる発熱量。 low neutron: 中性子カーマ係数を用いた際の発熱量。 photon: 光子カーマ係数を用いた際の発熱量。 ただし、electron=1 のときは、カットオフ電子による発熱量。 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  $e_{min}$  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。

表 75: [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 $d, t, {}^3\text{He}$ , $\alpha$ , residual nuclei の寄与。 ionization に対する $p, \pi^+, \pi^-$ , 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, $\pi^+, \pi^-$ , 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、ここで指定した粒子の荷電粒子のエネルギーロスによる発熱 (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。
unit =	0, 1, 2	0: [Gy/source] 1: [MeV/cm <sup>3</sup> /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
2d-type =	1,2,3,4,5,6,7 (省略可)	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
gshow =	0(省略時), 1, 2, 3, 4	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
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 =	1(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。
width =	0.5(省略時)	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。

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 76: [t-heat] パラメータ (3)

name	値	説明
volume	(省略可)	reg メッシュの時、各領域の体積を定義します。省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に volume 定義文が必要。
reg vol		volume 定義文。書式は 5.1.2 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	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
electron =	0(省略時), 1	電子の寄与のオプション 0: 光子の KERMA factor を使う 1: 電子の ionization loss から計算する。 この場合、電子の輸送が必要
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
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

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  $\frac{E_1}{M_1} + \frac{E_2}{M_2}$  となります。

## 6.4 [ T - Deposit ] セクション

このタリーは、[t-heat] とほぼ同等のタリーで、dose や deposit energy 分布をタリーします。[t-heat] との違いは、まず、ここでの dose は、荷電粒子のエネルギーロスだけをタリーします。このタリーでは、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。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  $Q(L)$  関係に乗じて線量当量を導出するプログラムが、usrdfn2.f には単にエネルギーロスを返すプログラムが入っています。また、時間メッシュと [Timer] を組み合わせることによって、TOF のカウンターを模擬でき、時間と deposit energy の相関を 2次元プロットすることができます。Version 2.70 より、output=deposit としてイベント毎の付与エネルギー分布を計算する際、検出器のエネルギー分解能を考慮して計算できるようになりました。

表 77: [t-deposit] 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-deposit] 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
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 $1\text{g}/\text{cm}^3$ 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 $1\text{g}/\text{cm}^3$ 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。
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 $Q(L)$ 関係に乗じて線量当量を導出するプログラムが、usrdfn2.f には単にエネルギーロスを返すプログラムが入っています。
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ロスをタリー イベント毎の deposit energy 分布をタリー e-type セクションが必要

表 78: [t-deposit] パラメータ (2)

name	値	説明
unit =	0, 1, 2, 3, 4	0: Dose [Gy/source]; output=dose のみ 1: Dose [MeV/cm <sup>3</sup> /source] 2: Dose [MeV/source] 3: Number [1/source]; output=deposit のみ 4: Number [1/nsec/source]; output=deposit のみ
axis =	eng, reg, x, y, z, r, t xy, yz, xz, rz, t-eng, eng-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
2d-type =	1,2,3,4,5,6,7 (省略可)	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
gshow =	0(省略時), 1, 2, 3, 4	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
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 =	1(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。
width =	0.5(省略時)	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、各領域の体積を定義します。 省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に volume 定義文が必要。 volume 定義文。書式は 5.1.2 を参照。
iechr1 =	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 79: [t-deposit] パラメータ (3)

name	値	説明
epsout =	0(省略時), 1	1 で出力ファイルを <i>ANGEL</i> で処理した <b>eps</b> ファイルを作成。ファイル名は出力ファイルの拡張子を <b>eps</b> に変えたファイル名
ctmin(i) =	(省略可、D=-9999)	<b>i-th</b> カウンターの最小値
ctmax(i) =	(省略可、D= 9999)	<b>i-th</b> カウンターの最大値
trcl =	(省略可)	<b>r-z, xyz</b> 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	0 で <b>gshow</b> で <b>lattice</b> の境界を非表示、1 で表示
dresol	(省略可、D=0.0)	検出器の分解能を表すパラメータ 1。 <b>output = deposit</b> の場合のみ有効。 <b>dresol</b> = $\sigma_r$ , <b>dfano</b> =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。
dfano	(省略可、D=0.0)	検出器の分解能を表すパラメータ 2。 <b>output = deposit</b> の場合のみ有効。 <b>dresol</b> = $\sigma_r$ , <b>dfano</b> =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。

[t-deposit] では、荷電粒子のエネルギーロス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ロスを **part =** で指定することはできません。領域に入る粒子毎のエネルギーロスをタリーするには、[counter] で **part =** を用いて、特定の入射粒子から引き起こされるエネルギーロスをカウンターで指定する必要があります。

**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  $\frac{E_1}{M_1} + \frac{E_2}{M_2}$  となります。

**output = deposit** の場合、**part=** で個別に粒子を指定して得られた結果を足し上げた値と **all** を指定した結果一致しない場合がありますのでご注意ください。また、**output = dose** の場合は、**part=** で個別に粒子を指定し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す。

## 6.5 [ 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、deposit energy をタリーし、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 deposit energy の相関を2次元プロットすることなどができます。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。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 には単にエネルギーロスを返すプログラムが入っています。また、時間メッシュと [Timer] を組み合わせることによって、TOF のカウンターを模擬でき、時間と deposit energy の相関を2次元プロットすることもできます。

表 80: [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 $1\text{g/cm}^3$ 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 $1\text{g/cm}^3$ の水に対する値が参照される。詳しくは、奨励設定の <b>particletherapy</b> 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 には単にエネルギーロスを返す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 81: [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
2d-type =	1,2,3,4,5,6,7 (省略可)	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
volume reg vol	(省略可)	reg メッシュの時、各領域の体積を定義します。 省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に volume 定義文が必要。 volume 定義文。書式は 5.1.2 を参照。
iechr1 =	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	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
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

[t-deposit2] では、荷電粒子のエネルギーロス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ロスを part = で指定することはできません。領域に入る粒子毎のエネルギーロスをタリーするには、[counter] で part = を用いて、特定の入射粒子から引き起こされるエネルギーロスをカウンターで指定する必要があります。

## 6.6 [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 82: [t-yield] 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
special =	D=0 (省略可)	> 0 の時、タリー領域での核反応を <b>special</b> 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] セクション最大 6 個まで 入射粒子を限定する
material =	(省略可) all, 数	スコアする <b>material</b> を限定する。複数定義可。 <b>all</b> : 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 <b>material</b> を次の行に記述する。 負の数にした場合は、それらの <b>material</b> を対象から外すことを意味する。
(次行)	2 5 8	<b>material</b> 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。 <b>all</b> : 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。 質量数を指定すれば、その核。 質量数を指定しなければ、 <b>Pb</b> 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。
(次行)	208Pb Pb	
nucleus =	(省略可) all, 数	出力する核種生成を限定する。複数定義可。 <b>all</b> : 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。 質量数を指定すれば、その核。 質量数を指定しなければ、 <b>Pb</b> の同位体全てを指定。
(次行)	208Pb Pb	
elastic =	0, 1(省略時)	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。 0:含めない, 1:含める
unit =	1, 2	1: [1/source] 2: [1/cm <sup>3</sup> /source]
ndata =	0(省略時), 1	1 を指定すると、核子入射で標的核が $\alpha$ , <sup>14</sup> N, <sup>16</sup> O の場合の核種生成断面積に核データを用います。

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。

${}^4\text{He}(n, x){}^3\text{H}$	${}^{14}\text{N}(n, x){}^3\text{H}$	${}^{14}\text{N}(n, x){}^7\text{Be}$	${}^{14}\text{N}(n, x){}^{11}\text{Be}$	${}^{14}\text{N}(n, x){}^{10}\text{C}$	${}^{14}\text{N}(n, x){}^{11}\text{C}$
${}^{14}\text{N}(n, x){}^{14}\text{C}$	${}^{14}\text{N}(n, x){}^{13}\text{N}$	${}^{16}\text{O}(n, x){}^3\text{H}$	${}^{16}\text{O}(n, x){}^7\text{Be}$	${}^{16}\text{O}(n, x){}^{11}\text{Be}$	${}^{16}\text{O}(n, x){}^{10}\text{C}$
${}^{16}\text{O}(n, x){}^{11}\text{C}$	${}^{16}\text{O}(n, x){}^{14}\text{C}$	${}^{16}\text{O}(n, x){}^{15}\text{C}$	${}^{16}\text{O}(n, x){}^{13}\text{N}$	${}^{16}\text{O}(n, x){}^{16}\text{N}$	${}^{16}\text{O}(n, x){}^{14}\text{O}$
${}^{16}\text{O}(n, x){}^{15}\text{O}$	${}^4\text{He}(p, x){}^3\text{H}$	${}^{14}\text{N}(p, x){}^7\text{Be}$	${}^{14}\text{N}(p, x){}^{11}\text{Be}$	${}^{14}\text{N}(p, x){}^{10}\text{C}$	${}^{14}\text{N}(p, x){}^{11}\text{C}$
${}^{14}\text{N}(p, x){}^{13}\text{N}$	${}^{14}\text{N}(p, x){}^{14}\text{O}$	${}^{16}\text{O}(p, x){}^3\text{H}$	${}^{16}\text{O}(p, x){}^7\text{Be}$	${}^{16}\text{O}(p, x){}^{11}\text{Be}$	${}^{16}\text{O}(p, x){}^{10}\text{C}$
${}^{16}\text{O}(p, x){}^{11}\text{C}$	${}^{16}\text{O}(p, x){}^{14}\text{C}$	${}^{16}\text{O}(p, x){}^{13}\text{N}$	${}^{16}\text{O}(p, x){}^{14}\text{O}$	${}^{16}\text{O}(p, x){}^{15}\text{O}$	

表 83: [t-yield] パラメータ (2)

name	値	説明
axis =	reg, x, y, z, r, xy, yz, xz, rz mass charge chart  dchain	出力データの x 軸 2次元表示 質量分布を表示。nucleus を指定した場合 isotope 分布 荷電分布を表示。nucleus の指定不可 nucleus chart 形式 (x 軸:N, y 軸:Z)。 nucleus の指定不可 dchain-sp 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
2d-type =	1,2,3,4,5,6,7 (省略可)	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

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、 $\gamma$ 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 84: [t-yield] パラメータ (3)

name	値	説明
gshow =	0(省略時) 1, 2, 3, 4	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
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 =	1(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。
width =	0.5(省略時)	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	(省略可)	reg メッシュの時、各領域の体積を定義します。省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に volume 定義文が必要。volume 定義文。書式は 5.1.2 を参照。
reg vol		
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	1 で出力ファイルを ANGLE 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
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
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

## 6.7 [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。低エネルギーの中性子、光子、電子の反応からの粒子は、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 85: [t-product] 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-product] 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
mother = (次行)	(省略可) all, 数  208Pb Pb	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。 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 86: [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 <sup>3</sup> /source] 3: [1/MeV/source] 4: [1/cm <sup>3</sup> /MeV/source] 5: [1/Lethargy/source] 6: [1/cm <sup>3</sup> /Lethargy/source]
	11, 12, 13, 14, 15, 16	11: [1/nsec/source] 12: [1/cm <sup>3</sup> /nsec/source] 13: [1/MeV/nsec/source] 14: [1/cm <sup>3</sup> /MeV/nsec/source] 15: [1/Lethargy/nsec/source] 16: [1/cm <sup>3</sup> /Lethargy/nsec/source]
	21, 22, 23, 24, 25, 26	21: [1/SR/source] 22: [1/cm <sup>3</sup> /SR/source] 23: [1/MeV/SR/source] 24: [1/cm <sup>3</sup> /MeV/SR/source] 25: [1/Lethargy/SR/source] 26: [1/cm <sup>3</sup> /Lethargy/SR/source]
	31, 32, 33, 34, 35, 36	31: [1/nsec/SR/source] 32: [1/cm <sup>3</sup> /nsec/SR/source] 33: [1/MeV/nsec/SR/source] 34: [1/cm <sup>3</sup> /MeV/nsec/SR/source] 35: [1/Lethargy/nsec/SR/source] 36: [1/cm <sup>3</sup> /Lethargy/nsec/SR/source]

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  $E_{\text{high}}$ ,  $E_{\text{low}}$  のときに、各 Lethargy の幅を  $\ln(E_{\text{high}}/E_{\text{low}})$  で与えます。

表 87: [t-product] パラメータ (3)

name	値	説明
axis =	eng, reg, x, y, z, r, xy, yz, xz, rz t, cos, the	出力データの $x$ 軸 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 =	source nuclear (default) nonela elastic decay fission	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
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
2d-type =	1,2,3,4,5,6,7 (省略可)	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
gshow =	0(省略時), 1, 2, 3, 4	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
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。

表 88: [t-product] パラメータ (4)

name	値	説明
<b>resol =</b>	1(省略時)	<b>gshow, rshow</b> オプションの時、領域境界を求める分解能を各辺 <b>resol</b> 倍します。
<b>width =</b>	0.5(省略時)	<b>gshow, rshow</b> 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。
<b>volume</b>	(省略可)	<b>reg</b> メッシュの時、各領域の体積を定義します。省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に <b>volume</b> 定義文が必要。 <b>volume</b> 定義文。書式は 5.1.2 を参照。
<b>reg vol</b>		
<b>iechrl =</b>	72(省略時)	<b>volume</b> のインプットエコーの時の最大コラム数
<b>volmat =</b>	(省略可、D=9)	<b>xyz</b> メッシュで <b>material</b> が指定されている時、各メッシュの <b>volume correction</b> を行う。(0 で無し) <b>volmat</b> の値は、 <b>xyz</b> メッシュ1辺のスキャン数
<b>epsout =</b>	0(省略時), 1	1 で出力ファイルを <i>ANGeL</i> で処理した <b>eps</b> ファイルを作成。ファイル名は出力ファイルの拡張子を <b>eps</b> に変えたファイル名
<b>ctmin(i) =</b>	(省略可、D=-9999)	<b>i-th</b> カウンターの最小値
<b>ctmax(i) =</b>	(省略可、D= 9999)	<b>i-th</b> カウンターの最大値
<b>trcl =</b>	(省略可)	<b>r-z, xyz</b> メッシュの座標変換番号もしくは座標変換定義
<b>dump =</b>	データ数	ファイルに粒子の情報をダンプします。 <b>mesh=reg</b> 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。
<b>gslat =</b>	1(省略時), 0	0 で <b>gshow</b> で <b>lattice</b> の境界を非表示、1 で表示

[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。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。そのためには、**e-type, a-type, t-type** のメッシュをひとつにすると便利です。(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。)

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。



## 6.8 [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。低エネルギーの中性子による DPA の寄与もライブラリーを使うことによってタリーし、同時に出力することができます。その場合、ライブラリーの指定が必要です。また、`e-mode=1` を用いると、ライブラリーがなくても、核データを用いた中性子の輸送からの寄与が計算できます。

表 89: [t-dpa] 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), 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 =  (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
mother =  (次行)	(省略可) all, 数  208Pb Pb  (ライブラリーでは無効)	スコア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。 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-dpa] セクションを定義する。
unit =	1, 2	1: [DPA/source*1.e+24] 2: [DPA/source]
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
2d-type =	1,2,3,4,5,6,7 (省略可)	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 90: [t-dpa] パラメータ (2)

name	値	説明
output =	dpa simple  all	total DPA の出力 total: 全体の DPA cutoff1: 生成荷電粒子のエネルギーが、計算打切エネルギー (emin) 以下の場合の DPA cutoff2: 輸送された荷電粒子のエネルギーが、計算打切エネルギー (emin) 以下になった場合の DPA transpt: 輸送による全ての荷電粒子の DPA library: neutron library からの DPA の出力 simple に加え、PKA として $d$ , $t$ , ${}^3\text{He}$ , $\alpha$ , Nucleus の寄与の出力。
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	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
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 =	1(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。
width =	0.5(省略時)	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、各領域の体積を定義します。 省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に volume 定義文が必要。 volume 定義文。書式は 5.1.2 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	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
library =	物質数 (省略可)	DPA ライブラリーを物質毎に指定します。 DPA ライブラリーサブセクションの書式は下に示します
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
gslat =	1(省略時), 0	0 で gshow で lattice の境界を非表示、1 で表示

ライブラリーの指定書式

```

library = number of material
part = proton
emax = 3000
mat   fac   lib   mt
  1   1.0   41   445
  2   1.0   42   445
  3   1.0   43   445
....
....
library = number of material
part = neutron
emax = 3000
mat   fac   lib   mt
  1   1.0   41   444
  2   1.0   42   444
  3   1.0   43   444
....
....

```

**part** = でライブラリーを用いる粒子を指定します。現在、**proton** と **neutron** が利用可能です。**emax** = でライブラリーを用いるエネルギーの上限を定義します。粒子を省略すれば、中性子が仮定されます。エネルギー上限を省略すれば、**emax = dmax(1) or dmax(2)** のようにライブラリー計算のエネルギー上限値が仮定されます。**mat** がライブラリーを用いる物質番号、**fac** は規格化係数、**lib** はライブラリーの入った物質番号、**mt** がライブラリー内の DPA データのレコード番号。原田氏作成のライブラリーでは、陽子が **mt=445**、中性子が **mt=444** です。**fac**, **mt** は省略可能です。**mt** を省略した場合は、444 がデフォルトとして入ります。**lib** で指定した物質番号が **[material]** セクションで定義されていなければなりません。ライブラリーの読み込みに際しては、**y** タイプのデータを想定しています。例えば、

```

m41      4009.12y 1
m42      13027.12y 1
m43      26054.12y 3.3066d-04 26056.12y 5.2290d-02
          26057.12y 1.2542d-03 26058.12y 1.5963d-04

```

**mat fac lib mt** の順番は、**mat lib mt fac** 等と変更できます。また、読み飛ばしコラム用の **non** も使えます。

## 6.9 [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ロスだけをタリーします。LET の単位は、keV/ $\mu\text{m}$  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 ( $e_{\text{min}}$ ) 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 91: [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 =  (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
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 $1\text{g}/\text{cm}^3$ 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 $1\text{g}/\text{cm}^3$ 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/ $\mu\text{m}$ ] 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 92: [t-let] パラメータ (2)

name	値	説明
unit =	1, 2, 3, 4, 5, 6  7, 8, 9, 10, 11, 12	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: Track [1/cm <sup>2</sup> /(keV/μm)/source] 8: Dose [MeV/cm <sup>3</sup> /(keV/μm)/source] 9: Track [1/cm <sup>2</sup> /ln(keV/μm)/source] 10:Dose [MeV/cm <sup>3</sup> /ln(keV/μm)/source] 11:Track [1/cm <sup>2</sup> /source] 12:Dose [MeV/cm <sup>3</sup> /source]
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。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
2d-type =	1,2,3,4,5,6,7 (省略可)	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
gshow =	0(省略時), 1, 2, 3, 4	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
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 =	1(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。
width =	0.5(省略時)	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。

表 93: [t-let] パラメータ (3)

name	値	説明
volume reg vol	(省略可)	reg メッシュの時、各領域の体積を定義します。省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に volume 定義文が必要。volume 定義文。書式は 5.1.2 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	1 で出力ファイルを ANGE $L$ 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
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
gslat =	1(省略時), 0	0 で gshow で lattice の境界を非表示、1 で表示

## 6.10 [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、 $\delta$  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<sup>13 14</sup> を参照してください。

[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ロスだけをタリーします。したがって、[T-SED] を使用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  $\varepsilon$  (MeV), lineal energy  $y$  (keV/ $\mu\text{m}$ ), 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 94: [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
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 $1\text{g}/\text{cm}^3$ となるように定義する必要がある。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 $1\text{g}/\text{cm}^3$ 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。
se-unit =	1,2,3	微小領域の付与エネルギー単位 1: Deposit energy $\varepsilon$ [MeV] 2: Lineal energy $y$ [keV/ $\mu\text{m}$ ] 3: Specific energy $z$ [Gy]
cdiam =	(省略可、D=1,0)	微小領域 (球) の直径。単位は $\mu\text{m}$

<sup>13</sup> 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).

<sup>14</sup> 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 95: [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。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 1 $\mu$ m 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	1: Track [cm/(keV/ $\mu$ m)/source] 2: Dose [MeV/(keV/ $\mu$ m)/source] $y * f(y)$ に相当 3: Track [cm/ln(keV/ $\mu$ m)/source] 4: Dose [MeV/ln(keV/ $\mu$ m)/source] $y * d(y)$ に相当 5: Track [cm/source] 6: Dose [MeV/source] (se-unit=2 の場合。se-unit=1,3 の場合は、(keV/ $\mu$ m) 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
2d-type =	1,2,3,4,5,6,7 (省略可)	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 96: [t-sed] パラメータ (3)

name	値	説明
<b>gshow =</b>	0(省略時), 1, 2, 3, 4	<b>mesh=xyz, axis=xy,yz,xz</b> の時、領域境界 (1)、物質番号 (2)、領域番号 (3)、LAT 番号 (4) を表示
<b>rshow =</b>	0(省略時), 1, 2, 3	<b>mesh=reg, axis=xy,yz,xz</b> の時、領域境界 (1)、物質番号 (2)、領域番号 (3) を表示。 この下に <b>xyz</b> 形状メッシュセクションが必要
<b>ginfo =</b>	0(省略時), 1 2	<b>gshow, rshow</b> 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 ( <b>.err</b> ) を出力。
<b>resol =</b>	1(省略時)	<b>gshow, rshow</b> オプションの時、領域境界を求める分解能を各辺 <b>resol</b> 倍します。
<b>width =</b>	0.5(省略時)	<b>gshow, rshow</b> 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。
<b>volmat =</b>	(省略可、D=9)	<b>xyz</b> メッシュで <b>material</b> が指定されている時、各メッシュの <b>volume correction</b> を行う。(0 で無し) <b>volmat</b> の値は、 <b>xyz</b> メッシュ1 辺のスキャン数
<b>epsout =</b>	0(省略時), 1	1 で出力ファイルを <b>AnGEL</b> で処理した <b>eps</b> ファイルを作成。ファイル名は出力ファイルの拡張子を <b>eps</b> に変えたファイル名
<b>ctmin(i) =</b>	(省略可、D=-9999)	<b>i-th</b> カウンターの最小値
<b>ctmax(i) =</b>	(省略可、D= 9999)	<b>i-th</b> カウンターの最大値
<b>trcl =</b>	(省略可)	<b>r-z, xyz</b> メッシュの座標変換番号もしくは座標変換定義
<b>gslat =</b>	1(省略時), 0	0 で <b>gshow</b> で <b>lattice</b> の境界を非表示、1 で表示

## 6.11 [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、輸送粒子でない粒子の発生時のスペクトルを取るタリーは、この時間タリー以外にはありません。輸送粒子でない粒子は、発生時に cut off 粒子と見なされてしまい、他のタリーではスコアできませんが、このタリーを用いることにより可能となります。

表 97: [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 <sup>3</sup> /source] 4: [1/nsec/cm <sup>3</sup> /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 =	(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
angel =	(省略可)	angel パラメータ

表 98: [t-time] パラメータ (2)

name	値	説明
2d-type =	1,2,3,4,5,6,7 (省略可)	2次元表示のオプション
gshow =	0(省略時), 1, 2, 3, 4	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
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 =	1(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。
width =	0.5(省略時)	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、各領域の体積を定義します。 省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に volume 定義文が必要。 volume 定義文。書式は 5.1.2 を参照。
iechr1 =	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	1 で出力ファイルを ANGLE 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
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。
gslat =	1(省略時), 0	0 で gshow で lattice の境界を非表示、1 で表示

[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ありません。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。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。そのためには、e-type, a-type, t-type のメッシュをひとつにすると便利です。(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。)

[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。

## 6.12 [T-Star] セクション

star density を出力します。ライブラリーによる反応は、中性子だけがタリーされます。

表 99: [t-star] 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), 反応入射粒子名	ひとつの [t-star] 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
mother =  (次行)	(省略可) all, 数  208Pb Pb	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。 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-star] 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 <sup>3</sup> /source] 2: [1/cm <sup>3</sup> /MeV/source] 11: [1/cm <sup>3</sup> /nsec/source] 12: [1/cm <sup>3</sup> /nsec/MeV/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。

表 100: [t-star] パラメータ (2)

name	値	説明
output =	all decay elastic nuclear fission absorption heavyion	全ての反応の star density decay 反応の star density elastic 反応の star density non-elastic + Hydrogen + HI 反応の star density fission 反応の star density absorption 反応の star density Heavy Ion 反応の star density
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
2d-type =	1,2,3,4,5,6,7 (省略可)	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
gshow =	0(省略時), 1, 2, 3, 4	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
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 =	1(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。
width =	0.5(省略時)	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、各領域の体積を定義します。 省略した場合、インプットエコーにデフォルト値が表示されます。この行の下に volume 定義文が必要。 volume 定義文。書式は 5.1.2 を参照。
iechr1 =	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	1 で出力ファイルを ANGLE 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

表 101: [t-star] パラメータ (3)

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
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

### 6.13 [T-Dchain]セクション

このセクションでは、DCHAIN-SP 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-SP の接続計算の概要を図 27 に示します。

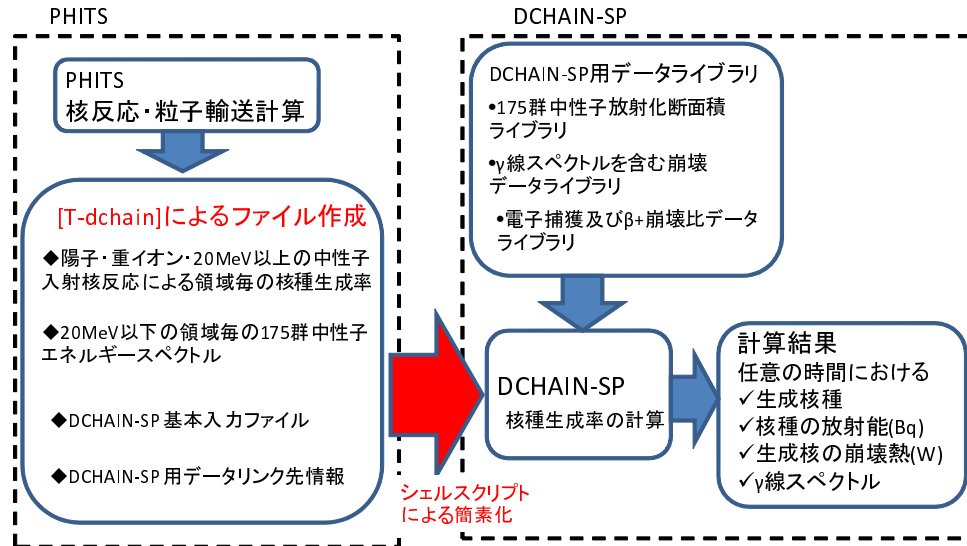


図 27: *PHITs* と DCHAIN-SP の接続計算の概要。

[t-dchain] は、1968 群の中性子スペクトル、核種生成率、DCHAIN-SP 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-SP 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-SP 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、20MeV 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。DCHAIN-SP の詳細は、パッケージに含まれるマニュアル<sup>15</sup> をご参照ください。また、DCHAIN-SP の実行方法や例題については、例題 `dchain.inp` をご覧ください。

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-SP 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-SP 計算に含まれないのでご注意ください。

<sup>15</sup> 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”, JAEA-Data/Code-2001-016 (2001).



表 102: [t-dchain] パラメータ (1)

name	値	説明
mesh =	reg	形状メッシュ (reg 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-SP 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。
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 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。 ※ List 6.1 [t-dchain] タリーの入力例を参照
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。※ List 6.1 [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-SP 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)。

表 103: [t-dchain] パラメータ (2)

name	値	説明
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-SP 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。(List 6.2 をご参照ください。)

これら以外にも、DCHAIN-SP 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, ifisyf, iyild, iggrp, ibetap, acmin, istabl, igsdef, iptb1, iptb2, rprt2, iptb3, igsorg, ebeam, prodpn

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は、DCHAIN-SP のマニュアルをご参照下さい。

#### List 6.1 ● [t-dchain] タリーの入力例

```

1:      mesh = reg           <-region メッシュの指定
2:      reg = 100           <-計算領域の指定
3:      file = testDC.spd   <-DCHAIN-SP 入力ファイル名の指定
4:      title = [t-dchain] test calc.
5:      amp = 1.0E12        <-線源の基準強度 (source/秒)
6:
7:      timeevo = 4         <-照射・冷却時間ステップの数
8:          3.0 h 1.0       <-3 時間運転
9:          2.0 h 0.0       <-2 時間運転停止
10:         3.5 h 1.0       <-3.5 時間運転
11:         15.5 h 0.0      <-15.5 時間運転停止
12:
13:      outtime = 3         <-計算結果出力時間の数
14:          3.0 h          <-運転開始から 3.0 時間後
15:          -1.0 h         <-最後の照射終了時間から 1 時間後
16:          -3.0 h         <-最後の照射終了時間から 3 時間後

```

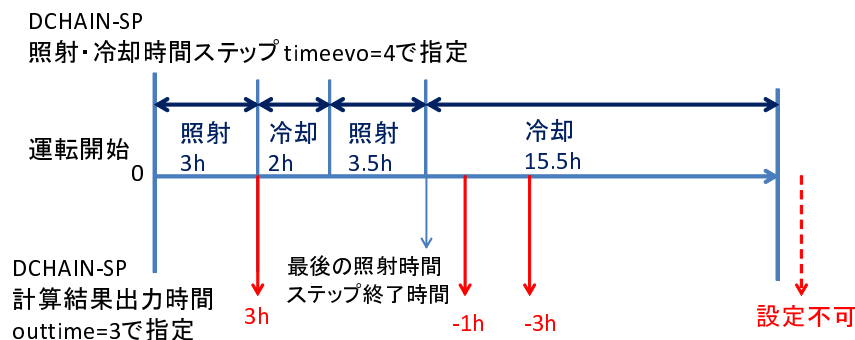


図 28: List 6.1 の例題における照射・冷却時間ステップと計算結果出力時間の関係。

List 6.2 ● [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-SP 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、この入力が必要となります。

このセクションが作成するファイルで、DCHAIN-SP 実行に必要なファイルを以下に示します。

- DCHAIN-SP 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 : 対象領域の数に応じた “`n.flux`” (20MeV 以下の 175 群中性子スペクトル)。ファイル名に、`mesh = reg` による対象領域の指定順に 01 から始まる枝番号が振られます。例 : `n.flux_01, n.flux_02, …, n.flux_XX`
- 核種生成率ファイル : “`nmtc_yield`”
- DCHAIN-SP 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`”

また、DCHAIN-SP 実行時には必要ないのですが、*PHITS* の再開計算等に必要なファイルが 4 つできます。これらは拡張子により区別され、`***.dtrk`, `***.dyld`, `***.err.dyld`, `***.dout` となります。ここで、`***`は `file=`で指定した、拡張子を除く名前を示します。

## 6.14 [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。各ユーザーがファイル `usr tally.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 usr tally` 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 usr tally` 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可能性があるため、十分に注意してください。

`[t-userdefined]` セクションで定義できる変数は、書き出し用ファイル名を与える `file` と `subroutine usr tally` で使用できるパラメータ `udtpara $i$`  ( $i = 0 - 9$ ) の 2 種類です。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 104: `[t-userdefined]` パラメータ

name	値	説明
<code>file =</code>	file name	<code>subroutine usr tally</code> 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)。
<code>udtpara<math>i</math> =</code> ( $i = 0 - 9$ )	(省略可)	<code>subroutine usr tally</code> 内で使用できるパラメータ。 <code>udtpara0</code> から <code>udtpara9</code> まで最大 10 定義できる。

List 6.3 ● [T-Userdefined] パラメータの入力例。

```

1:      file = output1.dat      <-装置番号は 151 で、subroutine usr tally 内では iudtf(1)
2:      file = output2.dat      <-装置番号は 152 で、subroutine usr tally 内では iudtf(2)
3:      udtpara0 = 10.0          <-subroutine usr tally 内では udtpara(0)
4:      udtpara3 = 20.0          <-subroutine usr tally 内では udtpara(3)

```

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

## (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) udtpara(0:9):

**udtpara $i$**  ( $i = 0 - 9$ ) で定義した値。

## (7) 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

## (8) 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 : wight of the particle at (x,y,z)

## (9) QS:

電子の阻止能 ( $dE/dx$ ).

## (10) 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。

## (11) 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

## (12) NAME, NCNT(1), NCNT(2), NCNT(3):

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

## (13) WT, U, V, W:

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

## (14) 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)

(15) 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)

(16) SPX, SPY, SPZ:

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

(17) NZST:

粒子がもつ電荷 (charge state)。

(18) NCLSTS:

意味は、衝突もしくは崩壊による生成粒子の個数です。原子核も含まれます。

(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

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	: px (GeV/c)
= 2	: py (GeV/c)
= 3	: pz (GeV/c)
= 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)



## 6.15 [T-G show] 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、パラメーターセクションで `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。

表 105: [t-gshow] パラメータ

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, 4 5, 6 7, 8	1: 境界線、2: 境界線+物質色 3: 境界線+物質番号、4: 境界線+物質色+物質番号 5: 境界線+領域番号、6: 境界線+物質色+領域番号 7: 境界線+LAT 番号、8: 境界線+物質色+LAT 番号
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
x-txt =	(省略可)	x-軸テキスト
y-txt =	(省略可)	y-軸テキスト
epsout =	0(省略時), 1	1 で出力ファイルを ANGLE 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
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	0 で gshow で lattice の境界を非表示、1 で表示
ginfo =	2(省略時), 0 1 2	領域エラーチェックのオプション 0 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。

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、4.5.4 節で示した図 18 は、次の様な [t-gshow] のインプットで出力しています。

List 6.4 ● [t-gshow] の例題

```
1: [ T - gshow ]
2:   mesh = xyz
3:   x-type = 2
4:   nx = 100
5:   xmin = -10.
6:   xmax = 10
7:   y-type = 1
8:   ny = 1
9:   -5.0 5.0
10:  z-type = 2
11:  nz = 100
12:  zmin = -10.
13:  zmax = 10.
14:  axis = xz
15:  output = 8
16:  file = cell-example6.dat
17:  epsout = 1
```

## 6.16 [T-R show] 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、色の変化を線形にするか、対数にするかは、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 106: [t-rshow] パラメータ

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 =	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
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
reg = value		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。
reg val		value 定義文。書式は volume 定義文と同じ。5.1.2 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
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
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	0 で gshow で lattice の境界を非表示、1 で表示
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。

例えば、4.5.4 節の図 18 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 29 の結果を得ることができます。

List 6.5 ● [t-rshow] の例題

```

1: [ T - rshow ]
2:   mesh = xyz
3:   x-type = 2
4:   nx = 100
5:   xmin = -10.
6:   xmax = 10.
7:   y-type = 1
8:   ny = 1
9:   -5.0 5.0
10:  z-type = 2
11:  nz = 100
12:  zmin = -10.
13:  zmax = 10.
14:  axis = xz
15:  output = 1
16:  file = cell-example6-rshow.dat
17:  epsout = 1
18:  reg = (201<101[-1 1 0]<1) (201<101[0 1 0]<1) (201<101[1 1 0]<1)
19:        (201<101[-1 0 0]<1) (201<101[0 0 0]<1) (201<101[1 0 0]<1)
20:        (201<101[-1 -1 0]<1) (201<101[0 -1 0]<1) (201<101[1 -1 0]<1)
21:  value
22:  non  reg    val    # reg definition
23:  1  1000001  1.0000E+00 # ( 201 < 101[ -1 1 0 ] < 1 )
24:  2  1000002  2.0000E+00 # ( 201 < 101[  0 1 0 ] < 1 )
25:  3  1000003  3.0000E+00 # ( 201 < 101[  1 1 0 ] < 1 )
26:  4  1000004  4.0000E+00 # ( 201 < 101[ -1 0 0 ] < 1 )
27:  5  1000005  5.0000E+00 # ( 201 < 101[  0 0 0 ] < 1 )
28:  6  1000006  6.0000E+00 # ( 201 < 101[  1 0 0 ] < 1 )
29:  7  1000007  7.0000E+00 # ( 201 < 101[ -1 -1 0 ] < 1 )
30:  8  1000008  8.0000E+00 # ( 201 < 101[  0 -1 0 ] < 1 )
31:  9  1000009  9.0000E+00 # ( 201 < 101[  1 -1 0 ] < 1 )

```

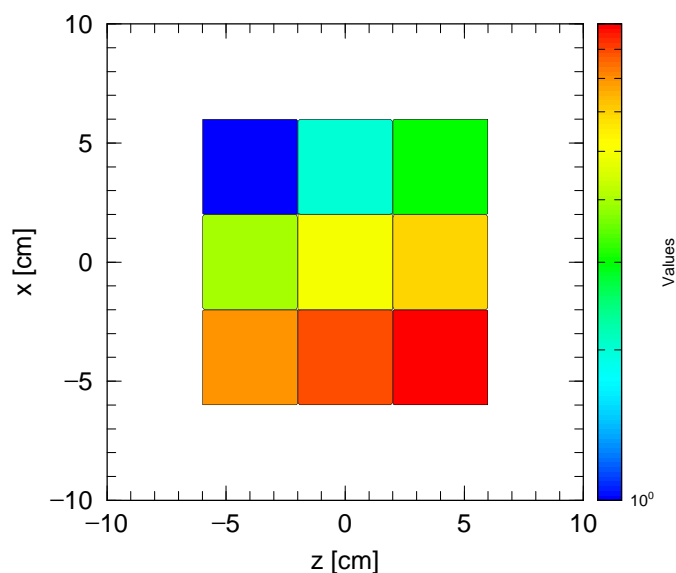


図 29: [t-rshow] の例題を用いて図 18 の各領域に色をつけた結果。

## 6.17 [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 107: [t-3dshow] パラメータ (1)

name	値	説明
<code>output =</code>	<code>0, 1, 2, 3</code> (省略時 3)	<b>0</b> : ドラフト、 <b>1</b> : 境界線のみ、 <b>2</b> : 境界線無し、 <b>3</b> : 境界線+カラー
<code>material =</code>  (次行)	(省略可) <code>all, 数</code>  <code>2 5 8</code>	表示する <b>material</b> を限定する。複数定義可。 <b>all</b> : 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 <b>material</b> を次の行に記述する。負の数にした場合は、それらの <b>material</b> を表示しないことを意味する。 <b>material</b> 番号
<code>x0 =</code> <code>y0 =</code> <code>z0 =</code>	(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。
<code>e-the =</code> <code>e-phi =</code> <code>e-dst =</code>	(D=80) (D=140) (D=w-dst*10)	視点の <i>z</i> 方向からの角度 $\theta$ (degree) 視点の <i>x</i> 方向からの方位角 $\phi$ (degree) 視点の視点原点からの距離 (cm)
<code>l-the =</code> <code>l-phi =</code> <code>l-dst =</code>	(D=e-the) (D=e-phi) (D=e-dst)	光源の <i>z</i> 方向からの角度 $\theta$ (degree) 光源の <i>x</i> 方向からの方位角 $\phi$ (degree) 光源の光線原点からの距離 (cm)
<code>w-wdt =</code> <code>w-hgt =</code> <code>w-dst =</code>  <code>w-mnw =</code> <code>w-mnh =</code> <code>w-ang =</code>	(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)
<code>heaven =</code>	(D=y)	画面上方向の向き <b>x</b> , <b>-x</b> , <b>y</b> , <b>-y</b> , <b>z</b> , <b>-z</b> で指定
<code>mirror =</code>	(D=0)	=-1 で画面の左右鏡面変換

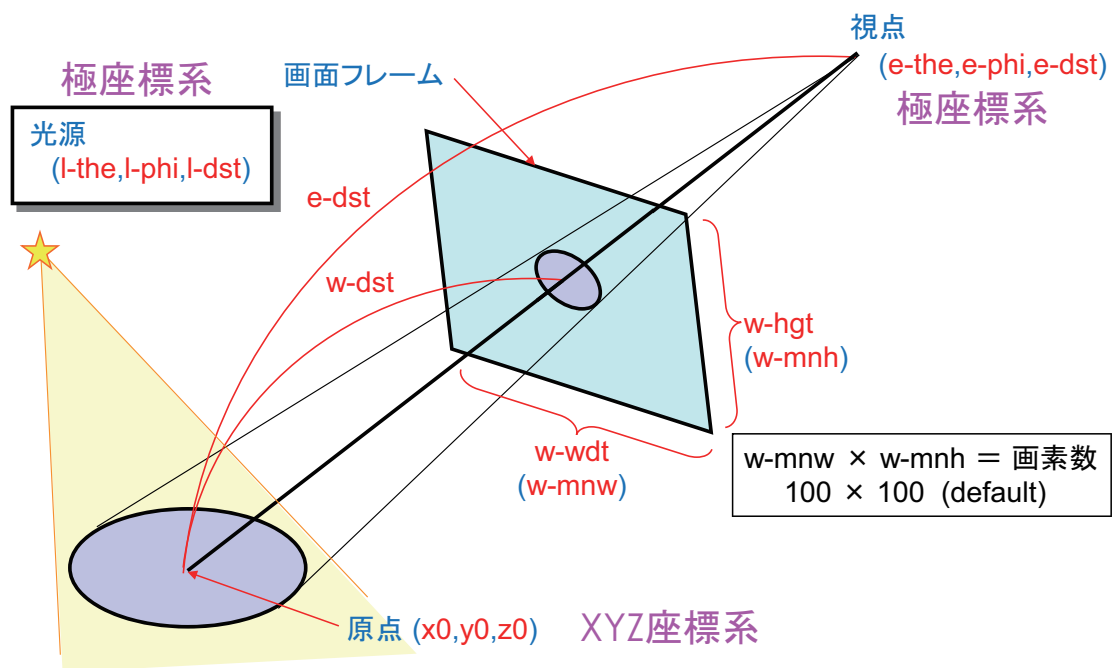


図 30: 3dshow タリー: 原点  $(x_0, y_0, z_0)$ 、視点  $(e-the, e-phi, e-dst)$ 、光源  $(l-the, l-phi, l-dst)$  及び画面フレーム  $(w-wdt, w-hgt, w-dst)$  の与え方。

表 108: [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 =	(D=0.8)	面の明るさの制限、1 で最大、0 で明るさ無し
dark =	(D=0.2)	面の暗さの制限、1 で暗さ無し、0 で最大
box =	(D=0)	透過 box の数、最大 5 個まで。
box	10 個の数字	透過 box の数だけ繰り返す
matinbox =	(省略可) all, 数	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。
(次行)	2 5 8	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 =	1(省略時)	境界を求める分解能を各辺 resol 倍します。
width =	0.5(省略時)	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
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
epsout =	0(省略時), 1	1 で出力ファイルを ANGE <sub>L</sub> 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

reg = また、reginbox = の表式は、サブセクション 5.1.1 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。

### 6.17.1 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。

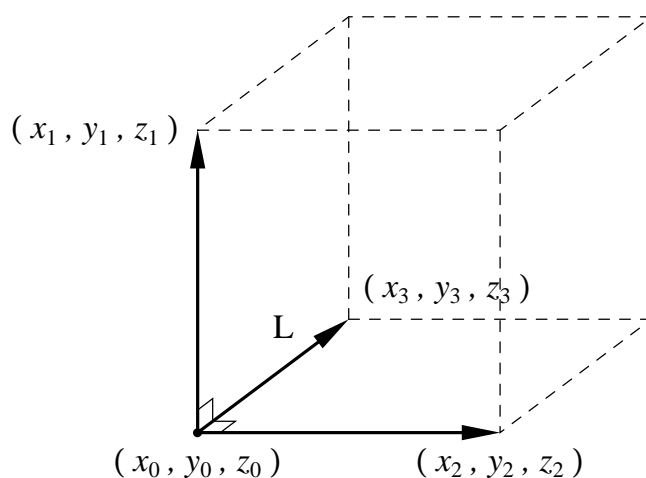
ただし、透過 box で定義した空間に真空(void)があると適切に機能しない場合があります。そのときは、密度を薄くした空気等の物質でその領域を満たしてください。

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

```





## 6.17.2 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。

## List 6.6 ● [t-3dshow] の例題

```

1: [cell]
2:   1  0 -1 fill=1
3:   2  0 -41 42 -43 44 -45 46  u=1 fill=5
4:  22  0 -41 42 -43 44 -45 46  u=1 trcl=(0 0 20) fill=6
5:  23 like 22 but trcl=(0 0 40) fill = 7
6:   5  0 -21 22 -23 24 -25 26  u=5 lat=1 fill=3
7:   6  0 -21 22 -23 24 -25 26  u=6 lat=1 fill= -1:1 0:0 0:0  2 2(0 0 5) 2
8:   7  0 -21 22 -23 24 -25 26  u=7  fill= -1:1 0:0 0:0  2 3 2 lat=1
9:   3  1 3.97300E-02  3 u=2
10:  4  6 4.18280E-02 -3 u=2
11:  13  5 8.47130E-04 -3 u=3
12:  14  3 1.23620E-01  3 u=3
13:   8 -1 +1
14: [surface]
15:   1  rpp -15 15 -5 5 -5 55
16:  21  px   5
17:  22  px  -5
18:  23  py   5
19:  24  py  -5
20:  25  pz  15
21:  26  pz  -5
22:  41  px  15
23:  42  px -15
24:  43  py   5
25:  44  py  -5
26:  45  pz  15
27:  46  pz  -5
28:   5  rpp -20 20 -5 5 -5 35
29:   6  rpp -20 20 -5 5 -5 15
30:   7  rpp -20 20 -5 5 35 55
31:   3  c/y  0 10 4

```

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。

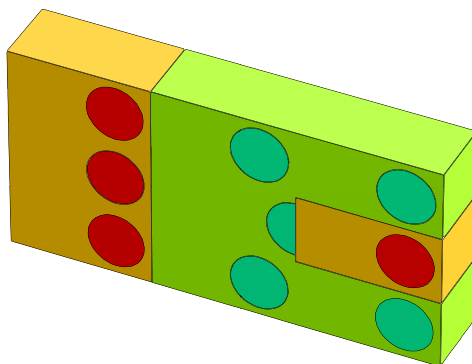
## List 6.7 ● [t-3dshow] の例題

```

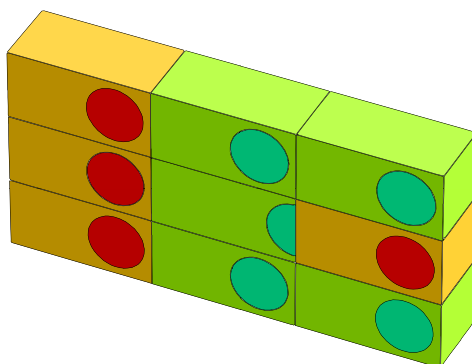
1: [t-3dshow]
2:   output = 3
3:   heaven = x
4:   resol = 2
5:   width = 0.1
6:   x0 = 0
7:   y0 = 0
8:   z0 = 25
9:   e-the = 70
10:  e-phi = 50
11:  e-dst = 1000
12:  l-the = 50
13:  l-phi = 25
14:  l-dst = 2000
15:  w-wdt = 60
16:  w-hgt = 40
17:  w-dst = 150
18:  file = dshow.dat

```

この結果は、



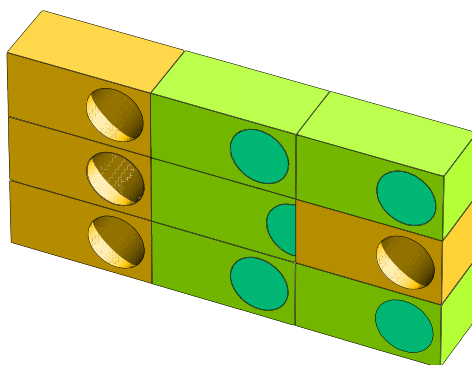
となります。これに、`line = 1`を加えて、領域境界の線も書き込んでみます。



`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
```

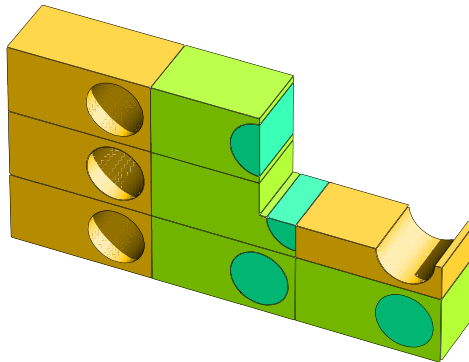
を加えます。



次に、box を設定します。

```
box = 1
box 0 10 30
    100 10 30
    0 10 100 100
```

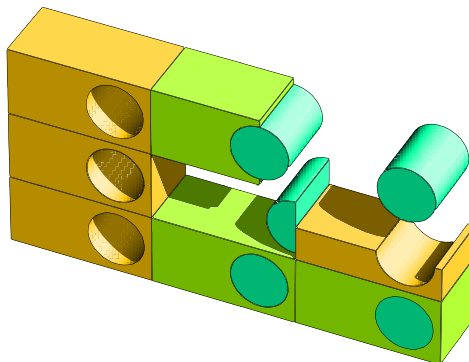
box により、一部が透明になり内部が見られます。



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。

## 7 タリーを用いた体積、面積計算

reg メッシュを用いたタリーにおいて、各領域の体積やある横断面の面積の値が必要になります。本説ではそれらの値をモンテカルロ積分により求める方法についてご紹介します。

モンテカルロ計算で体積や面積を求めるには、空間的に均一な粒子軌跡を用意して、領域を通過する、もしくは横断面を通過する flux を計算することが必要です。そのためには、**s-type=9** の球殻線源において **r1=r2** とし、**dir=-all** とします。その際、目的の領域や横断面が半径 **r1** の球に入るようにしてください。**dir=-all** は球殻から内向きに線源が発生する設定で、 $\cos^2\theta$  のバイアスを掛けてあるため、より中心部分で統計が良くなるようになっていきます。また、**totfact** として  $\pi r_1^2$  の値を入れてください。線源の発生する面積を入れることで適切に規格化が行われます。

体積を求める場合は **[t-track]** タリーを用います。**mesh=reg** とし、**unit=4** としてください。対して、面積を求める場合は **[t-cross]** タリーを使います。**mesh=reg**、**unit=1** とし、**output=flux** としてください。

上記の線源とタリーを設定した上で、**icntl=5** とし輸送計算を実行します。詳細は、`/phits/lecture/lec03` フォルダにある基礎実習3の講習会資料をご参照ください。

以下に **[source]** セクションや **[t-track]**, **[t-cross]** タリーの例題を示します。

### List 7.1 ● 体積、面積計算線源例

```
1: [ S o u r c e ]
2:   s-type = 9
3:   proj = proton
4:   e0 = 500.0
5:   x0 = 0.0
6:   y0 = 0.0
7:   z0 = 30.0
8:   r1 = 18
9:   r2 = 18
10:  dir = -all
11:  totfact = 18*2*pi
```

この例題では、中心が (0,0,30)、半径が 18cm の球面が設定されています。粒子種、エネルギーは任意です。目的の領域や横断面がこの球内に入る様に半径、中心位置を決めます。次に、体積、面積を求めるタリーの例を示します。

### List 7.2 ● 体積を求めるタリー例

```
1: [ T - T r a c k ]
2:   mesh = reg
3:   reg = 1 2 3 4 5
4:   e-type = 2
5:   emin = 0.
6:   emax = 1000.0
7:   ne = 1
8:   axis = reg
9:   unit = 4
10:  file = volume.dat
```

求めたい領域を指定し、エネルギー範囲は線源のエネルギーを含む 1 群とします。**unit=4** とします。**unit=4** より、体積は自動的に 1 にセットされますから、**volume** セクションは特に必要ありません。この計算の結果得られた体積の値を、実際に必要なタリーの体積定義のところにコピーして使います。

また、**multiplier** を使えば、領域の質量 (体積 × 密度) を直接計算することもできます。具体的には、上記 **[t-track]** セクションに

```

multiplier = all
  part = all
  emax = 1.0e10
mat mset1
all ( 1.0 -120 )

```

を加えます。ここで、`multiplier` 番号 `-120` は、計算した飛跡長に密度を乗じるための関数です。詳しくは、[6.1\[t-track\]](#) セクションをご参照ください。

次は面積の例題です。

**List 7.3 ● 面積を求めるタリー例**

```

1: [ T - C r o s s ]
2:   mesh = reg
3:   reg = 3
4:   r-in   r-out   area
5:   ( 1 2 ) ( 1 2 ) 1.0000E+00
6:   ( 2 3 ) ( 2 3 ) 1.0000E+00
7:   ( 3 4 ) ( 3 4 ) 1.0000E+00
8:   e-type = 2
9:   emin = 0.
10:  emax = 1000.0
11:   ne = 1
12:  axis = reg
13:  unit = 1
14:  file = area.dat

```

求めたい面を指定し、エネルギー範囲は線源のエネルギーを含む 1 群とします。`unit=1` とします。この計算の結果得られた面積の値を、上の `area` の値にコピーして使います。

## 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読み込んで、アスキーダンプファイルを書き換え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。

### List 8.1 ● dump-a.f のソース

```

1: *****
2: *
3: *   This program exchanges the binary data and the ascii data   *
4: *   of dump file.                                             *
5: *
6: *   modified by K.Niita on 2005/08/15                          *
7: *
8: *
9: *
10: *
11: *****
12:   implicit real*8 (a-h,o-z)
13: -----
14:   dimension isdmp(0:30)
15:   dimension jsdmp(0:30)
16:   data isdmp / 31*0 /
17:   data jsdmp / 31*0 /
18:   character chin*80
19:   character chot*80
20:   logical exex
21:   character dmpc(30)*4
22:   data dmpc / ' kf', ' x', ' y', ' z', ' u', ' v', ' w',
23: & ' e', ' wt', ' tm', ' c1', ' c2', ' c3',
24: & ' sx', ' sy', ' sz', ' n0', ' nc', ' nb', ' no',
25: & ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ',
26: & ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' /
27:   dimension dmpd(30)
28:   dimension dmpp(30)
29:   data dmpp / 2112., 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0,
30: & 100., 1.0, 0.0, 0.0, 0.0, 0.0,
31: & 0.0, 0.0, 0.0, 0.0, 1.0, 1.0, 1.0,
32: & 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
33: & 0.0, 0.0, 0.0/
34: -----
35:   in = 5
36:   io = 6
37:   id = 20
38:   ia = 21
39:   iserr = 0
40: -----
41: *   user program frag : 0 => no, 1 => with user program
42: -----
43:   iuser = 0
44: -----
45: *   read ascii or binary frag
46: -----
47:   write(io,*) ' ** 0 => read binary to ascii'
48:   write(io,*) ' ** 1 => read ascii to binary'
49:   read(in,*,end=993) iasb
50: -----
51: *   read the name of input dump file
52: -----
53:   write(io,*)

```

```

54:         write(io,*) ' ** put the file name of input dump file'
55:         read(in,'(a80)',end=998) chin
56:         inquire( file = chin, exist = exex )
57:         if( exex .eqv. .false. ) then
58:             write(io,*) ' ** Error : the file does not exist'
59:             goto 999
60:         end if
61:         if( iasb .eq. 0 ) then
62:             open(id, file = chin,
63: &             form='unformatted',status = 'old' )
64:         else
65:             open(id, file = chin,
66: &             form='formatted',status = 'old' )
67:         end if
68: *-----
69: *         read the number of data and data sequence
70: *-----
71:         write(io,*)
72:         write(io,*) ' ** put the number of data in a record'
73:         read(in,*,end=997) isdmp(0)
74:         write(io,*)
75:         write(io,*) ' ** put the ID numbers of data in a record'
76:         read(in,*,end=996) ( isdmp(i), i = 1, isdmp(0) )
77:         do k = 1, isdmp(0)
78:             if( isdmp(k) .gt. 20 .or.
79: &             isdmp(k) .le. 0 ) goto 992
80:             jsdmp(isdmp(k)) = k
81:         end do
82:         write(io,*)
83:         write(io,'(' # dump data : ',,30(a4))')
84: &         ( dmpp(isdmp(j)), j = 1, isdmp(0) )
85: *-----
86: *         read the name of output dump file
87: *-----
88:         write(io,*)
89:         write(io,*) ' ** put the file name of output'
90:         read(in,'(a80)',end=998) chot
91:         inquire( file = chot, exist = exex )
92:         if( exex .eqv. .true. ) then
93:             write(io,*)
94:             write(io,*) ' ** Warning : the file already exists'
95:             write(io,*) ' ** Do you want to overwrite ?'
96:             write(io,*) ' ** Yes <= 0, No <= 1'
97:             read(in,*,end=995) iyes
98:             if( iyes .ne. 0 ) goto 999
99:         end if
100:         if( iasb .eq. 0 .or. iuser .ne. 0 ) then
101:             open(ia, file = chot,
102: &             form='formatted',status = 'unknown' )
103:         else
104:             open(ia, file = chot,
105: &             form='unformatted',status = 'unknown' )
106:         end if
107: *-----
108: *         read the number of records to read
109: *-----
110:         write(io,*)
111:         write(io,*) ' ** put the number of records to read'
112:         write(io,*) ' ** all <= 0, or positive integer'
113:         read(in,*,end=994) irec
114: *-----
115: *         start reading the data
116: *-----
117:         write(io,*)
118:         write(io,*) ' ** start read and write the data'
119: *-----
120:         jrec = 0
121:         100 jrec = jrec + 1
122:         if( irec .gt. 0 .and. jrec .gt. irec ) goto 500
123:         687 continue
124:         if( iasb .eq. 0 ) then
125:             read(id,end=688,err=690)

```

```

126:      &      ( dmpd(isdmp(k)), k = 1, isdmp(0) )
127:      else
128:          read(id,'(30(1p1d24.15))',end=688,err=690)
129:      &      ( dmpd(isdmp(k)), k = 1, isdmp(0) )
130:      end if
131:      goto 689
132: 688      if( irec .gt. 0 ) then
133:          rewind id
134:          goto 687
135:      else
136:          goto 500
137:      end if
138: 690      continue
139:          iserr = iserr + 1
140:          write(io,'('' ** Error in dump file no ='',i5)') iserr
141:          goto 687
142: 689      continue
143: *-----
144: *      user program here
145: *-----
146:      if( iuser .ne. 0 ) then
147:          do k = 1, 20
148:              if( jsdmp(k) .gt. 0 ) dmpp(k) = dmpd(k)
149:          end do
150:              kf = nint( dmpp(1) )
151:              x = dmpp(2)
152:              y = dmpp(3)
153:              z = dmpp(4)
154:              u = dmpp(5)
155:              v = dmpp(6)
156:              w = dmpp(7)
157:              e = dmpp(8)
158:              wt = dmpp(9)
159:              t = dmpp(10)
160:              n1 = nint( dmpp(11) )
161:              n2 = nint( dmpp(12) )
162:              n3 = nint( dmpp(13) )
163:              sx = dmpp(14)
164:              sy = dmpp(15)
165:              sz = dmpp(16)
166:              n0 = nint( dmpp(17) )
167:              nc = nint( dmpp(18) )
168:              nb = nint( dmpp(19) )
169:              no = nint( dmpp(20) )
170:          end if
171: *-----
172: *      write data on the file
173: *-----
174:      if( iuser .eq. 0 ) then
175:          if( iasb .eq. 0 ) then
176:              write(ia,'(30(1p1d24.15))')
177:      &      ( dmpd(isdmp(k)), k = 1, isdmp(0) )
178:          else
179:              write(ia)
180:      &      ( dmpd(isdmp(k)), k = 1, isdmp(0) )
181:          end if
182:      end if
183: *-----
184:      goto 100
185: *-----
186: *      end of process
187: *-----
188: 500      continue
189:          write(io,*) ' ** end of read and write the data'
190:          write(io,'('' ** number of processed records is '' ,
191:      &      i8)') jrec-1
192:          write(io,*)
193:          close( id )
194:          close( ia )
195:          goto 999
196: *-----
197: 992      continue

```



```

198:     write(io,*) ' ** Error : ID should be 1 - 20'
199:     goto 999
200: 993 continue
201:     write(io,*) ' ** Error : the ascii or binary frag is wrong'
202:     goto 999
203: 994 continue
204:     write(io,*) ' ** Error : the number of records is wrong'
205:     goto 999
206: 995 continue
207:     write(io,*) ' ** Error : the answer should be 0 or 1'
208:     goto 999
209: 996 continue
210:     write(io,*) ' ** Error : the ID numbers is wrong'
211:     goto 999
212: 997 continue
213:     write(io,*) ' ** Error : the number of data is wrong'
214:     goto 999
215: 998 continue
216:     write(io,*) ' ** Error : file name is wrong'
217:     goto 999
218: 999 continue
219: stop
220: 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 66, 67 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` でインポートラ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` でインポートランスを含む場合 (`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` でインポートラ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` でインポートランスを含む場合 (`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、粒子のウェイト、`t(i)` は、時間 (ns) です。

`igcut = 3` の場合は、上の `incut = 2` の場合の `t(i)` の代わりに、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ます。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 31 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 31 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。

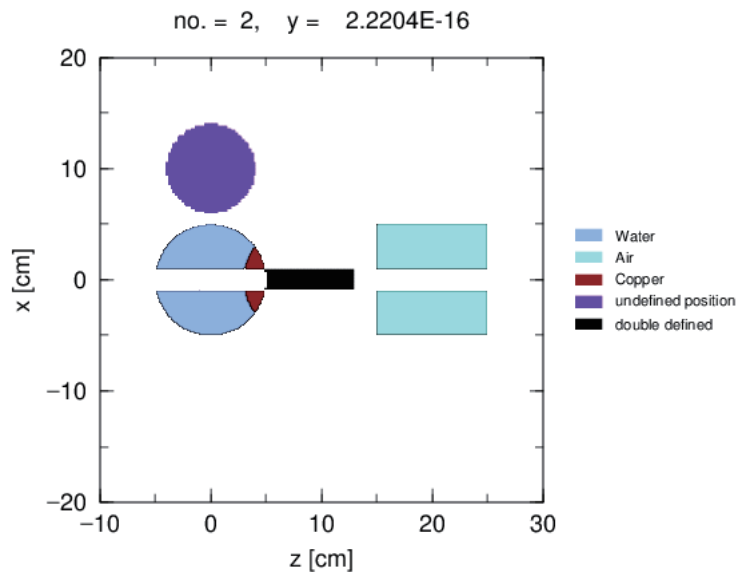


図 31: 領域エラーがある場合の出力結果

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、エラーが起きている領域の(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 coordinates
(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 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。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 2.4 節をご参照ください。

メモリ分散型並列では、各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。

### 11.1 メモリ分散型並列

#### 11.1.1 実行方法

メモリ分散型並列には **MPI** プロトコルを用いるため、その実行コマンドは

```
mpirun -np 8 phits_lin.exe
```

のようになります。ここで **-np** の後の数字は並列の実行 **PE** 数で、**phits\_lin.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下さい。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 (2.4 節参照)。

#### 11.1.2 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)。

### 11.1.3 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。

### 11.1.4 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。

### 11.1.5 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しますので、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。



## 11.2 メモリ共有型並列

### 11.2.1 実行方法

メモリ共有型並列版 *PHITs* の実行方法は、シングル版 *PHITs* の実行方法と同じく

```
phits_lin.exe < phits.inp
```

のようになります (`phits_lin.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、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

```
set OMP_NUM_THREADS=4 (Windows の場合)
export OMP_NUM_THREADS=8 (Mac, Linux の場合)
```

のように行います。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、その3行目でコア数を指定してください。例えば4コアで計算する場合は、

```
set PHITS_PARALLEL=4
```

と設定してください。なお、*PHITs* のメモリ共有型並列は、Hyper Threading 技術には対応しておりませんので、`PHITS_PARALLEL=0` としてデフォルトの `OMP_NUM_THREADS` を使っても計算時間が長くなる場合があります。その場合は、OS が認識しているコア数の半分の値を `PHITS_PARALLEL` に直接入力してください。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。

### 11.2.2 メモリ共有型並列計算の注意点

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倍くらい長くなります。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。

また、現在、下記の計算機能がメモリ共有型並列計算に対応できていません。

(1) *ANGEL* の起動 (タリーで `epsout=1` と指定した場合、Windows のみ)

(1) 番目は Windows 固有の問題で、Intel Fortran Compiler を用いてコンパイルした場合に起こります。そこでバージョン 2.52 では、実行時にバッチファイル `phits.bat` を用いることで、`epsout=1` と指定しても動作するにしました。このバッチファイルを使うと、*PHITs* 計算の最初に内部で `epsout=0` に設定し直し、計算終了後に *ANGEL* を用いて `eps` ファイルを作成します。ただしこの場合、`itall=1` を設定してバッチ終了毎にタリー結果を更新する、という機能が使えませんのでご注意ください。また、ユーザー自身でコンパイルする際は、`a-angel.f` を `a-angel-winopenmp.f` に変更する必要があります (2.5 節参照)。

また、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で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。両方で結果が異なる場合は、バグの可能性があるので PHITS 事務局までご連絡ください。

## 12 FAQ

### 12.1 パラメータ設定関連

Q1.1 低エネルギー中性子の輸送計算精度が良くないのですが。

A1.1 一般に 20MeV 以下の中性子の挙動は複雑で、核内カスケードモデル (INC) や量子分子動力学モデル (QMD) では再現できません。したがって、核データライブラリーを使用した計算が必要となります。中性子に対して核データを利用するには、 $\text{emin}(2)=1.0\text{e-}10$ ,  $\text{dmax}(2)=20.0$  のように、中性子の計算打切エネルギーを下げ、核データを利用する上限エネルギーを大きくしてください。

Q1.2 光子の輸送計算ができません。

A1.2 デフォルトの設定では、光子・電子の輸送計算は行われません。光子・電子の輸送計算が必要な場合は、 $\text{dmax}(12-14)$  と  $\text{emin}(12-14)$  をお手持ちの原子データライブラリーの上限值に設定してください。PHITS 用のライブラリー (jphlib01&jellib01) を使用している場合、光子に対しては 100GeV、電子・陽電子に対しては 10GeV が上限値で、下限値は 1keV です。ただし、バージョン 2.52 より前の PHITS には電子輸送計算に対するバグがあるため、電子・光子の輸送計算を行う場合は、必ず最新版にアップデートしてください。

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 計算する問題に依存しますが、バージョン 2.50 以降では、基本的にそうなるように設定しています。ただし、20MeV 以下の中性子を扱う場合、核データを用いた計算が必須になります (A1.1 参照)。

Q1.4 イベントジェネレーターモードは、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ーモード (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、半導体ソフトウェア発生率の計算など、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「4.2.21 Event Generator Mode」をご参照ください。

Q1.5 統計誤差導出方法 (istdev の設定) はどのような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。

### 12.2 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 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くなりました。PHITS 事務局で推奨しているコンパイラは Intel Fortran Compiler (11.1 以降) と gfortran (4.71 以降) です。なお、makefile に書かれていないオプションや環境でコンパイル&実行に成功した方は、PHITS 事務局までご連絡ください。詳しくは、「2.4Makefile を利用したコンパイル」をご参照ください。

Q2.2 巨大な体系について計算しようとする、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。詳しくは「2.9 配列の大きさ」をご参照ください。また、ボクセルファントムなど巨大な Lattice 構造を扱う場合は、同時に *latmax* パラメータも大きくしてください。なお、Windows および Mac 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、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:*」が使いません。

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。詳しくは、「2.7 実行シェル」をご参照ください。

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。

## 12.3 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 ユーザー定義関数 (*usrdfn*) を指定しない限り、基本的には同じになるはずですが、ただし、[t-deposit] は非荷電粒子 (中性子や光子) のカーマ近似やエネルギーカットオフによる発熱量をカウントしませんので、両者の結果には多少差が生じます。特に、イベントジェネレーターモードを使わずに発熱量を計算する場合に両者の違いが大きくなりますので、必ず [t-heat] を使う必要があります。[t-deposit] は、被ばくによる生物学的効果など同じ発熱量でも影響の大きさの違う量を計算する際、極めて有用となります。

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。ややこしくて申し訳ないのですが、従来からこの方針でプログラミングしておりますので、ご了承ください。

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<sup>3</sup>) 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。

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「4.2.2 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 メモリ分散型並列計算で dump モードは使えるのでしょうか？

A3.5 バージョン 2.30 より使えるようになりました。使用方法については、PHITS 事務局までご連絡ください。

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  $1\text{cm}^3$  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[ $1/\text{cm}^2/\text{source}$ ] と unit=4[ $\text{cm}/\text{source}$ ] は同じ値を与えます。

## 12.4 線源設定関連

Q4.1 等方線源ソース (s-type=9,10, 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  $\pi r_1^2$  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。ただし、PHITS 2.30 以前では、dir=iso にバグがありましたので、この線源を使う方は、必ず新しい PHITS をご利用ください。

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ください。

Q4.3 線源でエネルギー微分フラックスを直接与えることは可能でしょうか？

A4.3 バージョン 2.60 より使えるようになりました。[source] セクションにおいて、e-type=21,24,31,34 を指定してください。(「4.3.15 エネルギー分布の定義」参照)

## 索引

- <source>, 53
- [brems bias], 15, 113
- [cell], 4, 15–17, 54, 83, 86–90, 92, 93, 95–99, 102, 103, 108, 110, 112, 188
- [counter], 15, 120, 144
- [delta ray], 128
- [elastic option], 15, 126
- [electro magnetic field], 15, 54, 119
- [end], 16, 17
- [forced collisions], 15, 115
- [importance], 15, 37, 108, 109
- [magnetic field], 15, 54, 116, 119
- [mat name color], 15, 122
- [mat time change], 15, 124
- [material], 4, 15, 36, 82, 84–86, 90, 92, 93, 95–99, 173, 188
- [multiplier], 6, 129, 130, 150, 151
- [parameters], 15, 22, 83, 100, 119, 146, 156, 165, 189
- [photon weight], 15, 114
- [reg name], 15, 121
- [source], 2, 4, 15, 19, 52, 117, 119
- [super mirror], 15, 125
- [surface], 15–17, 46, 86–90, 92, 93, 95–99, 101, 103, 192
- [t-3dshow], 16, 22, 121, 199
- [t-cross], 3, 6, 16, 43, 131, 142, 152, 206
- [t-dchain], 5, 16, 131, 186
- [t-deposit2], 16, 24, 131, 162, 163
- [t-deposit], 3, 4, 16, 24, 59, 131, 159, 161
- [t-dpa], 16, 131, 171
- [t-gshow], 16, 22, 94, 121, 131, 140, 142, 195–197
- [t-heat], 3, 16, 131, 156
- [t-let], 16, 131, 174
- [t-product], 3, 6, 16, 22, 131, 167
- [t-rshow], 16, 22, 121, 131, 142, 143, 197, 198
- [t-sed], 16, 131, 177
- [t-star], 16, 131, 183
- [t-time], 3, 6, 16, 131, 180
- [t-track], 6, 16, 43, 119, 129–131, 148, 151, 206, 207
- [t-userdefined], 6, 16, 131, 190
- [t-yield], 16, 28, 131, 139, 164
- [temperature], 15, 112
- [timer], 15, 127
- [title], 15, 21
- [transform], 15, 87, 96, 101, 102, 105–107
- [volume], 15, 37, 111, 133, 188, 189
- [weight window], 15
- [wieght window], 110
- 2d-type, 141, 142, 149, 155, 158, 161, 163, 165, 169, 171, 176, 179, 182, 185
- 3dshow, 144
- a-curr, 155
- a-type, 2, 4, 54–59, 62, 64, 73, 74, 135, 136, 153, 155, 167
- absorption, 185
- ag1, 74
- ag2, 74
- all, 130, 148, 151, 153, 156, 159, 162, 164, 167, 171, 173, 174, 177, 180, 183, 185
- amp, 188
- andit, 30
- angel, 149, 155, 158, 161, 163, 165, 169, 171, 176, 179, 180, 185, 195, 197, 198, 201
- ANGEL, 12, 24, 141–143, 197
- angle stragglng, 31
- area, 155, 207
- ascat1, 31
- ascat2, 31
- ATIMA, 31
- axis, 24, 28, 43, 133, 138, 139, 141–143, 149, 153, 155, 156, 159, 161, 163, 165, 169, 171, 173, 176, 179, 180, 182, 183, 185, 195–198, 206, 207
- b-curr, 155
- batch.now, 13
- Bertini, 26, 28, 30
- bias, 113
- bnum, 42
- BOX, 93, 95, 97–99, 103
- bremsstrahlung, 42, 113
- brsh, 218
- C/X, 102
- C/Y, 102

- C/Z, 102
- cdiam, 177
- CGVIEW, 22, 40
- charge, 138, 165
- chart, 28, 138, 165
- cmin(i), 25
- color, 122, 124
- COND, 83
- cos, 138, 153, 167, 169
- CPU 時間, 37
- ctmax, 144, 150, 155, 158, 161, 163, 166, 169, 173, 176, 179, 182, 185
- ctmin, 144, 150, 155, 158, 161, 163, 166, 169, 173, 176, 179, 182, 185
- Cugnon, 28
- current, 79–81, 152, 155
- cutoff, 180
- CX, 102
- CY, 92, 96–98, 102
- CZ, 102
  
- dchain, 1, 28, 40, 138, 165
- DCHAIN-SP, 186
- decay, 169, 185
- decay 粒子, 180
- decay-turtle, 52, 60, 218
- dedxfnc, 159
- dedxfnc1, 162
- dedxfnc2, 162
- delayed neutron, 41
- delt0, 39
- $\Delta$  角分布, 30
- $\delta$  線, 5, 6, 15, 128
- deltb, 39
- deltc, 39
- deltg, 39
- deltm, 39
- deltt, 39
- density, 4, 82, 83
- deposit, 24, 59, 156, 158
- Deposit energy, 177
- dfano, 161
- dir, 55–62, 64, 73, 206
- dircha, 40
- dl0, 79
- dl1, 79
- dl2, 79
- dmax(i), 25, 164
- dnb, 41
- dom, 55–58, 61, 79
- dpa, 6, 171, 173
- dpf, 79
- drd, 79
- DRES, 28
- dresol, 161
- dump, 3, 6, 52, 55, 62–64, 145, 155, 170, 208
- dumpall, 3, 6, 22, 38, 40, 45
- DWBA, 3, 30
- dxw, 79
- dyw, 79
  
- e-dst, 199
- e-mode, 1–3, 31, 50, 51, 156, 164, 167, 171
- e-phi, 199
- e-the, 199
- e-type, 4, 62, 64, 68, 70, 72, 135–137, 148, 153, 155, 159, 167, 180, 183, 206, 207
- e0, 56–62, 64, 206
- e1-t, 138, 162
- e1-type, 162
- e12, 138, 162
- e2-t, 138, 162
- e2-type, 162
- e21, 138, 162
- EBITEM, 4, 28, 165
- eg0, 72
- eg1, 72
- eg2, 72
- eg3, 72
- EGS5, 2, 29, 40, 43
- eiefmax, 26
- eiefmin, 26
- einclmax, 26
- einclmin, 26
- eisobar, 26
- ejamnu, 26
- ejampi, 26
- ejamqmd, 26
- elastic, 185
- electron, 42, 51, 156, 158
- elf, 119
- ELIB, 83

- ELL, 5, 103  
 emax, 130, 150, 151, 173  
 emcnf, 41  
 emcpf, 42  
 emin(i), 25, 156, 174  
 emumax, 29  
 emumin, 29  
 energy cut off 粒子, 180  
 energy stragglng, 31  
 eng, 43, 110, 138, 139, 149, 153, 169, 180, 183  
 eng-t, 138, 159  
 eng1, 162  
 eng2, 162  
 ENSDF, 4  
 eps, 143, 150, 155, 158, 161, 163, 166, 170, 173,  
 176, 179, 182, 185, 195, 197  
 epsout, 143, 150, 155, 158, 161, 163, 166, 169, 173,  
 176, 179, 182, 185, 195–197, 201  
 eqmdnu, 26  
 escape, 180  
 escape 粒子, 180  
 ESTEP, 83  
 et0, 72  
 et1, 72  
 et2, 72  
 event generator mode, 31  
 Excel, 142  
  
 f-curr, 155  
 fac, 173  
 factor, 54, 140, 149, 155, 158, 161, 163, 165, 169,  
 173, 176, 179, 180, 185  
 FAQ, 221  
 fcl, 115  
 file, 24, 139, 149, 155, 156, 161, 163, 165, 169–171,  
 176, 179, 180, 182, 183, 188, 190, 195–  
 198, 201, 206, 207  
 file(20), 2  
 FILL, 86, 87, 92–95, 97–99  
 fission, 169, 185  
 flight mesh, 39  
 fluence, 148, 152  
 flux, 59, 152, 155, 206  
 Fortran, 9, 18, 141  
  
 gap, 116  
  
 GAS, 83  
 gcut, 35, 156, 218  
 GEM, 28  
 GG, 1, 15, 36, 37, 108, 110–112, 114–116, 131, 133  
 GG(General Geometry), 86, 87  
 ghostview, 143  
 ginfo, 150, 155, 158, 161, 166, 169, 173, 176, 179,  
 182, 185, 195, 197, 215  
 GQ, 102  
 grav, 30  
 gshow, 22, 121, 141, 142, 144, 149, 155, 158, 161,  
 166, 169, 173, 176, 179, 182, 185  
 gslat, 150, 155, 158, 161, 166, 170, 173, 176, 179,  
 182, 185, 195  
  
 heat, 156  
 heaven, 199  
 heavyion, 185  
 HEX, 103  
 HLIB, 83  
 HSB, 122  
  
 iaprim, 45  
 iauger, 43  
 ibad, 42  
 ibod, 17  
 ibound, 45  
 ibrdst, 45  
 icntl, 22, 40, 41, 142, 143, 146, 195, 197, 206  
 icput, 37  
 icrhi, 30  
 icxsni, 30  
 idam(i), 41  
 idbg, 17  
 ides, 42  
 idpara, 38  
 idwba, 30  
 iechrl, 150, 158, 161, 163, 169, 173, 176, 182, 185,  
 197  
 iedgfl, 43  
 iegsout, 43  
 iegsrand, 43  
 ieispl, 45  
 ielas, 28  
 ielctf, 30, 119  
 ieleh, 29



- ielms, 28
- igamma, 2, 4, 28, 40, 51, 165
- igchk, 39
- igcut, 35, 156
- igerr, 39
- iggcm, 37
- Ignatyuk, 29
- igpara, 35, 218
- ih2o, 31
- imagnf, 30, 39, 116
- imout, 36
- impacr, 45
- importance, 37, 108, 109
- imucap, 29
- imuint, 29
- INC-ELF, 6, 26
- incelf, 26
- INCL, 5, 6, 26
- inclg, 26
- incohr, 45
- incut, 35, 156, 213, 214
- infl, 13, 17
- info, 140, 165
- infout, 3, 32
- inmed, 28
- inmedium, 28
- inpara, 35, 218
- input echo, 22, 36, 37
- Intel Fortran, 1
- interpolation, 129
- inucr, 41
- ionization potential, 31
- ipara, 37
- ipcut, 35, 156
- iphot, 42
- iphter, 45
- ipngdr, 29
- ipnint, 29, 43
- ipout, 37
- ippara, 35, 218
- iprdst, 45
- ipreeq, 29
- iprofr, 45
- iprtb2, 188
- iraylr, 43
- ireschk, 23
- irqmd, 3, 26
- iscorr, 53
- ismm, 28
- iso, 59
- isobar, 28
- istdev, 5, 23, 217
- istrg, 42
- isumtally, 146
- itall, 13, 36, 143, 219
- itstep, 36, 119
- iunr, 41
- ivout, 37
- ivoxel, 38, 40, 100
- izst, 54, 117, 119
- JAM, 2, 5, 26
- JAMQMD, 2, 26
- jmout, 36, 189
- JQMD, 3, 19, 26
- K/X, 102
- K/Y, 102
- K/Z, 102
- kf コード, 19, 138
- kf-code, 19
- kmout, 36, 83
- KUROTAMA, 2, 6, 30
- KX, 102
- KY, 102
- KZ, 102
- l-dst, 199
- l-phi, 199
- l-the, 199
- l-type, 135, 136, 174
- Landau, 31
- LAT, 86, 87, 93, 95, 97–99, 195
- lattice, 5, 54, 86, 87, 93, 96, 97, 99, 108, 110–112, 114–116, 120, 125, 127, 128, 132, 133, 155, 196
- lattice 座標, 94, 97, 99, 132, 133
- LAT 番号, 142
- let, 159, 162, 174, 176, 177
- letmat, 159, 174, 177
- letmat1, 162
- letmat2, 162
- LET メッシュ, 135

- level, 29
- lib, 173
- library, 173
- LIKE *n* BUT, 86, 96
- line, 201
- Lineal energy, 177
- Linux, 9
- lost particle, 39
- lpolar, 43
- Lynch の式, 31
  
- Mac, 9
- magnetic field, 39
- makefile, 11
- MARS-PF, 22, 40
- mass, 138, 165
- MAT, 86, 87, 96
- mat, 36, 82, 84, 116, 122, 124, 130, 150, 151, 173
- matadd, 36, 86
- material, 113, 143, 148, 150, 156, 158, 159, 161, 162, 164, 166, 167, 169, 171, 173, 174, 176, 177, 179, 180, 182, 183, 185, 199
- maxbnk, 23
- maxcas, 23, 217
- MCNP, 36, 151, 213, 218
- mesh, 5, 94, 97, 100, 131–134, 137, 148, 149, 153, 155, 156, 159, 161–165, 167, 169, 171, 173, 174, 176, 177, 179, 180, 182, 183, 185, 188, 189, 195–198, 206, 207
- mgf, 116, 119
- mirror, 199
- Moliere 理論, 31
- mother, 164, 167, 171, 183
- mset, 130, 150, 151
- mt, 173
- multiplier, 5, 129, 130, 149–151, 207
  
- na, 2, 73
- name, 121, 122, 124
- NASA, 30
- naz, 17
- ncut, 35, 156, 218
- ndata, 164, 165
- ne, 68, 70, 129
- nedisp, 31, 39
- negs, 2, 29, 40, 43
  
- neispl, 45
- nevap, 28
- nfile, 146
- NLIB, 83
- nlost, 39
- nm, 4, 72
- NMTC/JAM, 1
- nn, 2, 4, 74
- no ionization, 22
- no reaction, 22
- nocoh, 42
- non, 116, 121
- nspred, 1, 31, 39
- ntmax, 62, 64
- nuclear, 169, 185
- nucleus, 164, 165
- num, 113
- numb, 42
- nwsors, 37
  
- o-curr, 155
- oa-curr, 155
- ob-curr, 155
- of-curr, 155
- output, 24, 59, 94, 140, 155, 156, 159, 162, 169, 173, 180, 185, 195–199, 206
- outtime, 188
  
- P, 95, 102
- p-type, 68, 70, 72
- param.inc, 3, 14
- part, 108, 110, 115, 120, 127, 130, 137, 138, 148, 150, 151, 153, 158, 159, 162, 167, 171, 173, 174, 177, 180, 183
- pcut, 35, 156, 218
- PE, 13, 23, 35, 36, 38, 217, 218
- phi, 55–58, 61
- phits.in, 13, 217
- photon, 19, 42, 114, 156
- PLIB, 83
- pwt, 114
- PX, 88–90, 92, 93, 95–99, 102
- PY, 88–90, 92, 93, 95–99, 102
- PZ, 88–90, 92, 93, 95–99, 102
- pz0, 60
  
- q-type, 73, 74

- q:, 17  
 qp:, 17  
 r, 138, 149, 153, 156, 161, 164, 169, 171, 176, 179, 180, 183  
 r-in, 155, 207  
 R-JQMD, 3, 26  
 r-out, 155, 207  
 r-type, 134, 136  
 r-z, 131, 134, 153, 155  
 r0, 56, 61  
 r1, 56, 58, 59, 61, 103, 104, 206  
 r2, 59, 61, 103, 104, 206  
 RCC, 103  
 rdam(i), 41  
 REC, 5, 103  
 reg, 22, 54, 62, 64, 94, 97, 100, 108–112, 114–116, 119–121, 126, 127, 131–133, 138, 142, 143, 148, 150, 153, 155, 156, 158, 159, 161–164, 166, 167, 169, 171, 173, 174, 176, 177, 179, 180, 182, 183, 185, 188, 189, 197–199, 206, 207  
 region error, 39  
 resfile, 24, 139, 149, 155, 156, 161, 163, 165, 169, 171, 176, 179, 180, 183  
 resol, 144, 150, 155, 158, 161, 166, 170, 173, 176, 179, 182, 185, 195, 197, 201  
 RHO, 86, 87  
 RHP, 103  
 rijk, 13  
 rn, 57, 58  
 RPP, 103  
 rseed, 23  
 rshow, 22, 121, 141–144, 149, 158, 161, 166, 169, 173, 176, 179, 182, 185  
 rx, 59  
 ry, 59  
 rz, 24, 138, 149, 156, 161, 165, 169, 171, 176, 179, 180, 183  
 S, 102  
 s-type, 52, 55–62, 64, 67, 206  
 $S(\alpha, \beta)$ , 84  
 SDM, 28  
 se-type, 179  
 se-unit, 177  
 sed, 177, 179  
 sfile, 146  
 Shen, 30  
 simple, 156, 173  
 size, 121  
 SO, 102  
 source, 169  
 special, 164  
 Specific energy, 177  
 SPH, 103  
 SQ, 102  
 star density, 183  
 straggling, 42  
 sumfactor, 146  
 sumtally, 146  
 sum タリー, 1, 2  
 surface, 101, 155  
 swtm(i), 27  
 SX, 102  
 sx, 54, 62–64  
 SY, 102  
 sy, 54, 62–64  
 symbol, 19  
 SZ, 87, 89, 90, 102  
 sz, 54, 62–64  
 t, 138, 149, 153, 169, 180  
 t-e1, 138, 162  
 t-e2, 138, 162  
 t-eng, 138, 159  
 t-type, 54, 62, 64, 135, 136, 148, 153, 155, 159, 162, 167, 180  
 t0, 54  
 target, 188  
 tc, 54  
 td, 54  
 tg-list, 188  
 the, 138, 153, 169  
 tim, 110  
 timeevo, 188  
 title, 140, 149, 155, 158, 161, 163, 165, 169, 173, 176, 179, 180, 185, 195, 197, 201  
 tmax(i), 25, 27  
 TMP, 86, 87  
 tmp, 112  
 tn, 54

- totfact, 53, 54, 206  
 TRC, 5, 103  
 TRCL, 54, 86, 87, 96, 105, 116, 144, 150, 155, 158,  
 161, 166, 169, 173, 176, 179, 182, 185,  
 195, 197, 202, 203  
 trcl, 119  
 trcle, 119  
 trclm, 119  
 trxcrd.dat, 2, 28, 31, 40  
 tw, 54  
 TX, 102  
 TY, 102  
 typ, 116  
 TZ, 102  
  
 U, 54, 86, 87, 90, 92, 93, 95, 97–99, 108, 110–112,  
 114–116, 120, 125, 127, 128, 132, 155  
 udtpara, 190  
 unit, 140, 149, 153, 155, 156, 158, 161, 162, 164,  
 168, 171, 176, 179, 180, 183, 206, 207  
 universe, 54, 87, 90, 93, 97, 99, 108, 110–112, 114–  
 116, 120, 125, 127, 128, 132, 155  
 usrelst, 31  
 usrmgt, 31  
 usrsors.f, 52  
 usrtally.f, 190  
  
 Vavilov, 31  
 VOL, 86, 87  
 vol, 111, 133, 150, 158, 161, 163, 166, 170, 173,  
 176, 182, 185  
 volmat, 143, 150, 158, 161, 166, 169, 173, 176, 179,  
 182, 185  
 volume, 37, 111, 133, 150, 158, 161, 163, 166, 170,  
 173, 176, 179, 182, 185, 197, 206  
 voxel, 38, 99  
  
 w-ang, 199  
 w-dst, 199  
 w-hgt, 199  
 w-mnh, 199  
 w-mnw, 199  
 w-wdt, 199  
 wc1(i), 27  
 wc2(i), 27  
 WED, 5, 103  
 weight cutoff, 115  
  
 weight window, 110  
 wem, 59  
 width, 144, 150, 155, 158, 161, 166, 170, 173, 176,  
 179, 182, 185, 195, 197, 201  
 Windows, 2, 9, 13, 18, 219  
 Wobbler magnet, 31, 116  
 wt0, 60  
 wwi, 110  
  
 x, 138, 139, 149, 153, 156, 161, 165, 169, 171, 176,  
 179, 180, 183  
 x-ray, 42  
 x-txt, 143, 149, 155, 158, 161, 163, 165, 169, 171,  
 176, 179, 180, 185, 195, 197, 201  
 x-type, 134, 136, 143, 196, 198  
 x0, 56–62, 64, 134, 199, 206  
 x1, 56, 57, 59, 61, 62, 64  
 xnum, 42  
 xp, 60  
 xq, 60  
 xsdir, 40, 83  
 XY, 102  
 xy, 24, 138, 141–143, 149, 153, 156, 161, 165, 169,  
 171, 173, 176, 179, 180, 182, 183, 185,  
 195, 197, 198  
 xyz, 5, 22, 131, 134, 142, 143, 153, 155, 195, 197  
 xz, 142, 153, 171, 173, 176, 179, 196  
  
 y, 138, 139, 149, 153, 156, 161, 165, 169, 171, 176,  
 179, 180, 183  
 y-txt, 143, 149, 155, 158, 161, 163, 165, 169, 171,  
 176, 179, 180, 185, 195, 197, 201  
 y-type, 134, 136, 143, 196, 198  
 y0, 56–62, 64, 134, 199, 206  
 y1, 56, 57, 59, 61, 62, 64  
 yp, 60  
 yq, 60  
 yz, 138, 142, 143, 149, 153, 156, 161, 165, 169, 171,  
 173, 176, 179, 180, 182, 183, 185, 195,  
 197  
  
 z, 138, 149, 153, 156, 161, 164, 165, 169, 171, 176,  
 179, 180, 183  
 z-txt, 143, 149, 155, 158, 161, 163, 165, 169, 171,  
 176, 179, 180, 185, 197, 201  
 z-type, 134, 136, 143, 196, 198  
 z0, 56–62, 64, 199, 206

- z1, 56–59, 61, 62, 64
- zlin, 197
- zlog, 197
- ZP, 102
- zx, 138, 142, 143
  
- 異常終了, 13, 218
- 階層構造, 97, 108, 132, 134
- 核種, 19, 83, 84, 164, 167, 171, 183
- 核データ, 14, 36, 83, 164, 165
- 角度メッシュ, 135
- 吸収反応, 29
- 境界線, 140, 142, 143, 195, 197
- 切り替えエネルギー, 25
- 空白, 15–18, 21, 86, 132, 138
- 繰り返し幾何形状, 90, 95
- 群数, 68, 136, 137
- 計算打切エネルギー, 5, 25, 29, 35, 40, 156, 173, 174, 213
- 計算打切時間, 27
- 継続行, 136
- 最小値, 39, 68, 103, 136, 137, 143
- 最大値, 68, 103, 136, 137, 143
- 座標変換, 15, 96, 101, 105, 116, 144, 150, 155, 158, 161, 166, 169, 173, 176, 179, 182, 185, 195, 197, 202
- 座標変換番号, 96
- 残留核, 28, 218
- 残留核の $\gamma$ 崩壊, 28, 40
- 四重極電磁石, 116
- 質量密度, 83, 86
- 集合代数, 86, 87
- 詳細モデル, 42
- 時間カット, 27
- 時間メッシュ, 135, 167, 180
- 磁場, 30, 116
- 重イオン, 1
- 重力, 30
- 状態密度, 29
- 蒸発モデル, 28
- 生成核種, 164
- セル, 86, 87
- セル定義文, 86, 94, 104
- セル番号, 86, 87, 91, 93, 96, 97, 99, 131, 132
- 体積, 111, 132, 133, 143, 206
- 体積補正, 143
- 体積、面積計算, 22, 59, 206
- タブ, 16
- 弾性散乱, 28, 41
- 中性子光学, 68, 117
- 中性子捕獲, 41
- 定数定義, 86, 101, 105
- 等高線, 141
- 二重極電磁石, 116
- 馬場, 29
- 標準出力, 12, 21, 40
- 標準入力, 12, 13
- 物質番号, 36, 82, 84, 86, 87, 90, 92, 96, 113, 130, 142, 143, 149, 151, 155, 158, 161, 166, 173, 182, 185, 195, 197
- 分解能, 142, 143
- 偏極率, 55, 118
- 密度, 4, 36, 83, 84, 86
- メッシュ幅, 136, 137
- メモリ共有型並列, 11, 24, 217
- 面記号, 88, 93, 99, 101, 102
- 面積, 155, 206, 207
- 面定義数値, 101
- 面番号, 86, 87, 92, 93, 99, 101, 104
- 余弦, 56–60, 105
- 乱数, 23
- 粒子定義, 137
- 粒子密度, 36, 83, 86, 90
- 領域番号, 108, 110–112, 114, 115, 120, 127, 131–133, 142, 143, 155
- 領域メッシュ, 131
  
- ソースのチェック, 22
- ソース粒子のタリー, 22
- ソースファイル, 218
- ダクトソース, 79
- インクルード, 17, 86, 101, 105
- インクルードファイル, 14
- インストール, 9
- ウエイトウインドウ, 27, 110
- ウエイトカット, 27
- データファイル, 40, 218
- エネルギーメッシュ, 135, 148, 153, 167, 180, 183
- エネルギー分散, 31
- カウンター, 63, 120, 144, 145, 150, 155, 158, 161, 163, 166, 170, 173, 176, 179, 182, 185
- カラープロット, 141

- クーロン散乱, 31
- クラスタープロット, 141
- バッチ, 13, 23, 36, 143, 217, 218
- バンク, 14, 23
- パラメーターセクション, 195, 197
- パラメータセクション, 13, 14, 142, 143
- ヒストリー数, 23, 217
- コメント文字, 17, 86, 101, 105
- コンパイル, 9, 11, 52
- スーパーミラー, 125
- スコアリングメッシュ, 131, 134
- スピン, 55, 63, 118, 145
- セルパラメータ, 86, 87, 90, 96, 112
- セル番号, 121
- ミューオン核反応, 2, 29
- ミューオン捕獲反応, 1, 29
- メッシュ定義文, 132, 134–136, 142, 143
- メッシュタイプ, 136
- メモリ分散型並列, 11, 13, 23, 35, 38, 145, 217–219,  
223
- ライブラリー, 22, 25, 83, 84, 171, 173, 183
  
- 位相, 116
  
- 光核反応, 2–4, 6, 29
  
- 再開計算, 5, 23, 139, 149, 155, 156, 161, 163, 165,  
169, 171, 176, 179, 180, 183, 217
  
- 統計誤差, 23, 139

本ソフトウェアおよび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) までお願いいたします。

## *PHITS Ver.2.76 User's Manual*

---

2011 年 08 月 18 日 Ver.2.30 発行  
2011 年 12 月 02 日 Ver.2.30a 発行  
2012 年 08 月 23 日 Ver.2.30b 発行  
2012 年 09 月 25 日 Ver.2.50 発行  
2012 年 10 月 03 日 Ver.2.51 発行  
2012 年 12 月 27 日 Ver.2.52 発行  
2013 年 08 月 22 日 Ver.2.60 発行  
2013 年 09 月 26 日 Ver.2.62 発行  
2013 年 10 月 23 日 Ver.2.63 発行  
2013 年 11 月 02 日 Ver.2.63a 発行  
2013 年 11 月 19 日 Ver.2.64 発行  
2014 年 01 月 09 日 Ver.2.64a 発行  
2014 年 01 月 30 日 Ver.2.65 発行  
2014 年 02 月 21 日 Ver.2.66 発行  
2014 年 05 月 22 日 Ver.2.67 発行  
2014 年 08 月 30 日 Ver.2.70 発行  
2014 年 09 月 26 日 Ver.2.71 発行  
2014 年 10 月 21 日 Ver.2.72 発行  
2014 年 11 月 05 日 Ver.2.73 発行  
2015 年 01 月 30 日 Ver.2.74 発行  
2015 年 02 月 09 日 Ver.2.75 発行  
2015 年 03 月 17 日 Ver.2.76 発行

*PHITS* 開発チーム :

仁井田浩二<sup>1</sup>, 松田規宏<sup>2</sup>, 橋本慎太郎<sup>2,†</sup>, 岩元洋介<sup>2</sup>, 岩瀬広<sup>3</sup>, 佐藤達彦<sup>2</sup>, 古田琢哉<sup>2</sup>, 小川達彦<sup>2</sup>, 安部晋一郎<sup>2</sup>, 中島宏<sup>2</sup>, 深堀智生<sup>2</sup>, 奥村啓介<sup>2</sup>, 甲斐哲也<sup>2</sup>, 千葉敏<sup>4</sup>,

Lembit Sihver<sup>5</sup>

<sup>1</sup>(財) 高度情報科学技術研究機構, <sup>2</sup>(独) 日本原子力研究開発機構

<sup>3</sup>(共) 高エネルギー加速器研究機構, <sup>4</sup>(国) 東京工業大学, <sup>5</sup>Chalmers University, Sweden

† Ver.2.30 より本マニュアルの編集を担当

---