

demons に関するメモ

田中佳奈

2009.05.13.

概要

demons に関する流れ・詳細事項をまとめる (随時更新)。

1 demons について

計算コード DEMONS(Differential Efficiency for Multi-cell Organic Neutron Scintillators) は多重に分割化されたプラスチックシンチレータの入射中性子に対する応答を評価できる。プラスチックシンチレータ中の水素や炭素と中性子との間で発生する反応を再現している。

2 初期設定

まず、初めて demons を使う際の初期設定についてまとめる。

demons は CERN を使ったプログラムであり、[Makefile] の中で CERN の場所を指定する必要がある。

```
$ which cern          !CERN がある場所を探す
私の場合は「/」の中で、「CERN = /opt/cern/2002」と書き直した。
$ mount /opt/CERN2002 !CERN は mg からマウントして使う
$ df                  !mount 状況確認
mg:/opt/CERN2002      12096760   3956632   7525640   35% /opt/CERN2002
上のようなメッセージが出ていれば OK。
```

3 基本的な使い方

3.1 コンパイル・実行

```
$ make
$ ./demons
```

というコマンドで、コンパイルと実行を行うことができる。実行すると4つのファイル名と energy division を聞かれる。いちいちファイル名を入力するのは面倒なので、以下のようなファイル [demons.in] を用意しておくとう便利。

```

=====
a.hst      !output されるヒストグラムの出力ファイル
input      !input file
b.dat      !出力ファイル
c.dat      !中性子のエネルギー vs 検出効率の出力ファイル
10.0       !中性子の入射エネルギーの刻み幅 energy division [MeV]
=====

```

2つ目が入力ファイル(詳細は次節)、1,3,4つ目が出力ファイルである。上の例では、10MeV刻みに1-250MeVまでの入射中性子の検出効率を順に計算するようになっていて、その結果がc.datに落ちる。さらに検出効率だけではなく、より詳細な計算結果がb.datに落ちている。また、fortファイルに必要な応じて様々な出力が落ちる。

このように書き込んだファイルを用意した上で、

```

$ make
$ ./demons < demons.in

```

として、[demons_core.f]のコンパイルと実行を行う。

エラーが出る場合は、mountが解除されていないか等をチェック。

3.2 input ファイル

inputファイル[input]では、以下のように中性子の入射エネルギー、発生数、標的の位置、中性子の入射の仕方(ペンシルビームなど)、検出器の形状等の情報を設定しておく。

以下はNEBULA用の設定である。250MeVの中性子をペンシルビームとして入射させる。厚さ1cmのVetoカウンターを30本、厚さ12cmの中性子検出器を30本×4面並べている。以下の例では、Vetoカウンターと中性子検出器の間の距離を3cm、中性子検出器の面同士の距離を0cmとしている。

```

=====
nebula      -->メモを書く
$INP
T00 = 250.0 ,      -->1-250MeVの範囲で計算する
rslne = 0. ,
exp_en = 0. ,
en3a = 170. ,
AEV = 4.0 ,      -->Vetoカウンターに当たる粒子の個数を10^4.0個発生
ISEED = 65631,
XI = 0. ,
YI = 0.0 ,
ZI = -1000.0 ,    -->これら3つは中性子発生点(標的)の座標(0,0,-1000cm)
x_width = 1. ,
y_width = 1. ,

```

```

z_width = 1. ,      -->これら 3 つは中性子源のサイズ
ri = 1. ,          -->中性子の発生条件、1=pencil ビーム、-1=4 ビーム
nst = 5,           -->スタックの数、Veto カウンター+中性子検出器 4 面=5
thr_cl = 0.0       -->スレッシュヨルド [MeVee](in cell)
thr_cl_v = 0.5,    -->Veto のスレッシュヨルド [MeVee](in cell)
$END
001 360.000 180.000 30.000 1.000 0.000 3.000 0.500 0.000 380.000
001 360.000 180.000 30.000 12.000 0.000 0.000 0.500 0.000 380.000
001 360.000 180.000 30.000 12.000 0.000 0.000 0.500 0.000 380.000
001 360.000 180.000 30.000 12.000 0.000 0.000 0.500 0.000 380.000
001 360.000 180.000 30.000 12.000 0.000 0.000 0.500 0.000 380.000
1231234567812345678123456781234567812345678123456781234567812345678
nps   hs   ws  acells   ts    tg    tf  efmins egmins attls
=====

```

変数	単位	意味
T00	MeV	中性子の入射エネルギー
rslne		beam energy distribution rslne > 0. -> 標準偏差が rslne のガウス分布 rslne = 0. -> const. rslne < 0. -> HWHM = rslne の平坦な分布
exp_en	MeV	指数関数型中性子エネルギー分布に関するオプション。 exp_en = 0. -> 効果なし exp_en > 0. -> $\exp(E_n/E_0)$ 分布
aev		発生イベント総数= 10^{aev} 個
iseed		乱数の種
xi,yi,zi	cm	これら 3 つは中性子発生点 (標的) の座標。 座標の原点は第一面目 (ここでは Veto カウンター) の前面の中心である。
ri		中性子源に関するパラメータ。 ri >= 0 -> ビーム半径=ri のペンシルビーム -1 <= ri < 0 -> 点源。4 ビーム。 ri < -1 -> 全面積への並行照射。
x_width	cm	点源に対して、xi,yi,zi は $\pm x_width$, $\pm y_width$, $\pm z_width$ 上に
y_width	cm	一様に分布させられる。
z_width	cm	
nst		スタックの数
thr_cl	MeVee	セル内部でのスレッシュヨルド。左右の光電子増倍管からの信号の波高に対して適用され、ソフトウェア的なスレッシュヨルドをシミュレートする。
thr_cl_v	MeVee	Veto カウンターのセル内部でのスレッシュヨルド
nps(i)		i 番目のスタックの面の数
hs(i)	cm	i 番目のスタックの高さ (x 方向の長さ)
ws(i)	cm	i 番目のスタックの幅 (y 方向の長さ)
acells(i)	cm	i 番目のスタックの一面当たりのセルの数
ts(i)	cm	i 番目のスタックの一面分のプラスチックの深さ (z 方向の長さ)
tgs(i)	cm	i 番目のスタックのプラスチックとプラスチックの間隔
tf(i)	cm	i 番目のスタックの背面とその次のスタック前面の間隔
efmins(i)	MeVee	i 番目のスタックの左右の個々の光電子増倍管に個別に適用される光出力のスレッシュヨルド値。ディスクリミネータ (ハードウェア) のスレッシュヨルドをシミュレートする。
egmins(i)	MeVee	i 番目のスタック後方のギャップ領域に適用されるエネルギー損失のスレッシュヨルド値。
attls(i)	cm	i 番目のスタックを構成する検出器の減衰長

表 1: input ファイルのパラメータの説明

4 全体の流れ

demons はエネルギーや方向を与えた中性子を 1 つ入射させた時、モンテカルロ法によって、検出器で検出される位置・時刻を求めるプログラムである。demons の大まかな流れを図 1 に示す。

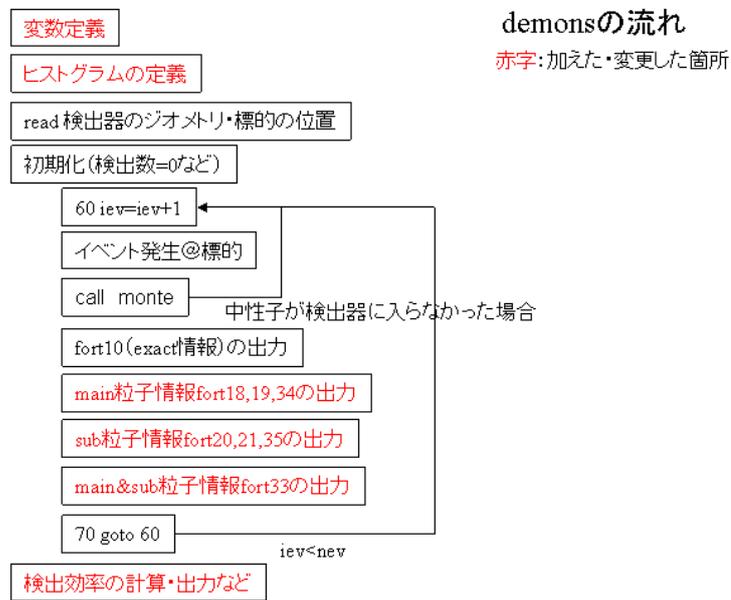


図 1: demons の流れ

5 サブルーチン monte の流れ

monte は demons 内の重要なサブルーチンで、大まかな流れを図 2 に示す。中性子が検出器に入ってから、検出されるまでのプログラムである。フォトマルの左右両方で、スレッシュヨールド以上の発光量で検出されるイベントを good event と呼んでいる。

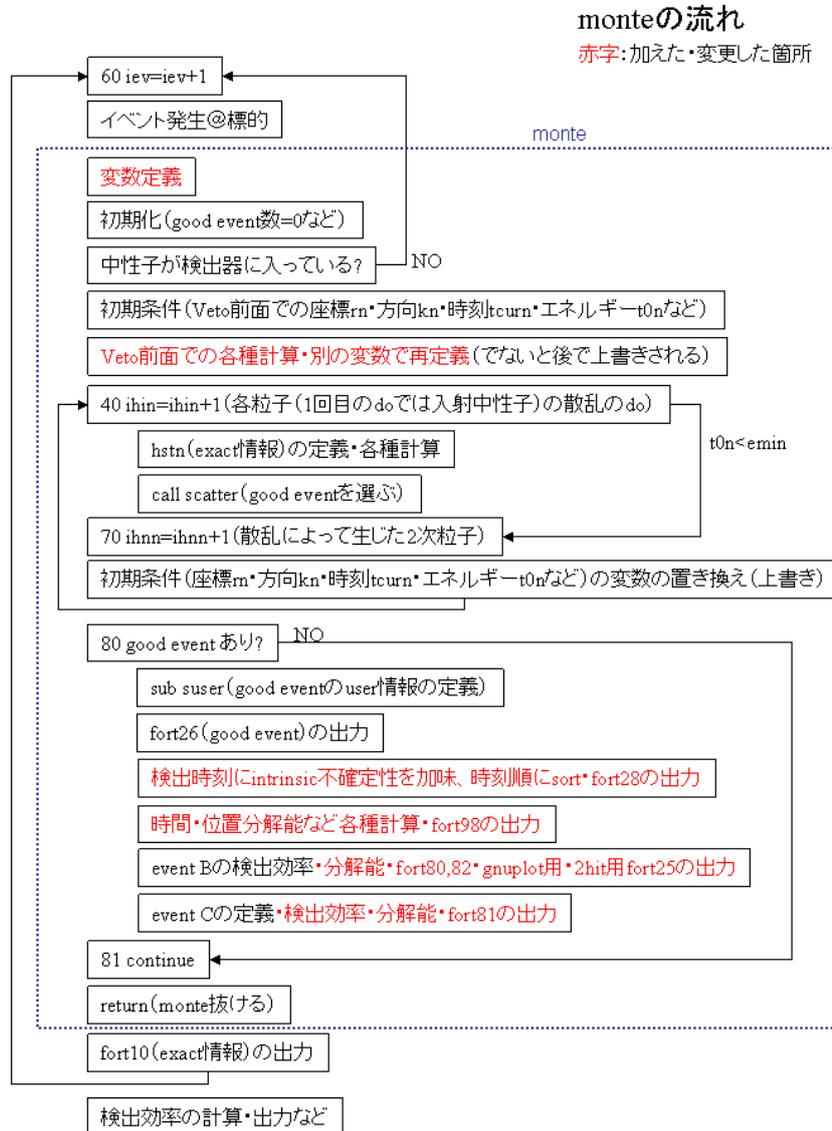


図 2: monte の流れ

6 ヒストグラムを書くためには

サブルーチン `hbook` と `hfill` をセットで用いる。これは CERN の中に入っているライブラリで、コンパイルした時に読み取ってくれる。まず `demons` の冒頭部分で

```
=====
Call Hropen(88,'NEW',pawfile,'N',4096,Ista)
=====
```

とおまじないを書いておく。次に `demons` の冒頭部分で、ヒストグラムの箱を定義をする。

1次元の場合

```
=====
call hbook1 (6,'E distribution',100,0.,70.,0.)
=====
```

2次元の場合

```
=====
call hbook2 (50,'PH 1 vs PH 2',50,0.,70.,50,0.,70.,0.)
=====
```

括弧内は順に、ID 番号、'グラフ名'、x 軸のヒストグラムの階級、x 軸の範囲、y 軸のヒストグラムの階級、y 軸の範囲、0.0(重み?) という意味である。

次にヒストグラムにしたい値を格納 (`fill`) する。`demonos` 内ならどこに書いても良いが、計算で値を求めた直後だとわかりやすい。

1次元の場合

```
=====
call hfill(または hf1) (6,t0,0.,1.)
=====
```

2次元の場合

```
=====
call hfill(または hf2) (50,phavg1,phavg2,1.)
=====
```

括弧内は順に、ID 番号、横軸の変数、縦軸の変数 (1次元の場合は 0)、重み、という意味である。重みはそのイベントを何カウントとして数えるか、を表しており、通常 1 で良い。

`paw` を立ち上げて、

```
=====
> hi/file 1 work.hst 4096 !lun1 というディレクトリに work.hst のデータを入れる
> i !i=hi/list、lun1 内のファイル一覧を表示
> ht 6 !ht=hi/plot、ID 番号 6 のヒストグラムを表示
=====
```

ヒストグラムにする変数が `real*4`(単精度) でないとエラーが出るので注意。

7 詳細・注意事項など

・ $ihnn=ihnn+1$ (次の粒子へ) とは、入射した中性子の散乱によって $2n$ が放出された際の2つの中性子などのこと。

・ org の $demons$ は検出器に入った個数 (Veto 前面に当たった個数) が nev になるように、何度も中性子を発生させ直す。

・ $user(1,i)$ は検出器とその間のガス (空気やアルゴンなど) に順に付けられた番号。変数 ipl は $user(1,i)$ のこと。 $np1_bb(i)=int(0.5*user(1,i))$ は検出器の番号。

	Veto	ガス	検出器 1	ガス	検出器 2	ガス	検出器 3	ガス	検出器 4
$user(1,i)$	1	2	3	4	5	6	7	8	9
$np1_bb(i)$	0	1	1	2	2	3	3	4	4
各面の厚さ	$t(1)$	$t(2)$	$t(3)$	$t(4)$	$t(5)$	$t(6)$	$t(7)$	$t(8)$	$t(9)$

表 2: 面番号

8 ファイル「fort.26」

ファイル「fort26」は $user$ を用いて以下のような実験で観測されるデータを出力している。ただし、同じセル内で複数回反応したものは区別せず足し合わせて出力される。

変数	意味
iev	何番目のイベントか (10000 イベント中当たったものだけ取り出す)
$igdev$	各入射中性子に対して、何番目の good event か
nci	initialize no. of c interactions
$np1_bb(i)$	plane number
$user(1,i)$	plane number
$user(2,i)$	cell number
$user(3,i)$	time left
$user(4,i)$	time right
$user(5,i)$	light left
$user(6,i)$	light right
$user(7,i)$	

表 3: fort26 の出力

9 ファイル「fort.10」

ファイル「fort.10」では表 4 のような $hstn(j,i)$ のデータが出力されている。「fort.26」とは異なり同じセル内で複数回反応したものも区別して出力される。また、good event 以外の微弱な反応のデータもすべて出力される。中性子が衝突することで反跳した粒子も追跡していく。

変数	意味 (main 粒子=入射中性子または int4 で放出された γ 線)
neve	何番目のイベントか
ihnn	1 イベント当たりの main 粒子の数。(ほとんど入射中性子の 1 個のみ)
ihin(i)(i=1,ihnn)	各イベントにおいて i 番目の main 粒子 (中性子や γ 線) の反応の回数。
hstn(1,i)=nd	main 粒子の種類。
hstn(2,i)=ich	反応番号 (中性子の場合は=0 としてあり、Veto での初期条件を与える)
hstn(3,i)=icell	cell number
hstn(4,i)=ipl	plane number (検出器の間のガスも含めた面番号)
hstn(5,i)=rn(1)	x in cell セル方向
hstn(6,i)=rn(2)	y in cell 横方向
hstn(7,i)=rn(3)	z in cell ビーム方向
hstn(8,i) ~ hstn(10,i)=kn(1) ~ kn(3)	direction cosine
hstn(11,i)=t0n	反応後の neutron kinetic energy
hstn(12,i)=tcurn	neutron interaction time
hstn(13,i) ~ hstn(23,i)	1 回目の散乱。hstn(2,i) ~ hstn(12,i) と同じ。
hstn(24,i) ~ hstn(34,i)	2 回目の散乱。hstn(2,i) ~ hstn(12,i) と同じ。
変数	意味 (sub 粒子=反跳荷電粒子または反跳中性子または 2 次的な γ 線)
ihnc	1 イベント当たりの sub 粒子の数。
ihic(i)(i=1,ihnc)	i 番目の sub 粒子の反応の回数 (ほとんど 1 回)。
hstc(1,i)=nd	sub 粒子の種類。
hstc(2,i)	main 粒子のどの反応で発生したかという反応の種類。
hstc(3,i) ~ hstc(12,i)	1 回目の散乱。hstn(3,i) ~ hstn(12,i) と同じ。
hstc(13,i) ~ hstc(22,i)	2 回目の散乱。hstc(3,i) ~ hstc(10,i)、hstc(12,i)、hstc(11,i) と同じ (逆転)。
hstc(23,i) ~ hstc(31,i)	3 回目の散乱。hstc(3,i) ~ hstc(10,i)、hstc(12,i)、hstc(11,i) と同じ。

表 4: fort.10 の出力

main 粒子が中性子の場合、1 ループ目では必ず $hstc(2,i)=0$ (反応なし) として設定あり、 $hstn(3,i) \sim hstn(12,i)$ は初期条件 ($z=-0.5\text{cm}$ での x,y 座標や入射方向、入射エネルギー (ここでは 250.0MeV)) を与えている。

$hstn(2,i)$ については反応によって表 5 の番号がつけられる。

main 粒子が γ 線の場合の、反応番号 $hstn(2,i),hstc(2,i)$ と反応	
int1: 光電効果	
int2: コンプトン散乱 (放出される 線は sub 粒子として追跡される)	
int3: 対生成	
main 粒子が中性子の場合の、反応番号 $hstn(2,i),hstc(2,i)$ と反応	粒子番号 $hstn(1,i),hstc(1,i)$ と粒子
int1: $n+p \quad n+p$	par0: γ 線
int2: $n+^{12}\text{C} \quad n+^{12}\text{C}+\gamma$ (非弾性散乱)(γ 線の検出)	par1:n(検出できない)
int3: $n+^{12}\text{C} \quad n+^{12}\text{C}$ (弾性散乱)	par2:p
int4: $n+^{12}\text{C} \quad n+^{12}\text{C}+\gamma$ (非弾性散乱) (^{12}C による中性子の検出)	par3: ^{12}C
int5: $n+^{12}\text{C} \quad \alpha+^9\text{Be}$	par4: α
int6: $n+^{12}\text{C} \quad n+3\alpha$	par5: ^9Be
int7: $n+^{12}\text{C} \quad n+p+^{11}\text{B}$ または $n+n+^{11}\text{C}$ または $p+^{12}\text{B}$	par7: ^{11}B
	par9: ^{11}C
	par10: ^{12}B

表 5: 反応と粒子の種類