

統計模型計算コード CoH₃

ロスアラモス国立研究所

河野 俊彦

kawano@lanl.gov

1. はじめに

2024年10月、統計模型計算コード CoH₃ をオープンソースソフトウェアとして BSD3 ライセンスで公開した。仕事の相棒として長らく使ってきたコードなのでご存知の方も多いと思うが、keV から MeV 領域での複合核反応断面積を計算する。所謂 Hauser-Feshbach 統計模型計算コードは数多く書かれているが、大雑把に2つのカテゴリに分けられる。一つは2体反応だけを計算するもので、ELIESE-3 [1] や CASTHY [2] がこのカテゴリに含まれる。数 MeV 程度までの計算であれば2体計算で十分なので、天体核物理分野でこういったコードが幾つか開発されてきた。さらに高いエネルギー領域では複数の粒子が蒸発するので、前平衡過程計算を含んだ多段階の Hauser-Feshbach 計算が必要になる。自分が学生の頃は P.G. Young による GNASH の独壇場であったが、現在では CCONE [3]、TALYS [4]、EMPIRE [5] などが一般的であろう。

CoH₃ について一番良く聞かれる質問は「CoH はどういう意味」というもの。プログラムを書き始めたのは大学院生時代で、自分の勉強も兼ねつつ光学ポテンシャルのエネルギー依存性を調べようとしたのが当初の目的だった。無料の C コンパイラを用い、16bit パソコン上で走るコンパクトなものを目指したので Compact optical model、作り始めたときは簡単に Co みたいな名前だったかもしれない。その後 Hauser-Feshbach が追加されたので CoH、さほど捻った名前では無いが、現在は Compact どころか Complicated である。オフィシャルには「特に意味はない」ことにしている。読み方も特に決まりは無いがシーオーエイチ派とコー派に分かれるようである。稀にコエッチと読む人もいる。

2. CoH 開発史

2.1 version 1 & 2

中性子入射反応では Coulomb 相互作用が無いので、光学模型を計算するプログラムを書くのは比較的容易い。五十嵐氏の ELIESE-3 のマニュアル [1] を参考に1次元 Schrödinger 方程式を解くプログラムを C 言語で書いてみた。全断面積などの計算はすぐにできたものの、実験データからポテンシャルパラメータを決めるためには弾性散乱角度分布も計算する必要があ

る。さらに低エネルギー領域では複合核弾性散乱というものも必要らしい。Hauser-Feshbach理論を使えば計算できるが揺らぎの補正が必要、そのうち欲が出てきて荷電粒子入射反応も計算してみたい、でも Coulomb 関数の計算はややしこそう。いつの間にか ELIESE-3 と同等のコードを C 言語で書いており、ver. 1 として一応の完成を見たのは 1992 年 5 月 11 日と旧 CoH のドキュメントファイルに書かれている。

その後、捕獲反応を含めたチャンネル数の増加や、ポテンシャルパラメータサーチ機能を搭載するなど、十分実用的なものになっていったので、ELIESE-3 の出番も無くなってきた。今となっては想像もできないが、当時のパソコンで使えるメモリはたったの 640kB である(さらに増やすテクニックはあった)。メモリの節約と計算効率は反比例の関係にある。無駄な計算を避けようと中間データを温存すればメモリを圧迫する。メモリをケチるには、重複計算に目を瞑る。とにかくこの世は住みにくい。とは言え C 言語ではポインタを使ってメモリ空間を自由に操作できるので、かなりポインタを乱用したコードになっていた。1995 年、ANSI C 標準に則ってコードを全面的に書き直し、ver. 2 とした。標準的な C コンパイラさえあればよいので、コンピュータも PC-9801 や RISC ワークステーション、ノートパソコン上の Linux と、ダウンサイジングが進んでいった。

ver. 2.3 ではチャンネル結合計算も可能になったが、大規模な反応断面積計算はまだ GNASH 頼りで、全て自分のコードで計算したいという野望が次第に膨らんでいった。実はその当時、GNASH を独自に書き直した GNASHACK というコードを書いていた。しかしある日、うっかり作業ファイルの保存を忘れるという事故を起こして気力を失い、放置されたままとまっている。そもそも FORTRAN はもう使いたくないという気持ちも強かった。

2.2 version 3

2003 年に渡米し、その後さほど進展もなく CoH+GNASH を使って仕事を続けていた。当時 LANL では同僚が GNASH の息子 McGNASH を書き始めており、あまり関わらないつもりでいた。しかし諸事情で自分も開発に加わることになり、ソースコードの現状に衝撃を受けた。大規模プログラムを書くテクニックがあまりにも欠けており、特にメモリ関連でバグに悩まされるのは明らかだった。McGNASH を続けるのは時間の無駄と、2008 年冬休みに CoH ver. 3 を書き始めた。

さすがに無印 C の時代ではないので、CoH2.3 をベースに全てを C++ で書き直した。他の目的で書いていた別のプログラム、Direct/Semidirect 捕獲計算や DWBA 計算、最小二乗法のツール等を統合し、Exciton モデルによる前平衡過程計算を追加して、一応の断面積計算ができるようになるのには半年もかからなかった。かつて GNASH のソースコードを詳細に読んで経験が活かされている。ちなみに勤務時間には一切コーディングをしていない。全て帰宅後と週末のホビーとしてコーディングした。

勿論断面積計算だけでは評価済み核データファイルの作成には使えない。放出粒子のエネルギー・角度分布や γ 線生成、そして最も複雑な排他的 (Exclusive) スペクトルまで計算できるようになったのは 2009 年のことである。この頃から ENDF ファイル操作のためのコード DeCE [6] を平行して開発しており、両者を使って ENDF フォーマットの核データファイルを作成することが可能になってきた。なお DeCE もオープンソースとして公開している。

ところで当初は CoH_3 という名前ではなく、CGM という名前で書き始められていた。Hauser-Feshbach 計算を Monte Carlo 法でも計算できるようにしたため [7, 8]、別の名前のほうが自然である。今となっては理由も思い出せないのだが、何故か CGM は CoH_3 から Monte Carlo 計算部分を抜き出した独立のコードとなり、さらに P. Talou が核分裂関連のドライバを付け加えた CGMF [9] というコードになってリリースされている。CGM はそちらに取られてしまったので、ver. 3 ということで CoH_3 に落ち着いた。バージョン番号が添え字になっているが、これといった理由はない。なんとなく水素化コバルトみたいな化学記号っぽく見えるかなというだけである。ブランド戦略というわけでもないが、ロゴに光学模型のような回折縞の絵を描き (図 1)、これをずっと使っている。

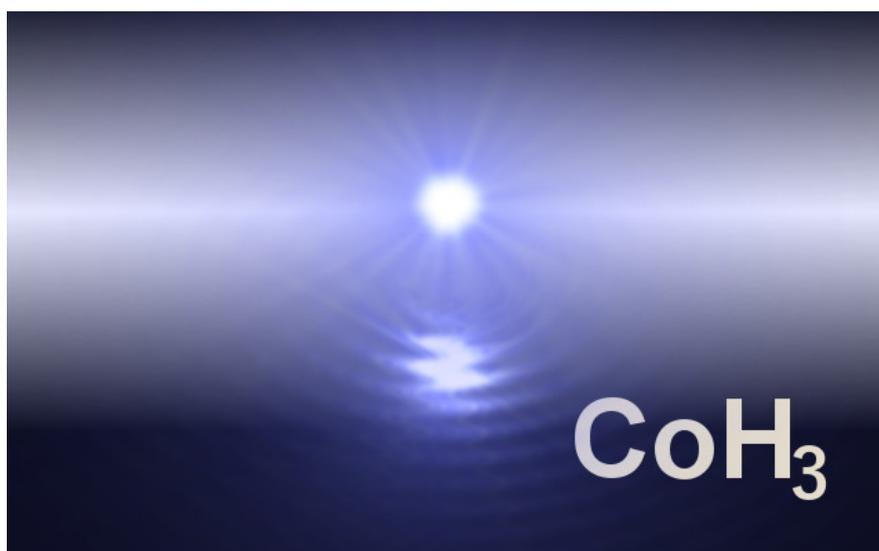


図 1 CoH_3 logo

3. CoH_3 の特徴

Hauser-Feshbach コードが幾つもある市場で、 CoH_3 にもセールスポイントが幾つかある。最初に目指したのは GNASH で苦労した (n,np) と (n,d) 反応の区別だった。GNASH ではこれらの残留核をひとまとめにするので、それぞれの断面積を分離するにはちょっとしたトリックが必要だった。図 2 の左側に複合核 $(Z, A + 1)$ が中性子、陽子、重陽子、三重水素を放出する様子を示している。計算で現れる複合核の総数は 6 個である。 CoH_3 ではそれらをまとめないので、各々の反応断面積を明確に分離できる。図 2 の例では 10 個の複合核オブジェクトが作られる。しかしこのアルゴリズムは高エネルギー計算には向かない。 (n,α) , $(n,2n2p)$, $(n,n^3\text{He})$ などの反応は全て同じ残留核を生成するが、 CoH_3 はそれらを独立に計算する。その結果生成されるオブジェクトに必要なメモリは増大し、計算の経路も非常に複雑なものとなる。 CoH_3 は複合核の崩壊経路の結合リスト (図 2 右側の矢印の情報を持つデータ) を自動的に生成してエネルギー的に可能な全ての経路を追跡しながら計算するため、入射エネルギーが高くなると計算に何十分もかかるようになる。

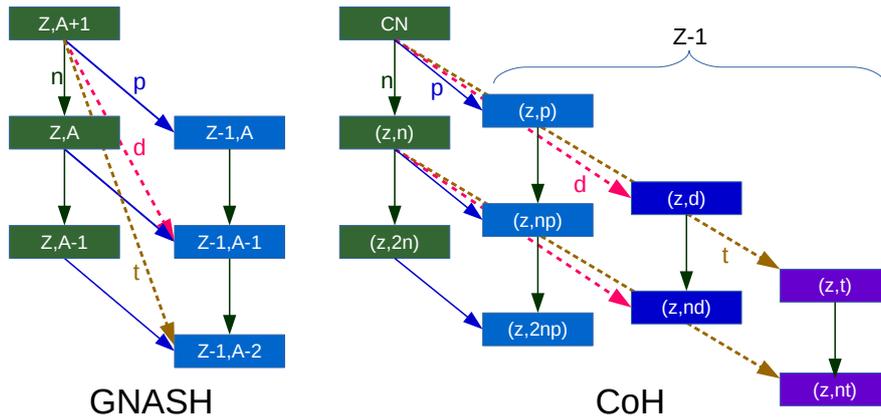


図2 複合核の崩壊経路。左はGNASHの方法、右はCoH₃。例えばGNASHの $(Z-1, A-1)$ はCoHでは $(z, n+p) + (z, d)$ と分離して計算される。

CoH₃は透過係数の補間を行わない。多くのHauser-Feshbachコードは透過係数を外部プログラムで予め計算しておき、必要なエネルギーでの値を補間で得ている。CoH₃は全ての透過係数をそのエネルギーで内部計算するので補間による誤差は生じない。その代償として、実行時に多数の光学模型計算を内部で行う必要がある。但しこのオーバーヘッドはさほど大きくはない。

CoH₃は他コードやライブラリに(ほぼ)依存しない。CCONEを例外として、他のHauser-Feshbachコードは透過係数計算を既存のコードに委ねている。一方CoH₃の内部光学模型ソルバは勿論CoH2.3から引き継がれたものである。特定のライブラリを使うことを前提にしてないので、インストールも簡単に行える筈である。唯一の例外が行列対角化で、CoH₃にはFORTRAN数値計算ライブラリEISPACKをf2cでCに変換したものが含まれる。LAPACKを呼ぶことも可能だが、行列対角化は断面積の本流では使われないので、さほど問題ないであろう。Clebsch-Gordan係数やCoulomb関数などは汎用科学ライブラリに含まれているが、これらも自作ソースコードとしてCoH₃に含まれているので、コンパイル時にライブラリをリンクする必要はない。

CoH₃のポリシーではあるが、滅多矢鱈にあれこれモデルを詰め込まない。例えば準位密度を計算するモデルが幾つか存在する。それらをモデル1、モデル2とコードに含めてあとはユーザーが好きなものを選ぶといった作り方はしていない。準位密度モデルとしてはGilbert-Cameron [10, 11]のみであり、その内部パラメータを変化させることで準位密度の計算結果への寄与を調節するようにしている。理由はどのモデルもそこそこ同じような数値を与えること、モデル1-5みたいな選択肢を与えるとユーザーは全部試してみてもその変動を「計算誤差」としがちなのでそれを避けること。

4. CoH₃ よもやま話

バージョン毎のニックネームは太陽系の月から取られている。木星のガリレオ衛星、Ganymede, Callisto, Io, Europaと進んで行ったが、これらはCGM時代の名前である。CoHと

名前を戻した後は天王星の月となり、ver. 3.1 が Ariel, Umbriel と続いて、現在の ver. 3.6 は Cordelia。シェークスピアの作品から名前を取ってるわけではないが、そう思ってる人は多い。なぜ木星の次が土星ではなく天王星なのかは定かではないが、単に土星のことを忘れていたのだと思う。

C 言語だったものを C++ で書き直しているの、やはり C っぽさはあちこちに残っている。元々が 16bit パソコン用だったこともあり、メモリを極力節約するような書き方もしている。その極端な例がビット操作。8bit のそれぞれのビットを使って 8 個分のスイッチを作る。例えば計算出力制御に使われており、0x01 (=0001) なら基本の情報のみ、0x02 (=0010) なら断面積結果のみ、0x03 (=0011) だとその両方といった具合。ちなみに 0 を与えると計算はするが何も出力しないモードになる。もう一つの例が同位体を指定する際の (Z, A, M) の組み合わせ。これをナイーブに 3 次元配列にすると無駄なスペースだらけになる。CoH₃ では Z, A, M に数 bit ずつ割り当てて一つの整数で表し、この数値がいわばハッシュ値の役割を果たしている。なお同僚からビット操作は理解できないと苦情が来たので、現在のコードでの使用は控えめになっている。

乱用というほどでもないがポインタは多用されている。特に多次元配列には vector クラスを使わないので、ポインタのポインタのポインタの... という宣言があちこちにあり、これも同僚から嫌われる要因となっている。

同僚からのもう一つの苦情、それは彼がソースコードを読んでいてある箇所に辿り着いたときのこと。そこに書かれているのは、

```

/*****/
/*      Calculation of Particle Emission Spectra      */
/*      -----      */
/*      The most complicated subroutine in CoH.      */
/*      You are not supposed to understand this.      */
/*****/

```

実際ここは自分でもしばしば理解不能になる。再帰計算にすればもう少しすっきり書けるのだが、そうするとデバッグが面倒になるのでそのままにしている。

以前核データニュースに ECIS の記事 (核データニュース No.127) を書いたときに少し触れたが、CoH₃ は変形核に対する Hauser-Feshbach 理論を最も正しく計算する。根幹の理論は以前からあったものの、実際に計算できるようにしたのは CoH₃ が最初だと自負している。特に変形核での揺らぎ補正は困難で、ランダム行列を利用した定式化ができたのは 2016 年になってからであった [12]。理論の詳細は省略するが、その中に $e^{i(\phi_\alpha - \phi_\beta)}$ という位相因子が含まれる。これを $\phi_\alpha = \tan^{-1} \tilde{S}_{\alpha\alpha}$ と与えている理由が論文に書かれていない。実はこれ「まあこんなもんやろ」と計算結果を見ながら自分で推定したもの。これを思いついたとき、ふと昔の記憶が蘇った。ELIESE-3 の中の

$$b = \exp [2i \arg (\phi(r_M))] \quad (1)$$

という式の由来が分からず、ついに作者の五十嵐氏に直接尋ねたのである。そのときの氏の答えが「こんなもんやろ」であった。御本人は覚えておられないかもしれないが、自分はしっ

かりと記憶している。「こんなもんやろ」が30年後にひょっこり顔を出して当時を思い出し、にやりとしたものである。

月にちなんだニックネームを付けてることもあり、ちょっとした Easter Egg も仕込まれてる。隠しているというほどはないが、月齢に従ってバナー表示される詩が変化する。最初は A. Giraud の詩 *Pierrot Lunaire* を O.E. Hartleben がドイツ語に翻訳したものをバナーに表示していたが、その後幾つかの詩を追加し、月齢によって変わるようにした。最後に追加したのは中原中也の「湖上」で、自ら英訳したもの(図3)。理系人間なのでどれだけ文学っぽくなってるかは全く不明。ちなみに月齢を正確に計算するのは案外面倒だとわかり、CoH₃ の中では簡易式が使われている。

```
#####
#
#
# On the Lake .oooooooo. ooooo ooooo
# ===== d8P Y8b 888 888
# 888 .ooooo. 888 888
# When the moon has risen and floating 888 d88 88b 888ooooo888
# Let us push a boat into the lake and go sailing 888 888 888 888 888 dP^Y8b
# Waves may gently hit the boat 88b ooo 888 888 888 888 8P
# and a light breeze may blow too Y8bood8P Y8bod8P' o888o o888o <8B
# 8b 8b
# It may be dusky when off the shore Still you are talking on and on Y8oo8P
# The sound of water dripping from the oars sometimes rambling, sometimes sulky
# may hint the intimate feeling while I'm listening to all of your words
# --- in the silence between your words --- but never stop my rowing hands
#
# The moon may be listening secretly When the moon has risen and floating
# and may come down a little Let us push a boat into the lake and go sailing
# At the moment we kiss Waves may gently hit the boat
# the moon would be just above our heads and a light breeze may blow too
#
# Chuya Nakahara
#
##### Moonage 20
```

図3 CoH₃ バナーのひとつ。

5. おわりに

CoH₃ のソースコードは GitHub で公開されている。

<https://github.com/toshihikokawano/coh3>

main ブランチが安定版、develop ブランチが開発版となっている。新版リリースの期間は特に決めてないが、これまでは数ヶ月から1年毎にリビジョンを上げている。次の3.6.1-Cordelia はメンテナンスリリースとなり、おそらく2025年の早い時期に出せると思う。

少し触れたが CoH₃ は粒子やγ線の放出を Monte Carlo 計算で追跡できる。このイベント生成機能は実験装置の検出効率シミュレーションで何度か使われた程度で、今のところさほど役立ってはいない。究極の目標として、核データファイルを作成するのではなく、CoH₃ そのものを粒子輸送計算におけるイベントジェネレータとして使いたい。これにより核データをファイル化すると失われてしまう中性子やγ線の相関を粒子輸送計算に取り入れることができる。図4は⁵⁶Fe(n,2n)反応での2個の中性子がどのような関係で放出されるかの確率を

表しているが [7]、このような情報は Monte Carlo 法でしか得ることができない。計算速度の問題で現状ではまだ実用的ではないが、核分裂片からの粒子放出を追う CGMF コードは既に MCNP の中に含まれ、核分裂イベントジェネレータとして利用されている。共鳴領域はどうしようもないが、より高いエネルギー領域での原子核から原子分子スケールに跨る完全な Monte Carlo 計算はそう遠くない将来に実現可能ではないかと考えている。

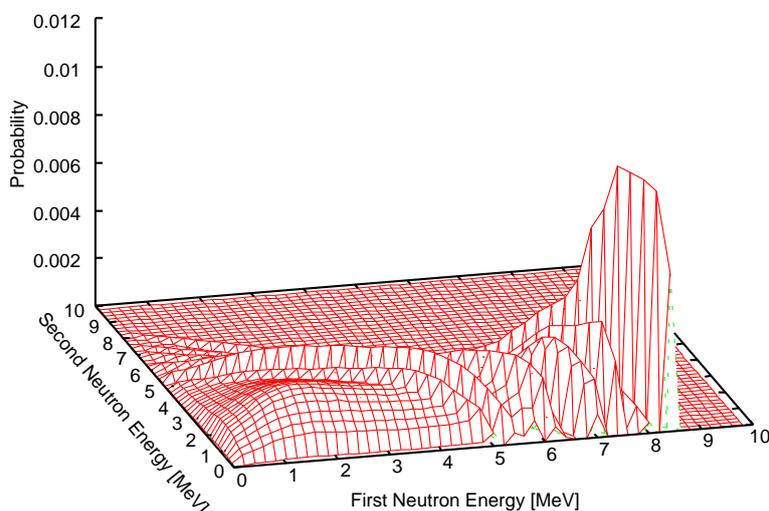


図 4 ^{56}Fe への 20 MeV 中性子入射反応で放出される 2 個の中性子の相関。

参考文献

- [1] S.-i. Igarasi, “Program ELIESE-3; program for calculation of the nuclear cross sections by using local and non-local optical models and statistical model,” JAERI-1224, Japan Atomic Energy Research Institute (1972).
- [2] S.-i. Igarasi and T. Fukahori, “Program CASTHY — statistical model calculation for neutron cross sections and gamma ray spectrum —,” JAERI-1321, Japan Atomic Energy Research Institute (1991).
- [3] O. Iwamoto, “Development of a comprehensive code for nuclear data evaluation, CCONE, and validation using neutron-induced cross sections for uranium isotopes,” *J. Nucl. Sci. Technol.*, **44**, 687 (2007).
- [4] A. J. Koning and D. Rochman, “Modern nuclear data evaluation with the TALYS code system,” *Nucl. Data Sheets*, **113**, 2841 (2012).
- [5] M. Herman, R. Capote, B. V. Carlson, P. Obložinský, M. Sin, A. Trkov, H. Wienke, and V. Zerkin, “EMPIRE: nuclear reaction model code system for data evaluation,” *Nucl. Data Sheets*, **108**, 2655 (2007).

- [6] T. Kawano, “DeCE: the ENDF-6 data interface and nuclear data evaluation assist code,” *J. Nucl. Sci. Technol.*, **56**, 1029 (2019).
- [7] T. Kawano, P. Talou, M. B. Chadwick, and T. Watanabe, “Monte Carlo simulation for particle and γ -ray emissions in statistical Hauser-Feshbach model,” *J. Nucl. Sci. Technol.*, **47**, 462 (2010).
- [8] T. Kawano, P. Talou, M. B. Chadwick, and T. Watanabe, “Particle-gamma and particle-particle correlations in nuclear reaction using Monte Carlo Hauser-Feshbach model,” *Prog. Nucl. Sci. Technol.*, **2**, 757 (2011).
- [9] P. Talou, I. Stetcu, P. Jaffke, M. Rising, A. Lovell, and T. Kawano, “Fission fragment decay simulations with the CGMF code,” *Comp. Phys. Comm.*, **269**, 108087 (2021).
- [10] A. Gilbert and A. G. W. Cameron, “A composite nuclear-level density formula with shell corrections,” *Can. J. Phys.*, **43**, 1446 (1965).
- [11] T. Kawano, S. Chiba, and H. Koura, “Phenomenological nuclear level densities using the KTUY05 nuclear mass formula for applications off-stability,” *J. Nucl. Sci. Technol.*, **43**, 1 (2006).
- [12] T. Kawano, R. Capote, S. Hilaire, and P. Chau Huu-Tai, “Statistical Hauser-Feshbach theory with width-fluctuation correction including direct reaction channels for neutron-induced reactions at low energies,” *Phys. Rev. C*, **94**, 014612 (2016).